

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное  
учреждение высшего образования  
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ

---

# ШЕСТЬДЕСЯТ ДЕВЯТАЯ МЕЖДУНАРОДНАЯ СТУДЕНЧЕСКАЯ НАУЧНАЯ КОНФЕРЕНЦИЯ ГУАП

Часть I

ТЕХНИЧЕСКИЕ НАУКИ

Сборник докладов

18–22 апреля 2016 г.



Санкт-Петербург  
2016

УДК 001  
ББК 72  
Ш51

Ш51                   Шестьдесят девятая международная студенческая научная конференция ГУАП:  
Сб. докл.: В 2 ч. Ч. I. Технические науки / СПб.: ГУАП., 2016. 419 с.: ил.  
ISBN 978-5-8088-1122-5(Ч. I)  
ISBN 978-5-8088-1121-8

Представлены доклады Шестьдесят девятой международной студенческой научной конференции ГУАП, прошедшей в апреле 2016 г.

Научные работы студентов ГУАП посвящены актуальным проблемам авиационного и аэрокосмического приборостроения, проблемам развития новых информационных технологий, радиотехники, электроники и связи, современным проблемам экономики, философии и права.

Под общей редакцией  
доктора юридических наук, профессора  
*В. М. Боера*

Редакционная коллегия:

*А. Р. Бестугин*, доктор технических наук, доцент,  
*А. С. Будагов*, доктор экономических наук, доцент,  
*Е. А. Крук*, доктор технических наук, профессор,  
*К. В. Лосев*, доктор экономических наук, доцент,  
*И. А. Павлов*, кандидат военных наук, доцент,  
*Е. Г. Семенова*, доктор технических наук, профессор,  
*М. Б. Сергеев*, доктор технических наук, профессор,  
*В. А. Фетисов*, доктор технических наук, профессор,  
*Н. А. Чернова*, кандидат экономических наук,  
*Л. И. Чубраева*, доктор технических наук, профессор,

Ответственный секретарь редакционной коллегии *В. Н. Соколова*

ISBN 978-5-8088-1122-5(Ч. I)  
ISBN 978-5-8088-1121-8

© Санкт-Петербургский государственный  
университет аэрокосмического  
приборостроения, 2016

# ТЕХНИЧЕСКИЕ НАУКИ

## Аэрокосмические приборы и системы

УДК 51-77

**А. Ю. Асенчик**

студент кафедры системного анализа и логистики

**Н. Н. Майоров**

доц., канд. техн. наук – научный руководитель

### ИССЛЕДОВАНИЕ ВЛИЯНИЯ ФУНКЦИИ ОТКАЗА НА ЛОГИСТИЧЕСКУЮ КРИВУЮ

Логистическая кривая – вид функции, используемой для выравнивания и прогнозирования временных рядов с насыщением (асимптотой), относится к S-образным кривым. По логистической кривой можно выравнивать ряды, отражающие три фазы развития:

- 1) формирование базы развития (медленный рост);
- 2) вторая – бурный рост;
- 3) третья – насыщение (медленный рост).

В частности логистическую кривую можно использовать для прогнозирования роста продаж той или иной продукции. Для данной задачи основным предположением является то, что скорость роста числа людей, знающих о продукции, пропорциональна как числу осведомленных в данный момент покупателей, так и числу неосведомленных, т.е.:

$$\frac{dx}{dt} = kx(N - x),$$

где  $x$  – число знающих о продукции людей,  $N$  – общее число потенциальных покупателей,  $k$  – положительный коэффициент пропорциональности.

Произведя некоторые математические преобразования, получим общий вид функции, описывающей процесс распространения информации о продукции для данной задачи:

$$x = \frac{N}{1 + (\alpha - 1)e^{-Nkt}},$$

где  $\alpha$  – некая функция отказа.

Содержательный смысл этой функции может быть обусловлен целым рядом причин: от несоответствия цены продукции ожиданиям покупателей до нежелания покупать рассматриваемый продукт по причине консервативности взглядов.

Формализована функция отказа может быть всеми теми же методами, которые используются при прогнозировании спроса:

- социологический метод, основанный на опросах конечных покупателей, выявлении их мнений и намерений;
- экспертный метод, основанный на подборе и формировании группы достаточно компетентных специалистов, которые высказывают свое мнение на основе знаний, опыта, интуиции, и оно рассматривается как экспертная оценка (метод Дельфи, метод мозгового штурма);
- статистические методы – методы моделирования (строится прогнозная модель, которая характеризует зависимость изучаемого параметра от ряда факторов), расчет коэффициента эла-

стичности спроса, экстраполяция (базируется на прошлом опыте, который пролонгируется на будущее).

Задачей данной работы является исследование полученной логистической функции (2) при различных функциях отказа.

$$N = 1, k = 0.5.$$

Рассмотрим следующие случаи отказа

$$\alpha = const$$

1. В зависимости от значения константы меняется характер роста осведомленности общества на первых стадиях, но рано или поздно при любом  $\alpha = const$  логистическая функция достигает уровня насыщения.

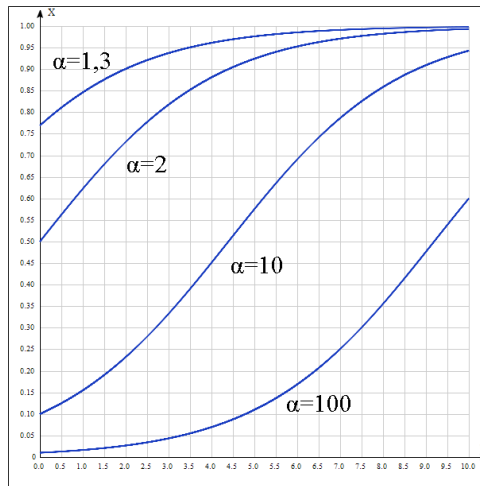


Рис. 1. Логистическая функция при  $\alpha=const$

$$\alpha = ax + b$$

2. Исходя из полученных графиков при  $b = 0$  и  $b = 5$  (рис. 2), для дальнейшего исследования можно сделать важный вывод: свободный член в любом виде функции отказа не может быть равен нулю. Иначе функция ведет себя неадекватно и уходит за пределы всего возможного потенциального числа покупателей. Таким образом, здесь и в дальнейшем к исследуемым функциям будем прибавлять свободный член  $b = 10$ . Выбор числового значения сделан произвольно для удобного вида получаемых графиков функций,  $b$  может быть равен как 10, так и 2, 100 и т.д.

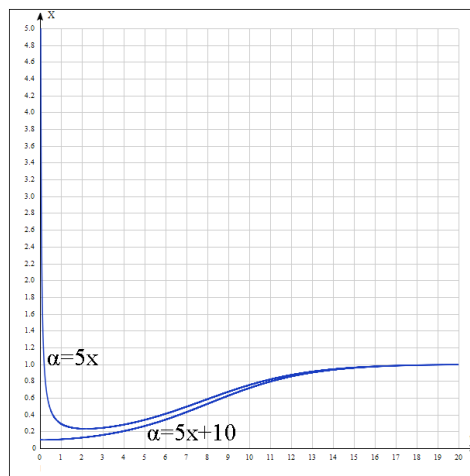


Рис. 2. Исследование аргумента  $b$  для линейной функции отказа



Далее рассмотрим влияние аргумента  $a$  на внешний вид логистической функции.

Вначале рассмотрим случай, когда  $a > 0$ . Чем  $a$  больше, тем более значительное падение функции происходит на начальных этапах (рис. 3). Для задачи прогноза это означает резкий спад числа продаж на ранних стадиях и большее время для стабилизации уровня продаж на рынке. В любом случае продажи рано или поздно достигают своего максимального значения, как и уровень осведомленности населения.

Значения функции  $a < 0$  можно рассматривать, только если они соответствуют еще и неравенству  $a > -1$ . При значениях  $a \leq -1$ , логистическая функция начинает вести себя неадекватно, уходя в бесконечность при некоторых значениях  $t$ .

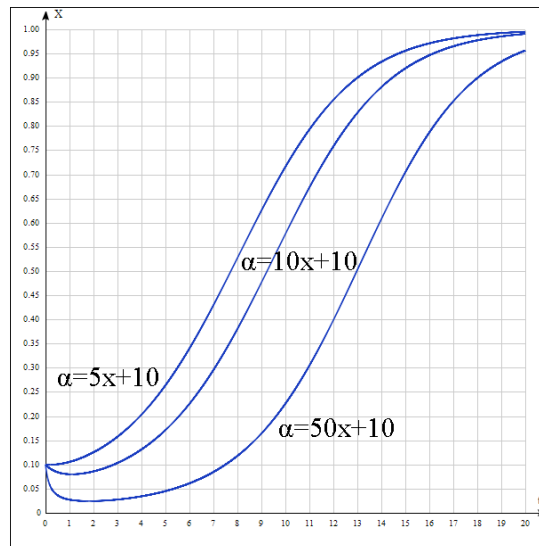


Рис. 3. Исследование аргумента  $a > 0$  для линейной функции отказа

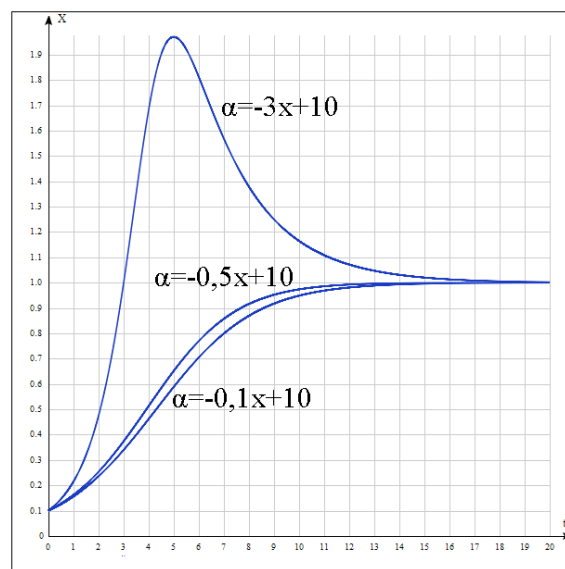


Рис. 4. Исследование аргумента  $a < 0$  для линейной функции отказа

$$\alpha = a \sin(cx + d) + b.$$

3. Здесь:

- $b$  характеризует начальный уровень продаж, а также резкость роста этого уровня;
- $a$  характеризует растяжение графика по оси  $0x$ . Чем больше увеличивается  $a$ , тем сильнее возрастает амплитуда колебаний;

–  $c$  характеризует растяжение графика по оси  $0t$ . При увеличении  $c$  частота колебаний повышается;

–  $d$  характеризует сдвиг графика по оси  $0t$ . При увеличении  $d$  график двигается в отрицательном направлении оси абсцисс.

Стоит отметить, что, как и было сказано ранее,  $b \neq 0$ . Для наглядности вышеприведенных замечаний для параметров данной тригонометрической модели построены графики с различными  $c$  (рис. 5).

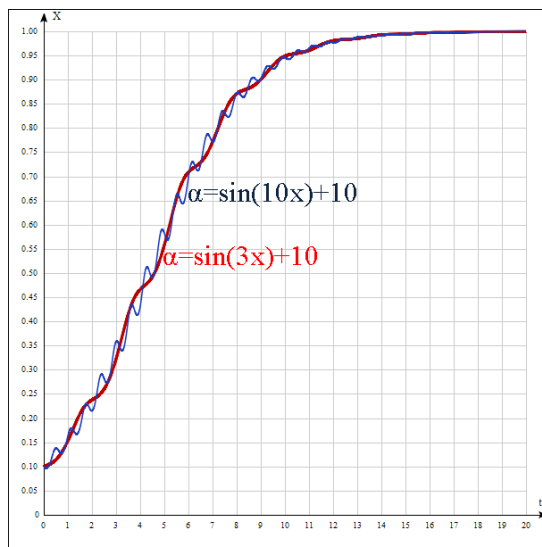


Рис. 5. Исследование параметра  $c$  для тригонометрической функции отказа

Синусоидальную функцию отказа в частности можно объяснить фактором сезонности, колебаниями спроса из-за колебаний курса валют и т.д.

Рассмотренные выше виды функции отказа являются частными случаями, обоснованными с точки зрения психологии покупателей. Другие виды функций могут быть рассмотрены только после детального рассмотрения причин того или иного поведения населения (степенная функция, экспоненциальная и пр.).

Таким образом, если у лица, принимающего решения, есть данные о показателях функции отказа (экспертные оценки, статистика и пр.), он может увидеть полную картину будущего для той или иной задачи (продажи, информированность населения и пр.). А эти полученные знания, в свою очередь, могут стать основанием для дальнейших действий руководителя.

УДК 681.51:621.452

**П. О. Борисов**

студент кафедры аэрокосмических измерительно-вычислительных комплексов

**В. Г. Никитин**

доц., канд. техн. наук – научный руководитель

### ИССЛЕДОВАНИЕ СИСТЕМЫ УПРАВЛЕНИЯ ГАЗОТУРБИНЫМ ДВИГАТЕЛЕМ С СЕЛЕКТОРОМ КАНАЛОВ

Исследована многосвязная система автоматического управления газотурбинным двигателем с селектором каналов, (рис. 1,2).

При автоматическом регулировании параметров газотурбинного двигателя (ГТД) возможно попадание системы в зону неустойчивой работы. Применение селектора каналов в много-

связных системах автоматического управления (САУ) позволяет исключить зону совместной работы регуляторов, т. е. на один регулирующий фактор воздействует только один регулятор, зависящий от режима работы ГТД.

В большинстве случаев в САУ с селектором каналов реализуется закон управления, при котором регулируется параметр двигателя, наиболее приблизившийся к значению, заданному программой регулирования. Возможны два вида селектирования: по минимуму (применяется селектор минимальных сигналов управления) и селектирование по максимуму (применяется селектор максимальных сигналов управления).

В работе рассматривается САУ двухвального ГТД с селектором минимальных сигналов управления.

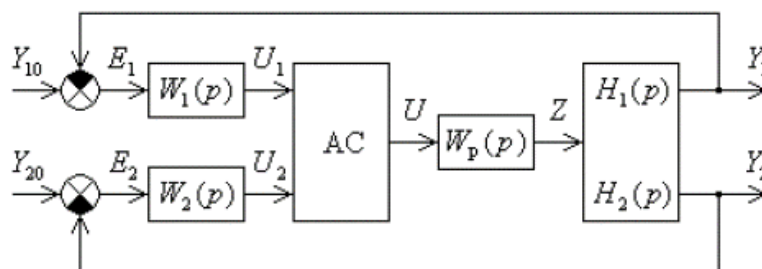


Рис. 1. Структурная схема САУ с селектором каналов

В данной схеме присутствуют следующие обозначения:  $Y_{10}$  – сигнал в канале частоты вращения;  $Y_{20}$  – сигнал в канале температуры газа;  $W_1$  – передаточная функция регулятора по каналу частоты вращения;  $W_2$  – передаточная функция регулятора по каналу температуры газа, АС – алгебраический селектор (АС);  $W_p$  – передаточная функция изодромного регулятора, общего для обоих каналов;  $H_1$  и  $H_2$  – передаточные функции ГТД в первом и во втором каналах.

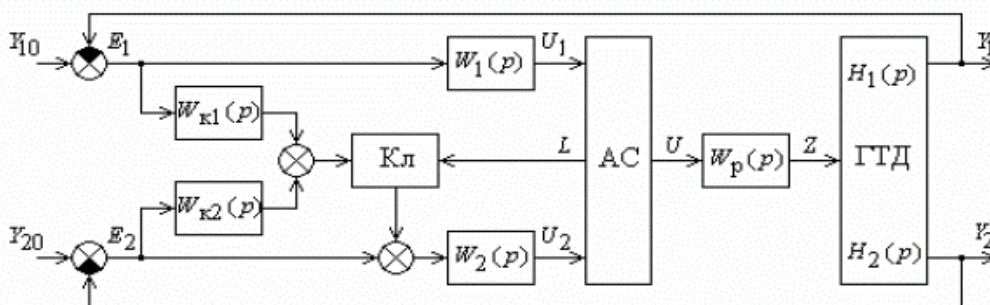


Рис. 2. Структурная схема САУ ГТД с регуляторами каналов до селектора

На данном рисунке присутствуют те же обозначения, что и на рисунке выше (см. рис. 1) со следующими дополнениями:  $W_{k1}$  и  $W_{k2}$  – передаточные функции корректирующих звеньев перекрестных связей, введенных для формирования условия селектирования, Кл – ключ, L – логический выход АС, U – сигнальный выход АС.

В качестве исходных данных были выбраны передаточные функции двухвального ГТД, имеющие следующий вид:

$$- \text{ по температуре газов после турбины [3]: } H_2 = 0,333 \frac{0,064p^2 + 0,667p + 1}{0,133p^2 + 0,94p + 1};$$

– по частоте вращения ротора [3]  $H_1 = 0,866 \frac{0,206p + 1}{0,133p^2 + 0,94p + 1}$ .

В данной работе также использованы следующие передаточные функции:

– передаточная функция регулятора по каналу частоты вращения

$$W_1(p) = 1,155 \frac{0,174p + 1}{0,206p + 1};$$

– передаточная функция регулятора по каналу температуры газа:

$$W_2(p) = 3 \frac{0,174p + 1}{0,064p^2 + 0,667p + 1};$$

– передаточная функция общего издромного регулятора:

$$W_{up}(p) = \frac{3(0,766p + 1)}{p(0,02p + 1)};$$

– передаточные функции корректирующих звеньев

$$W_{k1}(p) = \frac{(-0,0018p + 0,0517)(0,064p^2 + 0,667p + 1)}{(0,206p + 1)(0,174p + 1)},$$

$$W_{k2}(p) = 0,333 \frac{0,064p^2 + 0,145p - 2}{0,174p + 1}.$$

Схема, изображенная ранее (см. рис. 2) в пакете MathWorks– Simulink, имеет следующий вид (рис. 3).

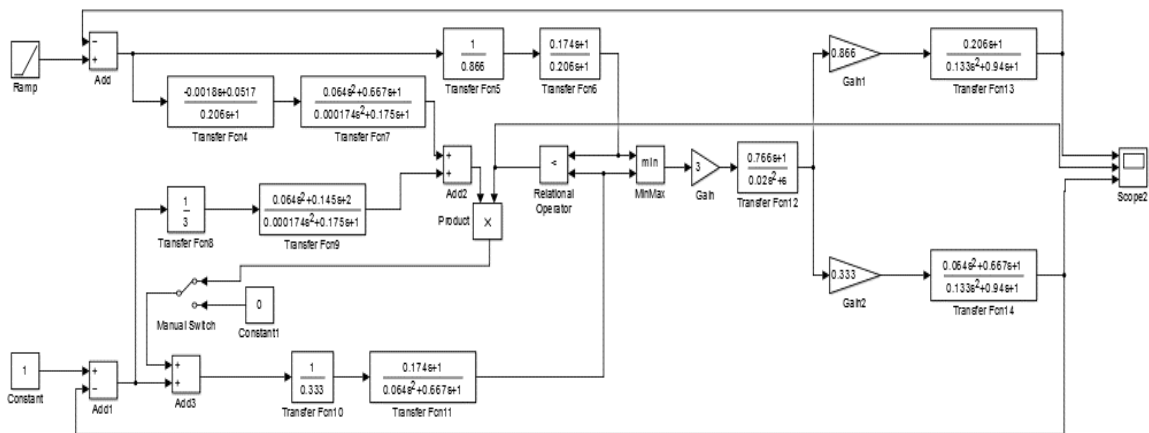


Рис. 3. Структурная схема САУ с селектором каналов, выполненная в Simulink

Моделирование работы данной схемы дает результат, представленный на рис. 4.

Здесь представлен график выходного сигнала с канала частоты вращения ротора, график сигнала переключения регулирующего канала и график выходного сигнала с канала температуры газа после турбины соответственно. На данных графиках отображается ситуация переключения управления с канала частоты вращения ротора на канал температуры газа после турбины. Как видно из 3-го графика, при переключении происходит заброс сигнала по каналу температуры, это обусловлено инерционностью терморпары. Для устранения перерегулирования по каналу температуры в САУ вводится схема компенсации сигнальных возмущений при переключении каналов [1]. На разомкнутый канал через блок согласования на переключатель каналов поступает сигнал  $\varepsilon = U_{\min} - U_{\max} < 0$ , что в свою очередь уменьшает задающее воздействие разомкнутого канала и тем самым корректирует момент переключения (рис. 5).

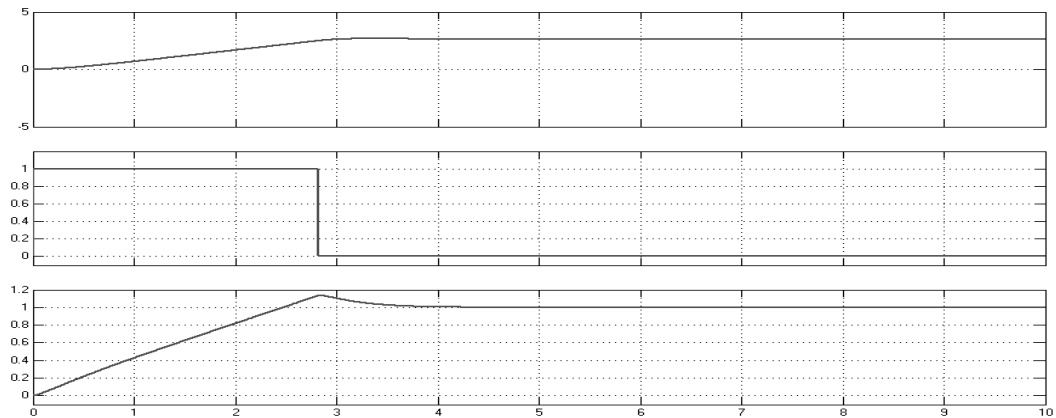


Рис. 4. Результат моделирования работы САУ с селектором каналов

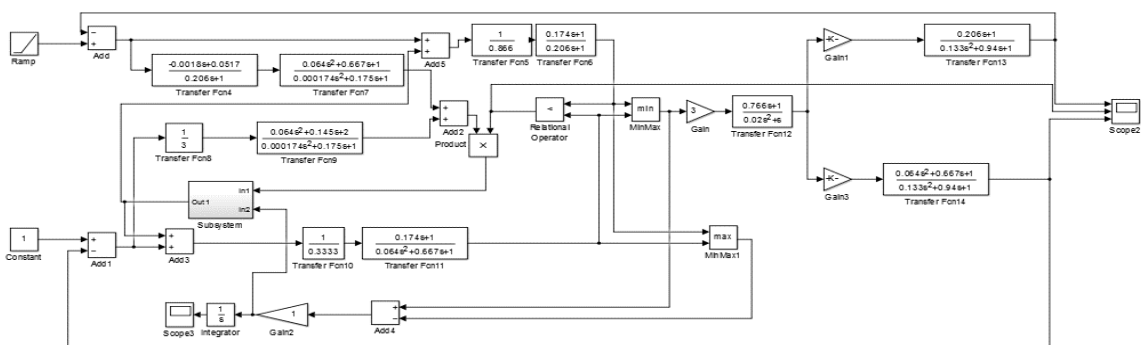


Рис. 5. Структурная схема САУ с компенсацией сигнальных возмущений, выполненная в Simulink.

Моделирование работы данной схемы дает результат, представленный на рис. 6

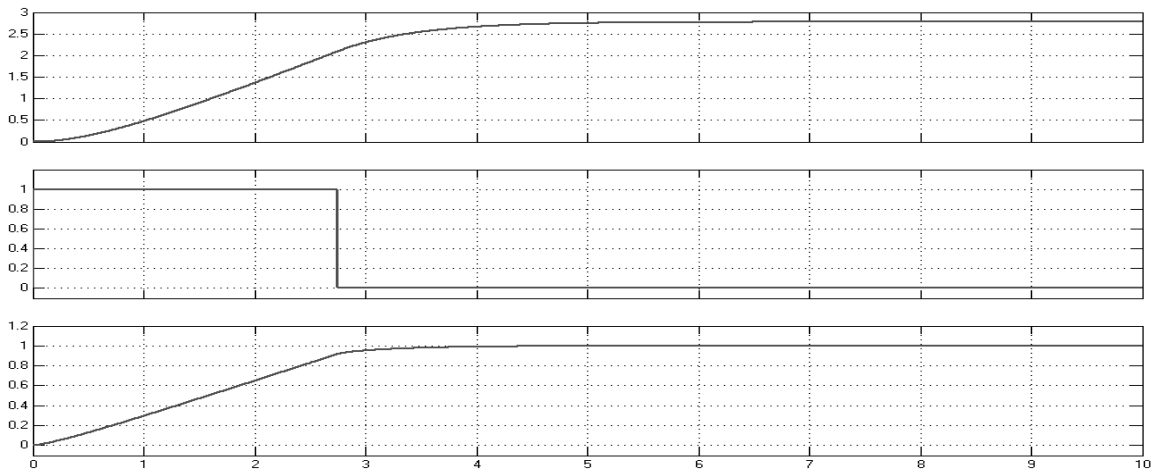


Рис. 6. Результат моделирования работы САУ с компенсацией сигнальных возмущений

По логарифм-частотным характеристикам (ЛЧХ)(рис. 7, 8) определяется запас устойчивости по фазе и по амплитуде.

Из ЛЧХ (см. рис. 7, 8) можно заметить, что запас устойчивости по фазе при внедрении селектора в САУ увеличивается примерно на 10%, а по амплитуде САУ без селектора является слабоустойчивой в отличие от САУ с селектором каналов.

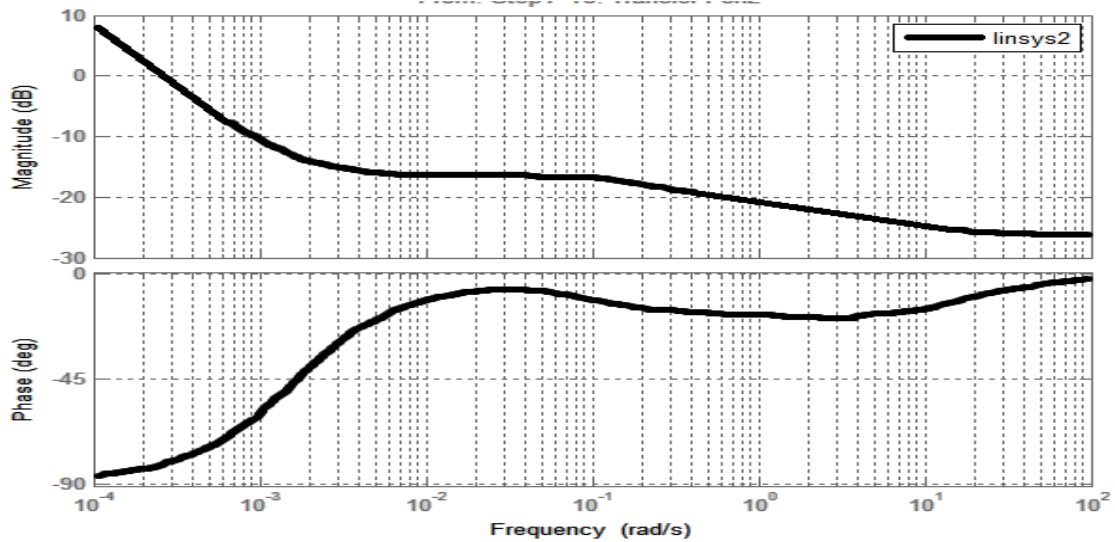


Рис. 7. ЛАХ и ЛФХ САУ ГТД без селектора

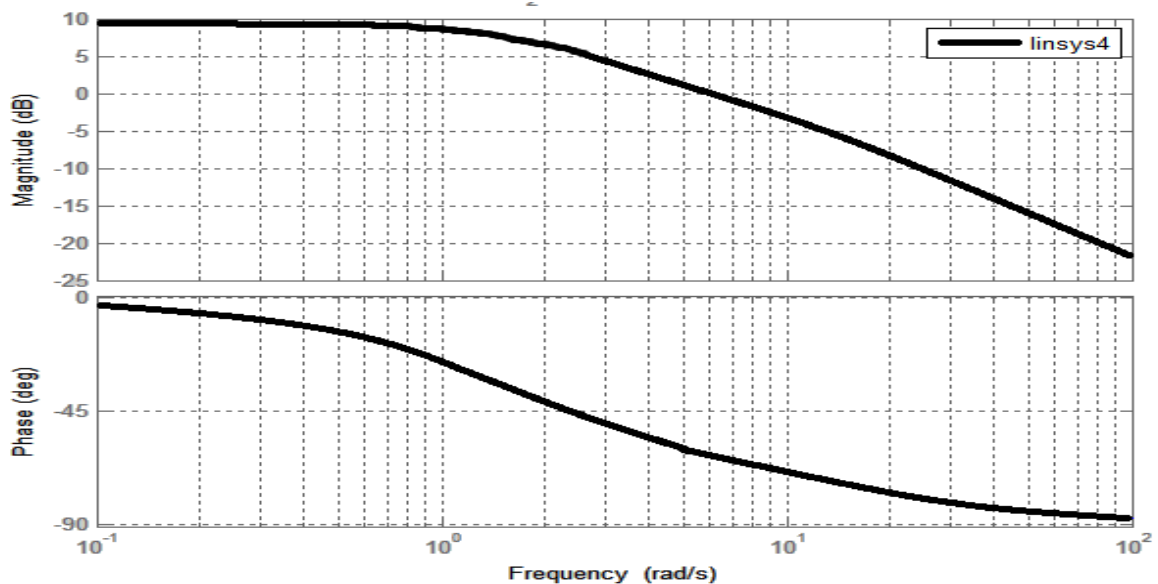


Рис. 8. ЛАХ и ЛФХ САУ ГТД с селектором каналов

По итогам проведенного исследования ясно, что внедрение в многосвязные САУ ГТД селектора каналов позволяет увеличить запас устойчивости системы, так как устраняется влияние на один регулирующий фактор нескольких регуляторов. Также в данной работе рассмотрен метод устранения перерегулирования по каналу температуры газов после турбины, вызванного инерционностью термомпары.

**Библиографический список**

1. Петунин В.И., Фрид А.И. Метод построения адаптивных логико-динамических систем автоматического управления с селекторами// Известия высших учебных заведений. Приборостроение. 2011. № 5. С. 49–56.
2. Петунин В.И. Синтез систем автоматического управления газотурбинными двигателями с селектором каналов// Вестник УГАТУ. 2008. № 1(28). С. 3–10.
3. Черкасов Б.А. Автоматика и регулирование воздушно-реактивных двигателей: учебник для вузов. М.: Машиностроение, 1988. 360 с.

УДК 629.735.33

**С. А. Булавин**

студент кафедры эксплуатации и управления аэрокосмическими системами

**М. Е. Тихомиров**

доц. кафедры эксплуатации и управления аэрокосмическими системами – научный руководитель

## СРАВНИТЕЛЬНЫЙ АНАЛИЗ РАДИОТЕХНИЧЕСКИХ СИСТЕМ ПОСАДКИ ВОЗДУШНЫХ СУДОВ

Цель исследования: Исследование перспективной системы посадки (СП) для дистанционно пилотируемых летательных аппаратов, разработка концепции одного из возможных вариантов технической реализации системы посадки малоразмерного дистанционно-пилотируемого летательного аппарата с учетом анализа современных систем посадки для воздушных судов (ВС).

Необходимость создания различных инструментальных систем обеспечения приземления ВС была связана с возросшими скоростями полета по глиссаде и связанными с этим метеорологическими ограничениями (дальностью горизонтальной и вертикальной видимости взлетно-посадочной полосы с борта ВС). Параметры видимости постепенно были сведены в посадочные метеоминимумы, принятые в СССР и за рубежом. Международная организация гражданской авиации ИКАО установила следующие посадочные категории (таблица 1).

Метеоминимумы устанавливаются для командира ВС (КВС), для ВС, для аэродрома, оснащенного соответствующими средствами обеспечения захода на посадку. Под вертикальной видимостью в таблице понимается высота нижней кромки облаков, высота принятия решения командиром ВС о прекращении захода на посадку и об уходе на «второй круг». Принцип установления данных норм связан с запретом посадки на данный аэродром, если хотя бы один из параметров допусков реальных метеоусловий на аэродроме, приведенных в таблице, для КВС, или ВС или аэродрома, хуже, чем указанный для соответствующей посадочной категории, приведенный в таблице.

Категория	Дальность горизонтальной видимости, м	Высота принятия решения, м
I	800	60
II	400	30
IIIА	200	15
IIIВ	50	0
IIIС	0	0

При автоматическом заходе на посадку КВС на высоте принятия решения обязан отключить автоматический режим и продолжать полет, непосредственно пилотируя самолет. При скорости захода на посадку 250 – 275 км/ч категория III после начала пробега по взлетно-посадочной полосе требует автоматического удерживания самолета на оси ВПП вплоть до полной остановки или отворота от ВПП.

Первые системы автоматической посадки ВС в СССР были разработаны к 1938 г. в Особом конструкторском бюро (г. Ленинград) и проходили испытания на Комендантском аэродроме Ленинграда. При этом на аэродроме был установлен всенаправленный радиомаяк, а на борту ВС типа ТБ-1 и ТБ-3 использовались автоматический радиоконпас (АРК) и ультразвуковой высотомер. Система называлась «Ночь-1».

В послевоенные годы для обеспечения приземления скоростных самолетов разрабатывались радиотехнические системы посадки различного диапазона длин волн. По мере совершенствования оптической и лазерной техники, а также систем технического зрения осуществляются попытки создания оптико-электронных систем посадки.

Классификация радиотехнических систем посадки радиотехнических систем посадки (РТСП), работающих на различных физических принципах и разных диапазонах длин радиоволн представлена на рис. 1.

Курсоглиссадные системы предусматривают размещение в непосредственной близости от ВПП курсового и глиссадного радиомаяков. Эти радиомаяки своими диаграммами направленности (точнее их равносигнальными зонами) задают направление к оси ВПП (курсовой радиомаяк – КРМ), а также траекторию снижения – глиссаду (глиссадный радиомаяк – ГРМ). При использовании таких систем ВС должно выполнить на так называемой «высоте круга» («круг» - это прямоугольный маршрут около аэродрома, заканчивающийся «четвертым разворотом» перед выходом на посадочный курс). В результате предпосадочного маневра ВС должно войти в зону действия сначала КРМ, а позднее – ГРМ. Для упрощения выполнения данного маневра по оси ВПП в зоне направления захода на посадку на различных удалениях от ВПП установлены дальний (в 4000 м от торца ВПП) и ближний (в 1000 м от торца ВПП) приводные радиостанции, включающие в себя всенаправленные радиомаяки дальний (ДПРМ) и ближний (БПРМ) – ориентиры для бортовых АРК, размещенных на ВС, и маркерные радиомаяки (МРМ). МРМ позволяют на борту ВС сигнализировать экипажу о факте пролета над данными радионавигационными точками. В такие моменты экипаж должен оценить положение ВС, как в горизонтальной плоскости по указателю АРК, так и в вертикальной плоскости (в случае равнинной местности над ДПРМ ВС должно пойти на высоте 200 м, а над БПРМ – на высоте 60 м). Так функционирует наземная аппаратура РТСП метрового диапазона длин (ILS) волн с дальностью действия 45–46 км [1].

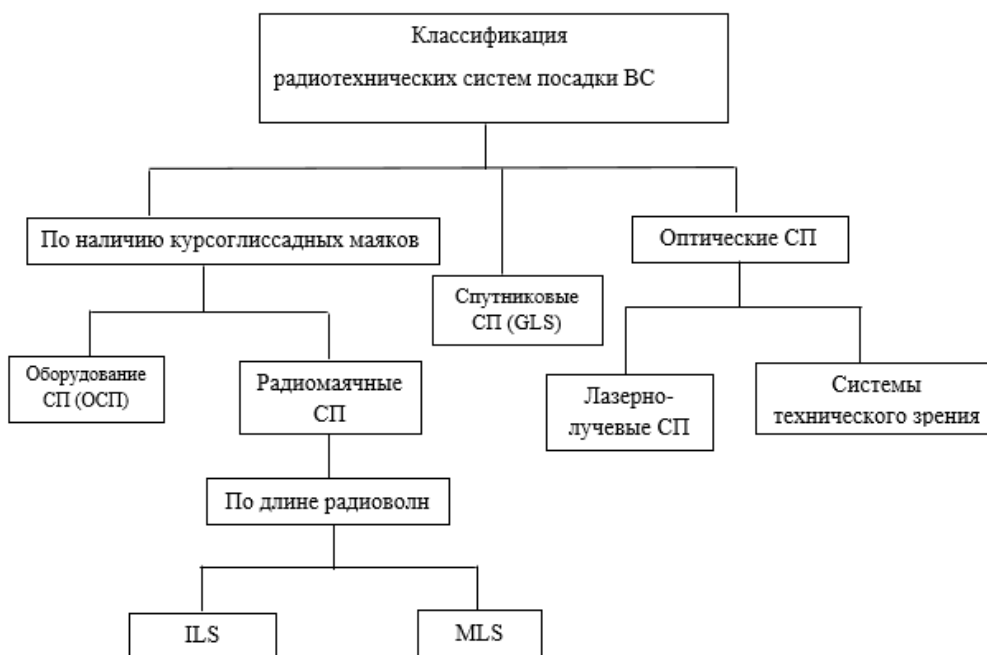


Рис. 1. Классификация РТСП, работающих на различных физических принципах и разных диапазонах длин радиоволн

Бортовая аппаратура ВС состоит из радиоприемников, настраиваемых на частоты диаграмм направленности КРМ и ГРМ, и формирует сигналы в виде напряжения постоянного тока, соответствующих отклонениям «правее - левее», «выше - ниже». Эти сигналы поступают на навигационно-посадочные индикаторы и в систему траекторного управления, являющуюся частью системы автоматического управления (САУ) или автоматической бортовой системы управления (АБСУ) для обеспечения директорного или автоматического захода ВС на посадку.

Первое поколение РТСП и его современные модификации имеют высокую стоимость, малую маневренность, длительный период настройки на новом аэродроме. Бортовая аппаратура



является аналоговой, с высоким потреблением электроэнергии и не может быть применена для посадки беспилотных летательных аппаратов.

В ряде стран получили развитие микроволновые системы посадки (MLS), имеющие дальность действия 37 км. Эти крайне дорогостоящие радиомаячные системы обладают рядом достоинств, могут обеспечить приземление самолета при категории IIIВ. На борту самолета применяется лишь один приемник для двух азимутальных и двух угломестных радиомаяков и радиодальномер. Однако вследствие высокой стоимости и сравнительно малой вероятности усложнения метеоусловий до уровня IIIВ несколько авиационных администраций посчитали экономически нецелесообразным внедрение MLS. Микроволновая система посадки обеспечила автоматическую посадку орбитального корабля 11Ф35 «Буран» 15 ноября 1988 г. При этом была успешно продемонстрирована возможность автоматической посадки тяжелого беспилотного ВС на ВПП в сложных метеоусловиях [2].

Широкие возможности открыло применение РТСП с использованием систем спутниковой навигации. GLS – спутниковая система захода на посадку, которая в настоящее время активно внедряется во всем мире.

Основная цель такого оборудования – повысить точность определения местоположения воздушного судна в пространстве и избежать ошибок при всех возможных внешних воздействиях на сигнал со спутников, который принимает воздушное судно, в том числе и во время выполнения точного захода на посадку.

Принцип действия системы простой: местоположение самолета определяется по спутникам ГЛОНАСС и GPS. Для обеспечения точного захода вводятся наземные корректирующие станции ЛККС (локальная контрольно-корректирующая станция), передающие дополнительный сигнал. Поскольку они, в отличие от спутников, неподвижны и при этом находятся значительно ближе, точность определения координат значительно возрастает и погрешность не превышает 5 м [4].

Использование GLS имеет ряд преимуществ по сравнению с системой РТСП метровых волн (ILS) – основным на сегодня способом точного захода на посадку по приборам. Так, одна ЛККС может обслуживать сразу несколько полос и направлений, в то время как для ILS требуется по два радиомаяка (КРМ и ГРМ) возле каждого из торцов каждой ВПП. Кроме того, ЛККС не так требовательна к месту размещения. Поэтому с ее помощью можно обеспечить точным заходом даже те ВПП, где невозможно установить ILS. Также ЛККС требует менее частых проверок и обслуживания и меньше зависит от влияния помех, ведение по глиссаде осуществляется более плавно. Данная установка имеет малые габариты, что является большим преимуществом над крупногабаритной ILS.

Лазерно-лучевая система посадки (ЛЛ СП) является одной из перспективных систем посадки. Она устойчива к любым погодным условиям и способна обеспечить посадку даже при сильной облачности. Светящиеся нити лазерных лучей образуют посадочный коридор. Комбинация лучей курсоглиссадной группы лазерных маяков воспринимается в виде символа, однозначно определяющего положение самолета относительно расчетной траектории. Курсовой лазер устанавливается на продолжении оси ВПП. Луч, направленный под углом к горизонту, лежит в плоскости курса и указывает летчику направление посадки. Глиссадные лазерные маяки располагаются по боковым сторонам ВПП, лучи лежат в плоскости глиссады. Выполняя заход по расчетной траектории и находясь над курсовым лучом, летчик воспринимает его проекцию в виде вертикальной прямой, а снижаясь в посадочном коридоре между лучами глиссадных маяков видит их проекции в виде горизонтальных линий. Если точно выдерживать посадочную траекторию, лучи курсоглиссадной группы формируются в виде символа, форма которого напоминает букву Т.

В случае запрета посадки все лучи курсоглиссадной группы переводятся в прерывистый режим излучения, экипаж переводит самолет в режим ухода «на второй круг».

Для увеличения мощности излучения и резервирования системы на случай отказа каждый лазерный маяк состоит из двух или четырех оптических квантовых генераторов, надежно

защищенных от любых атмосферных воздействий. В систему включен передвижной пункт дистанционного управления и контроля за работой лазеров. Данная система нашла свое применение как на гражданских, так и на военных аэродромах [3].

Не менее перспективными разработками в авиации являются беспилотные летательные аппараты (БЛА), которые в свою очередь используются в разведывательных целях. Полет данных БЛА осуществляется с помощью оператора, который составляет маршрут и при необходимости корректирует его. Команды управления передаются на БЛА в онлайн-режиме, данную возможность обеспечивает канал связи. На борту БЛА установлены система связи, навигационные системы, приемник, передатчик и система вычисления. Как правило, данный вид ЛА осуществляет посадку при помощи парашюта. .

Рассмотрим предлагаемую оптическую систему технического зрения (рис. 2), предназначенную для обеспечения автоматической посадки и взлета беспилотного ЛА (БЛА). Оптическая система обеспечения автоматической посадки БЛА состоит из наземного и бортового модулей.

Основой наземного модуля является лазерный сканер (лидар), установленный на специальном основании. Наземный лазерный сканер (лидар), сканируя в заданном секторе захода БЛА на посадку, формирует облако точек, описывающих положение БЛА относительно требуемых координат захода на посадку. Таким образом, лидар формирует трехмерное изображение в виде облака точек. Параметры положения БЛА, получаемые на экране лидара, получаются путем обработки изображения в ЭВМ.

Для обеспечения посадки БЛА с использованием системы на основе лазерного локатора на борту БЛА размещается рефлектор, представляющий собой призматический отражатель для лазерного луча. Для обеспечения устойчивости процесса визирования призматического отражателя, наблюдаемого лидаром при различных погодных условиях, особенно при большой влажности и низких температурах воздуха, узел призматического отражателя оснащается датчиком обледенения и нагревательным элементом. Лазерный сканер (лидар) передает данные азимута объекта, угол места обнаруженного объекта, признак калибровки, количество отраженных лучей, состояние лазера на борт БЛА.

Измеренные таким образом параметры отклонения БЛА от требуемой траектории посылаются как в автоматизированное рабочее место (АРМ) оператора на наземном пункте управления, так и транслируются по радиотелеметрическому каналу на борт БЛА. На борту БЛА в режиме автоматического управления система автоматического управления (САУ) формирует команды на отклонение аэродинамических рулевых поверхностей с целью устранения возникших в процессе захода на посадку отклонений от заданной траектории по курсу и по глиссаде.

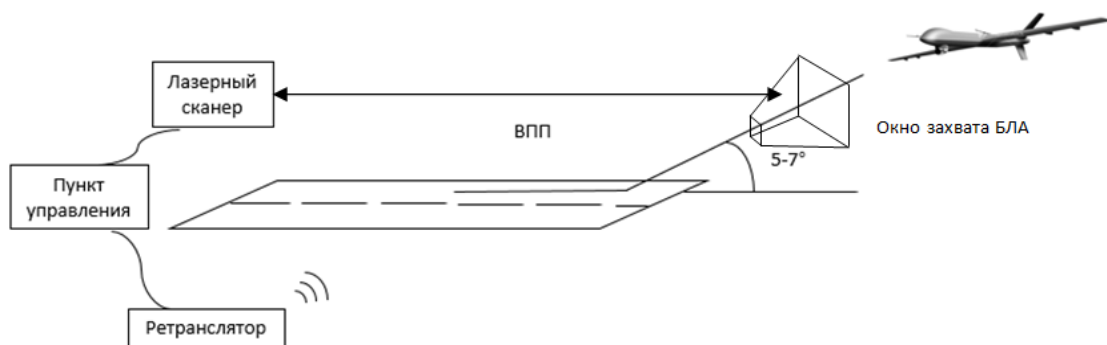


Рис. 2. Схема захода на посадку с использованием системы технического зрения

Применение призматического отражателя лазерного луча может существенно повысить точность определения координат положения БЛА при заходе на посадку. Одновременно будут существенно снижены требования к характеристикам средств вычислительной техники, приме-

няемой в процессе обеспечения посадки. Важным достоинством предлагаемой системы будет возможность пробега при посадке или разбега БЛА при взлете в автоматическом режиме.

#### Заключение

Произведенный анализ технической литературы позволяет сделать следующие выводы:

MLS менее чувствительна к местным условиям, способна обслуживать кратное прибытие ВС и может задавать переменные схемы захода на посадку.

Частотный диапазон ILS подвергался интерференции на концах диапазона от мощных коммерческих радиовещательных станций и других источников, что устранено в MLS.

Наиболее точной системой посадки, применяемой в настоящее время, является GLS. Совместная работа систем ILS и GLS обеспечивает еще более высокую точность, а также надежность на случай отказа одной из систем во время захода на посадку. Кроме того, данная система может обеспечить заход воздушных судов по свободным траекториям, сокращая интервалы продольного эшелонирования и повышая пропускную способность данного направления захода.

Точность навигационного обслуживания в GLS не зависит от удаления ВС от ВПП и от станции в пределах установленной зоны для процедур посадки, в то время как в случае применения ILS точность навигационного обслуживания существенно зависит от удаления ВС от ВПП.

Применение GLS не требует размещения ЛККС на осевой линии ВПП или вблизи ВПП, что исключает влияние впереди идущего ВС на прием посадочных данных сзади идущим ВС и снижает требования к пространственному разделению ВС при посадке по сравнению с ILS.

Применение лидара и системы технического зрения для обеспечения захода на посадку имеет ряд ограничений, в том числе по дальности горизонтальной и вертикальной видимости в атмосфере, а также по высоким требованиям к средствам вычислительной техники, используемым в контуре автоматической посадки БЛА.

#### Библиографический список

1. *Верещака А.И., Олянюк П.В.* Авиационное радиооборудование. Москва: Транспорт, 1996. 342с.
2. *Сосновский А.А., Хаймович И.А.* Авиационная радионавигация. Москва: Транспорт, 1980. 255 с.
3. Гончаров И.Н., Дежин В.Н., Кутахов В.П., Петухов А.В., Сидорин В.М., Сухарь И.М. Лазеры в авиации. Воениздат, 1982. 158с.
4. Интернет-ресурс: [http://www.frequentflyers.ru/2015/01/20/gls\\_landing/](http://www.frequentflyers.ru/2015/01/20/gls_landing/)

#### УДК 629.7.018.3

##### **А. Н. Булухова**

магистрант кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

##### **А. П. Григорьев**

научный руководитель

##### **В. Я. Мамаев**

канд. техн. наук, доц. – научный консультант

### **МЕТОДИКО-ДИДАКТИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ РАБОТЫ С ДИСТАНЦИОННЫМИ АДАПТИВНЫМИ ОБУЧАЮЩИМИ СИСТЕМАМИ**

Процесс обучения в обучающих системах наиболее часто рассматривается как контролируемый процесс решения адаптивных тестовых заданий [1], как правило, с последующей выдачей комментариев (диагностика) и восстановлением знаний (компенсация неполученных, недо-

полученных или же неувоенных знаний), посредством обращения к теоретическому материалу учебного пособия (преимущественно электронному).

Л. А. Растринин сформулировал цель обучения в виде стандартной задачи, состоящей в том, чтобы обучаемый наилучшим образом запомнил определенные порции информации. Алгоритм обучения в данном подходе представляет собой некоторое правило выбора порции обучающей информации. Эффективность такого обучения оценивается по результатам периодического контроля обучаемого (тестирование). Очевидно, что [1] задача обучения связана с решением задачи адаптации алгоритма обучения и модели пользователя с целью максимизации эффективности процесса обучения на всех его стадиях.

Отметим достоинства данного подхода: строгая математическая формализация задачи, наличие модели обучаемого в виде объекта управления (сложная система с присущими ей характерными особенностями). Тем не менее, анализируя данный подход, Л. В. Зайцева [2], обратила внимание на отсутствие конкретизации отдельных этапов контроля знаний (входной, промежуточный, рубежный, итоговый), что является существенным недостатком, так как на каждом из этапов перед контролем знаний (адаптивный тест) ставятся независимые целевые задачи.

Авторами данной статьи в дополнении выявляется ещё ряд недостатков такого подхода: отсутствие описания предметной области, моделей и алгоритмов адаптивного контроля, наличие одноконтурной системы восстановления знаний (наличие одной обратной связи с теоретическим материалом).

Таким образом, в связи с вышесказанным возникает необходимость модернизации и/или модификации баз адаптивных тестовых заданий и классических алгоритмов контроля выполнения заданий.

Характерные особенности классических адаптивных тестовых заданий подробно рассматривались в работе [3], а внешний вид наиболее часто используемых тестовых заданий представлен на рисунке, где приведены примеры неадаптивных и адаптивных моделей тестирования.



Отметим достоинства классических моделей тестирования знаний: простота реализации; потенциальная возможность распараллеливания работы над тестовыми заданиями.

Анализируя недостатки, заметим лишь наиболее существенные: потенциально низкие адаптационные возможности, низкий уровень дифференциации и индивидуализации при организации стратегии обучения, расчет оценки производится в «серых» баллах (вес каждого из вопросов одинаков по отношению к общему числу вопросов); игнорирование аспектов, связанных с вероятностью угадывания и/или списывания; не учитывается скорость выполнения задания.

Таким образом, отметив ряд серьезных недостатков классических адаптивных тестовых заданий, оперируя данными статей [1–3], целесообразно сформулировать и рассмотреть потенциальную возможность реализации «Сценарного метода» контроля знаний, ориентированного на использование в дистанционных адаптивных обучающих системах в качестве независимого модуля контроля знаний.

Авторами статьи и разработчиками сайта [http://cpmssmedia.dlinkddns.com/other\\_web/test2](http://cpmssmedia.dlinkddns.com/other_web/test2) для организации дистанционного адаптивного контроля знаний используется «Сценарный метод» тестирования, являющийся более эффективным по сравнению с классическими методами адаптивного и неадаптивного контроля. Суть данного метода рассмотрена в работе [4].

Основной задачей сценария является формирование адаптивного алгоритма, направленного на построение индивидуальной последовательности заданий, генерируемых из базы адаптивных тестовых заданий, направленных на выполнение целевых задач текущего этапа контроля знаний. Основные этапы теоретического контроля знаний и соответствующие им требования к сценариям представлены в [4]. Таким образом, для организации эффективного контроля знаний посредством «Сценарного метода» необходимо и достаточно иметь сценарии входного, промежуточного, рубежного и итогового тестирования [4].

Характерные особенности и описания всех сценариев приведены на сайте [http://cpmssmedia.dlinkddns.com/other\\_web/test2/index.php?route=прос](http://cpmssmedia.dlinkddns.com/other_web/test2/index.php?route=прос), а также представлены в таблице.

Сценарии контроля знаний на основных этапах обучающего процесса

Этапы контроля знаний (вид контроля)	Временная характеристика этапа контроля знаний	Цели этапа контроля знаний	Сценарий контроля знаний, рекомендуемый авторами статьи
Входной	Начало теоретического обучения	Оценка уровня знаний в начале обучения	«Пизанская башня», «Сизифов камень»
Промежуточный	Недетерминированный момент времени (по субъективному усмотрению преподавателя-эксперта)	Оценка усвоения знаний в течение обучения	«ГАШИГ», «Путеводная нить-Ариадны»
Рубежный (Тематический)	Детерминированный момент времени (окончание обучающего модуля, завершение текущей темы, раздела, главы, параграфа ит. д.).	Оценка усвоения знаний, после изучения темы, раздела и т. д.	«Неоклассицизм», «Маятник»
Итоговый (предэкзаменационный и экзаменационный)	Окончание теоретического обучения	Применяется для систематизации пройденного материала в конце обучения	«Маятник», «Загадка сфинкса», «Ахиллесова пята»

Авторами статьи выделяется ряд сценариев[4], направленных на повышение интенсификации и эффективности обучения, наиболее актуальных в плане современных требований федеральных государственных образовательных стандартов (ФГОС).

Сценарии психофизиологической диагностики позволяют оценить и заранее предусмотреть на основе определения особенностей учащихся, в каких учебных ситуациях, условиях эти проявления окажутся благоприятными для достижения успеха, а в каких могут препятствовать ему.

Задача преподавателя состоит в том, чтобы помочь учащемуся найти наиболее подходящий для него индивидуальный стиль деятельности. Таким образом, можно сделать вывод, что обучение можно описать, проанализировать, систематизировать и запрограммировать, тем самым повышая общую эффективность занятий, учитывая индивидуальные особенности каждого студента и оптимизируя для него обучающий процесс.

Резюмируя все вышесказанное, отметим, что использование «Сценарного метода» [4] в контроле знаний, а также при построении обучающей системы различного типа позволяет на потенциально более высоком уровне обеспечить методическое и дидактическое сопровождение с одной стороны, а также предоставить преподавателю более широкие возможности по вариации отдельных этапов обучающего процесса.

В связи с характерными особенностями ФГОС третьего и четвертого поколения к снижению аудиторных часов теоретической подготовки, наиболее перспективными сценариями контроля знаний в настоящее и в ближайшее время, по мнению авторов статьи, будут: мотивационные и сценарии быстрого контроля с учетом психофизиологической диагностики [5, 6]. Также стоит отметить, что при условии объединения данных сценариев, учитывая требования эффективности (успеваемость, достоверность, адекватность, работоспособность), потенциально возможно еще большее повышение интенсификации обучающего процесса, причем без потери качества, что, безусловно, является задачей первостепенной.

В заключении необходимо добавить, что задача тестового контроля по своей природе двойка, с технической точки зрения тестовый контроль является гибкой обратной связью между субъектом и объектом обучения. Рассматривая обучение как процесс управления сложным объектом, профессор Л. А. Растрин установил, что цели субъекта (преподаватель) и объекта (обучаемый) управления не совпадают, более того, объект, как правило, враждебен по отношению к сигналам управления субъекта [1]. Тест с точки зрения преподавателя должен освобождать его от монотонной трудоемкой процедуры обработки материалов и оценивания, помогать, проверять (но не опрашивать) каждого. С точки зрения объекта тест должен быть интересен, «прозрачен» в плане оценки и интерпретируем в реальном масштабе времени, т. е. результаты должны быть выданы обучаемому незамедлительно, с соответствующими пояснениями и уточнениями.

Помимо этого, стоит особо отметить и акцентировать на этом пристальное внимание: «тестирование – это эффективная процедура аудиторной и/или внеаудиторной работы и самодиагностики, которая особенно важна в тот период, когда обучаемый не может установить точно и самостоятельно уровень собственной подготовки на том или ином этапе обучения». Очевидно, что при общей тенденции ФГОС к сокращению аудиторных часов и повышению значимости самостоятельной работы, на первое место выступают вопросы мотивации, что достигается за счет получения результатов работы обучаемыми в реальном масштабе времени, за счет самоорганизации и активизации, а также познавательной деятельности, которая должна начинаться в рамках аудиторной работы и продолжаться в рамках внеаудиторной.

То время, что преподаватель экономит, минуя процедуру проверки (возлагая ее на компьютерный тест), может быть направлено на творческую работу, иными словами преподаватель не тратит «львиную долю» времени на проверку качества усвоения теоретического материала, а может обсудить устно тонкие нюансы и/или проблемные аспекты читаемой дисциплины, проводя занятие на интеллектуально более высоком уровне с уже подготовленной (в рамках самостоятельной домашней работы и самодиагностики) аудиторией.

Элементы тестирования способны интенсифицировать процесс обучения за счет интеграции в такие «приемы», как опережающая самостоятельная работа. Необходимо понимать, что в должной степени тестирование не раскрывает творческий потенциал обучаемого. Задача тестирования сводится к проверке знаний (контроль, диагностика, восстановление), а не умений, не навыков и не компетенций (для этого в курсе той или иной дисциплины присутствуют практические и лабораторные занятия, семинары и пр. методико-дидактические элементы).

На базе тестов, преимущественно адаптивных, с использованием игровых технологий не трудно реализовать элементы обучающего тестирования с эффективными мотивационными алгоритмами. Таким образом, можно сделать вывод, что тестовый контроль – не является панацеей, тем не менее это эффективный инструмент в руках умелого преподавателя, который точно знает, как именно, для каких целей и на каких этапах обучения его использовать.

### Библиографический список

1. *Растрюгин Л. А.* Обучение как управление сложной системой. Техническая кибернетика, №2, 1993.
2. *Зайцева Л. В.* Модели и методы адаптации к учащимся в системах компьютерного обучения. EducationalTechnology&Society. 6(4) 2003.
3. *Глова В. И., Дуплик С. В.* Модели педагогического тестирования обучаемых // Вестник Казан, гос. техн. ун-та им. А. Н. Туполева. 2003. № 2. С. 74–79.
4. *Григорьев А. П.* Сценарный метод контроля навигационных знаний при проектировании обучающих систем. Научная сессия ГУАП: сб. докл. в 3 ч. Ч. I. Технические науки / СПб.: ГУАП, СПб., 2015. С 44–48.
5. *Григорьев А. П., Чернелевский А. О.* Разработка сценария мотивационного тестирования для системы дистанционного адаптивного контроля знаний. Педагогическая наука и современное образование. Сборник статей II Международной научно-практической конференции, посвященной Дню российской науки. Санкт-Петербург. Издательство РГПУ им. А. И. Герцена, 2015. С. 254–257.
6. *Григорьев А. П., Аргатюк А. А.* Быстрое адаптивное дистанционное тестирование знаний закрытого типа. Педагогическая наука и современное образование: сб. статей II Международной научно-практической конференции, посвященной дню российской науки. Санкт-Петербург: РГПУ им. А. И. Герцена, 2015. С. 229–232.

УДК 004.451.44

**В. В. Виноградов**

студент кафедры аэрокосмических компьютерных и программных систем

**И. Р. Федоров**

студент кафедры аэрокосмических компьютерных и программных систем

**С. В. Осмоловский**

аспирант – научный руководитель

## МЕТОДЫ ПЛАНИРОВАНИЯ ЗАДАЧ В СИСТЕМАХ РЕАЛЬНОГО ВРЕМЕНИ НА МНОГОЯДЕРНЫХ ПРОЦЕССОРАХ

Введение

В наши дни микропроцессорная отрасль бурно развивается, благодаря чему многопроцессорные (multiprocessor) и многоядерные (multicore) платформы становятся привлекательным решением для удовлетворения растущих требований по производительности вычислительных систем. К многоядерным архитектурам проявляется значительный интерес, так как проблемы тепловыделения и энергопотребления ограничивают дальнейшее увеличение производительности одноядерных процессоров. Тенденция к переходу встраиваемых систем (embedded systems) и систем смешанной критичности (mixed-criticality systems) к использованию многоядерных аппаратных платформ позволяет повысить производительность такой системы, а также реализовать больше функциональных возможностей в рамках единой универсальной вычислительной системы [1]. В настоящее время подобные системы становятся неотъемлемой частью многих современных комплексов, требующих гарантий реального времени (real-time) и обладающих повышенными требованиями к безопасности (safety-critical). В связи с этим наблюдается возрастающий интерес к теории планирования задач реального времени на многоядерных платформах и к практической реализации полученных методов и подходов [2].

Основной целью данной статьи является проведение сравнительного анализа существующих методов планирования задач в системах реального времени на многоядерных процессорах, определение их ограничений и возможных областей применения.

Планирование реального времени.

В системах реального времени (СРВ, real-time systems) временная предсказуемость (predictability), выраженная в форме гарантий того, что каждая задача будет иметь требуемое время отклика (response time), которое уложится в строгие критические сроки (дедлайн, deadline), столь же важна, как производительность (то, как быстро каждая задача будет завершена) или пропускная способность (количество задач, которое может быть завершено за некий заданный период времени). Переход от выполнения одной задачи к другой осуществляется в результате планирования (scheduling).

В научно-исследовательских работах, посвящённых СРВ, большое внимание уделяется алгоритмам планирования – набору правил, согласно которому производится переключение задач и выделение им требуемых ресурсов. Эти алгоритмы базируются на модели задач реального времени, которые содержат важную информацию о временных характеристиках и параметрах задач в СРВ [3].

Алгоритмы планирования должны предоставлять априорные доказательства того, что все задачи будут успевать завершать свое выполнение относительно дедлайнов при условии, что все задачи удовлетворяют свойствам и характеристикам, определенным в модели задач. Эти гарантии должны быть аналитически доказаны до начала фактического функционирования СРВ (в режиме офлайн).

Можно выделить два типа алгоритмов планирования: политики планирования (scheduling policies) и тесты планируемости (schedulability tests). Политики планирования, которые иногда называют планировщиками (schedulers), контролируют расписание задач. Они работают в системе одновременно с задачами и в режиме онлайн принимают решения по планированию, основываясь на характеристиках и ограничениях задач. Тесты планируемости до запуска системы производят проверку соблюдения гарантий выполнения задач к заданным критическим срокам.

С точки зрения теории планирования реального времени планирование называется выполнимым (feasible), если все задачи могут быть завершены в соответствии с набором указанных ограничений. Набор задач называют планируемым (schedulable), если существует хотя бы один алгоритм, который может производить выполнимое планирование.

Модель задач реального времени

В большом количестве тематических научно-исследовательских работ СРВ описываются набором независимых периодических/спорадических задач, которые должны быть планируемы на одном или нескольких процессорах в соответствии с их временными параметрами [4]. Экземпляр задачи называется работой (job). Задача может порождать бесконечное количество работ. Задача имеет следующие временные параметры:

- Дедлайн  $D_j$  : – критический срок, к которому задача должна успеть выполняться.
- Период  $T_j$  : – минимальный временной интервал между активизациями экземпляров задачи – моментами порождения работ (release time, arrival time).
- Наихудшее время исполнения задачи (worst-case execution time)  $C_j$  : – наихудшее значение времени исполнения задачи без ее приостановок и прерываний.

Простым способом проведения проверки планируемости  $n$  периодических задач с неявным дедлайном ( $T_j = D_j$ ) на процессоре является расчет коэффициента использования процессора (processor utilization factor)  $U$  :

$$U = \sum_{i=1}^n \frac{C_i}{T_i}. (1)$$



Если  $U > 1$ , то задачи не могут быть успешно планируемы на данном процессоре.

Виды многопроцессорных систем

В многопроцессорных вычислительных платформах (многоядерных процессорах) для выполнения задач доступно несколько процессоров (процессорных ядер). В теории планирования СРВ на многоядерных процессорах различают по крайней мере три вида многопроцессорности [5]:

- Идентичная (identical) – процессоры полностью идентичны, обладают одинаковыми функциональными возможностями и частотой.
- Однородная (uniform) – процессоры обладают одинаковыми функциональными возможностями, но разной частотой.
- Неоднородная (unrelated) – процессоры имеют разные функциональные возможности и частоту. Задачи могут быть не в состоянии выполняться на всех процессорах.

Типы многопроцессорного планирования

Существующие на данный момент способы планирования реального времени на многоядерных процессорах можно разделить на три категории: распределенное (partitioned, non-migrating), совместное (global, full migrating) и гибридное (hybrid, restricted migrating) планирования.

При распределенном многопроцессорном планировании набор задач размещается в нескольких непересекающихся множествах и для каждого множества назначается свой процессор (процессорное ядро). На каждом процессоре присутствует планировщик с собственной очередью задач, готовых к исполнению, и задачи/работы не могут перемещаться («мигрировать») на другие процессоры. Таким образом, когда распределение задач между процессорами выполнено, данный тип многопроцессорного планирования позволяет применять существующие алгоритмы планирования и тесты планируемости реального времени, созданные для однопроцессорных систем. Это является весомым преимуществом данного способа планирования. В данном типе планирования отсутствуют накладные расходы, связанные с миграцией задач между процессорами. Накладные расходы, связанные с управлением очередью задач, при распределенном планировании меньше, чем при совместном за счёт наличия на каждом процессоре планировщика с отдельной очередью задач в отличие от общей очереди при совместном планировании. Еще одним преимуществом распределённого планирования является отсутствие в нем так называемого «эффекта Дала» – аномалии планирования, сформулированной Dhall и Liu [6]. Распределенное планирование проиллюстрировано на рис. 1, где  $\tau$ -задача.

Тем не менее распределенное планирование по сравнению с совместным имеет важный недостаток: расходование процессорных ресурсов далеко не всегда происходит эффективно. Может возникнуть ситуация, когда внутри многопроцессорной системы некоторые процессоры простаивают без выполнения полезной работы – исполнения задач, а некоторые, наоборот, не могут исполнять задачу, так как для нее не хватает вычислительных мощностей данного процессора (миграция задач с одного процессора на другой для балансировки нагрузки недопустима). Также при распределении задач между ядрами необходимо решать NP-трудную проблему упаковки в контейнеры (bin-packing) в режиме офлайн.

При совместном многопроцессорном планировании задачи планируются, находясь в единой очереди готовности к исполнению, и могут перемещаться между процессорами. Основным преимуществом совместного планирования является наличие возможности преодоления алгоритмической сложности, присущей распределенному планированию. Поскольку все процессоры используют единую очередь задач, готовых к исполнению, это избавляет от необходимости решения проблемы распределения задач между процессорами в офлайн режиме, которая является источником вычислительной сложности распределенного планирования. Другим важным преимуществом совместного планирования является то, что, как правило, при его использовании требуется производить меньше операций вытеснения (preemption) задач: планировщик будет

вытеснять задачу только когда нет простаивающих (не выполняющих полезную работу, idle) процессоров. Совместное планирование проиллюстрировано на рис. 2.

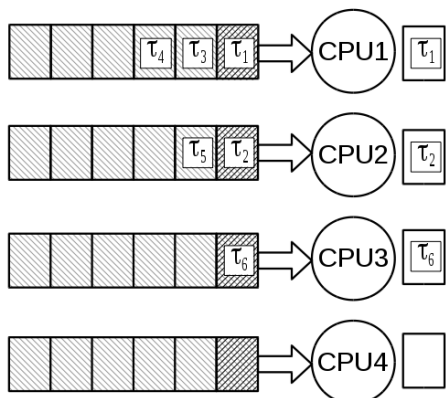


Рис. 1. Распределенное планирование

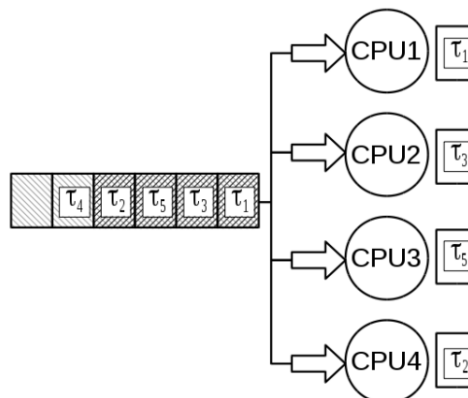


Рис. 2. Совместное планирование

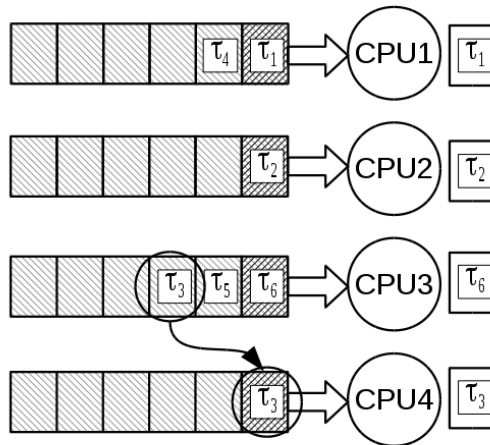
Но в отличие от распределенного планирования, алгоритмы однопроцессорного планирования не подходят для использования при совместном планировании из-за “эффекта Дала”. Также при совместном планировании необходимо решать задачу корректной работы с разделяемой памятью (shared-memory).

При совместном планировании накладные расходы мигрирующих задач могут быть очень высоки (зависит от архитектуры многопроцессорной системы). По факту задержки связаны с промахами кэша (cache miss) и коммуникационными нагрузками. А это в свою очередь может повлечь увеличение наихудшего времени исполнения задачи, что является крайне нежелательным в СРВ. С другой стороны, распределенные алгоритмы часто страдают от потерь вычислительной мощности, когда нагрузка на процессоры распределяется неравномерно. Чтобы преодолеть все эти недостатки предлагается использовать гибридные подходы планирования: частично распределенный (semi-partitioned) и кластерный (clustering).

При частично распределенном планировании (рис. 3) большинство задач выполняются только на одном процессоре, как и при классическом распределенном планировании. Однако несколько задач (или работ) могут мигрировать между двумя и более процессорами. Основная идея подобного метода планирования заключается в повышении коэффициента использования благодаря применению совместного планирования для тех задач, которые не могут быть закреплены только за одним процессором из-за ограничений эвристических алгоритмов решения задачи об упаковке в контейнеры, что является важным преимуществом. Задачи, которые не могут быть полностью закреплены за одним процессором, будут расщепляться (split) и связываться с несколькими процессорами. Но процесс связывания задач с процессорами происходит в режиме офлайн, что можно отнести к недостаткам данного подхода. На данный момент исследования в области повышения эффективности разбиения задач продолжают, позволяя сокращать накладные расходы, связанные с миграцией и вытеснением задач.

Кластерное планирование (рис. 4) также сочетает в себе преимущества распределенного и совместного планирований. Основной идеей кластерного планирования является разделение

$m$  процессоров на  $\left\lfloor \frac{m}{c} \right\rfloor$  наборов по  $c$  процессоров в каждом, образуя кластеры [7]. И распределенный, и совместный методы планирования можно рассматривать как частные случаи кластерного при  $c = 1$  и  $c = m$  соответственно.



$\tau_3$  - мигрирующая задача

Рис. 3. Частично распределенное планирование

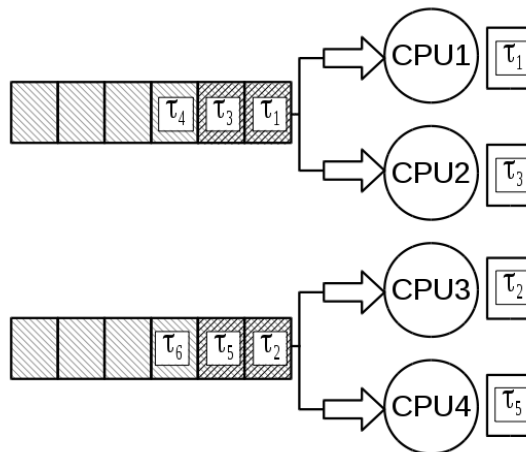


Рис. 4. Кластерное планирование

Изначально кластерное планирование было схоже с распределенным планированием, где задачи связывались с процессорами в режиме офлайн. В случае с кластеризацией задачи связывались с конкретным кластером, которому затем выделялся набор процессоров. Такой подход упрощал решение задачи упаковки в контейнеры для распределенного метода, и теперь задачи должны были быть распределены по кластерам. Для распределения задач между кластерами, повышения коэффициента использования, снижения накладных расходов, связанных с миграцией задач и уменьшением их времени отклика, могут применяться различные эвристические алгоритмы. Каждый кластер обрабатывает небольшое количество задач на небольшом количестве выделенных процессоров и тем самым устраняет проблему длинной очереди задач на исполнение, с которыми часто сталкиваются алгоритмы совместного планирования. Кластеризация также имеет определенную гибкость с точки зрения возможности создания кластеров для различных типов задач и задач с большим или маленьким значением коэффициента использования. Другим преимуществом кластеризации является то, что можно создать кластеры с различным объемом ресурсов, например с большим или маленьким количеством процессоров, имеющих один и тот же кэш второго уровня и т.д.

**Заключение**

На основании перечисленных ранее особенностей, преимуществ и недостатков существующих стратегий многопроцессорного планирования можно сделать следующие выводы. Распределенное планирование имеет более низкие накладные расходы во время работы системы, но процессорные мощности могут быть использованы неэффективно из-за проблемы упа-

ковки в контейнеры. В противоположность этому, совместное планирование устраняет проблему упаковки в контейнеры. Недостатком совместного планирования является увеличение накладных расходов, связанных с использованием общего разделяемого планировщика [8].

В зависимости от допустимости нарушений временных ограничений принято различать системы мягкого (soft, МРВ) и жесткого (hard, ЖРВ) реального времени. ЖРВ предусматривает наличие гарантированного времени отклика системы на конкретное событие и не допускает нарушения временных ограничений. В МРВ желательно, но не обязательно соблюдение всех временных предписаний. Как правило, распределенное планирование является предпочтительным в системах ЖРВ, а совместное планирование чаще применяют в системах МРВ.

Гибридные способы представляют собой «золотую середину» между распределенными и совместными способами планирования. Например, частично распределенные алгоритмы требуют меньшее количество процессоров, чем полностью распределенные для планирования одинакового набора задач. В то же время, эти алгоритмы не несут большие накладные расходы памяти, а также накладные расходы, связанные с миграцией и вытеснением задач, такие как при использовании совместного планирования. Гибридное планирование может применяться как в системах ЖРВ, так и в системах МРВ.

### Библиографический список

1. Boudjadar, A. David, J. Kim, K. Larsen, U. Nyman, A. Skou Schedulability and Energy Efficiency for Multi-core Hierarchical Scheduling Systems // Proceedings of the International Congress on Embedded Real Time Software and Systems ERTS, 2014, pp. 35-44.
2. K. Chakma, S. Debbarma, N. Kar, N. Debbarma, T. Debbarma Hierarchical Scheduling Approach for Symmetric Multiprocessing Based Real Time Systems on VxWorks // Lecture Notes on Software Engineering, Vol. 1, No. 1, February 2013, pp. 61-65
3. Осмоловский, С.В. Методы планирования задач в операционных системах реального времени // Научная сессия ГУАП: Ч.1 Технические науки. Сборник докладов 9-11 апреля 2012 г./ СПб.: - ГУАП, 2012, с. 100-104.
4. Li, H. Scheduling mixed-criticality Real-Time Systems // Ph.D. dissertation, University of North Carolina at Chapel Hill, 2013, p. 104.
5. Brandenburg, B. Scheduling and Locking in Multiprocessor Real-Time Operating Systems // Ph.D. dissertation, University of North Carolina at Chapel Hill, 2011, p. 614.
6. S. K. Dhall and C. L. Liu. On a real-time scheduling problem. // Operations Research, 26(1), 1978, pp. 127–140.
7. J. M. Calandrino, J. H. Anderson, D. P. Baumberger A hybrid real-time scheduling approach for large-scale multicore platforms. // Real-Time Systems, 2007. ECRTS '07. 19th Euromicro Conference on, April, pp. 247–258.
8. Осмоловский, С.В. Метод многоуровневого планирования задач в операционных системах реального времени для многоядерных процессоров // Научная сессия ГУАП: Ч.1 Технические науки. Сборник докладов 6-10 апреля 2015 г./ СПб.: ГУАП, 2015, с. 109-116.

### УДК 004.728.3

**А. Ю. Виноградов**

студент кафедры аэрокосмических компьютерных и программных систем

**Е. Н. Яблоков**

научный руководитель

### СОПОСТАВЛЕНИЕ ПРОТОКОЛОВ КАНАЛЬНОГО УРОВНЯ ДЛЯ СЕТИ SPACEWIRE

Канальный уровень – это второй уровень модели OSI, отвечающий за доставку кадров между устройствами, подключенными к одному сегменту сети.

Его основные функции:

1. Получение доступа к среде передачи.

- 2. Аппаратная адресация.
- 3. Обеспечение достоверности принимаемых данных.

Стандарт SpaceWire используется для создания высокоскоростной коммуникационной инфраструктуры для всех видов бортовых коммуникаций (1) [1]. Стандарт описывает транспортный, сетевой, канальный и физический уровни. Канальный уровень состоит из пакетного уровня и уровня обмена. Физический – из символьного, сигнального и физического уровней.

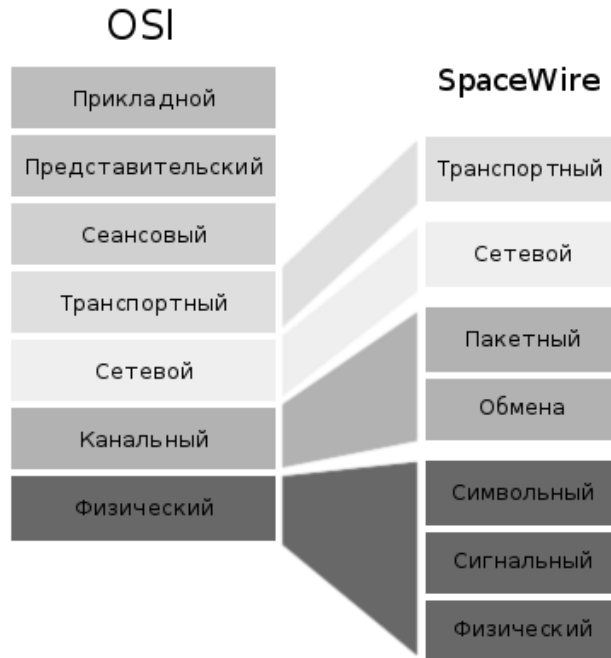


Рис. 1. Стек протоколов SpaceWire

Рассмотрим типы символов и пакеты стандарта SpaceWire на канальном уровне (рис.2). Символы стандарта SpaceWire, состоят из символа паритета, флага управления и информационных битов. Флаг управления показывает, какой символ передается в текущий момент. Если флаг равен 0, то это символ данных, если – 1, то это служебный символ. Бит паритета используется для выявления ошибок при передаче. Разный размер символов может вызвать проблемы при декодировании.

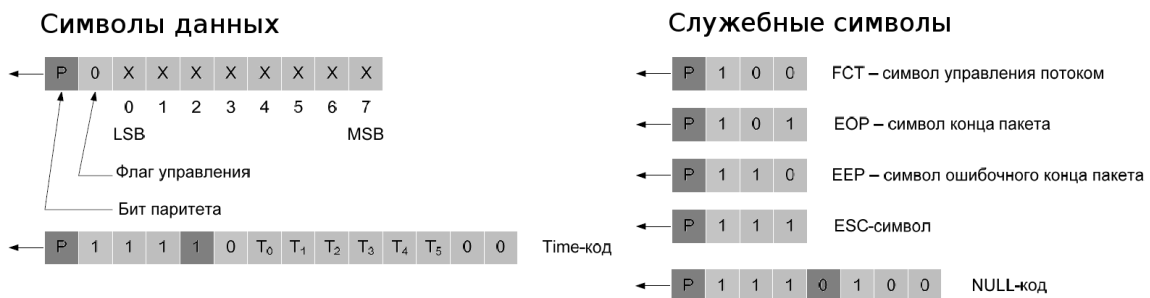


Рис. 2. Канальные символы SpaceWire

На рис. 3 изображён пакет стандарта SpaceWire, который состоит из трёх полей: адрес назначения, передаваемые данные и символ конца пакета. Адрес назначения имеет размер от 0 или более символов. Конец пакета может быть либо нормальным, либо ошибочным.

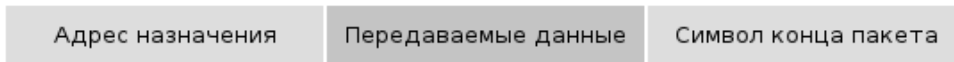


Рис.3. Пакет SpaceWire

Стандарт GigaSpaceWire дополняет существующий SpaceWire (рис. 4) и позволяет передавать данные на большей скорости и на большие расстояния[2]. До 100 м по медному кабелю и более 100 метров через оптический. Скорость передачи более 1 Гигабита. На рис. 4 видно, что для достижения таких показателей и обеспечения совместимости с существующим стандартом SpaceWire, GigaSpaceWire заменяет несколько нижних уровней стека протоколов.

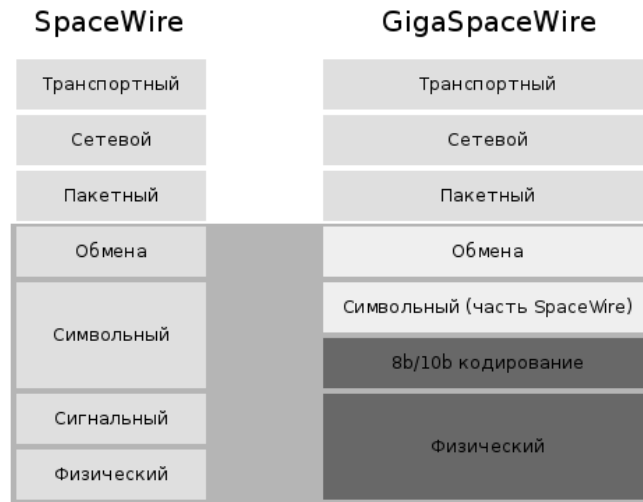
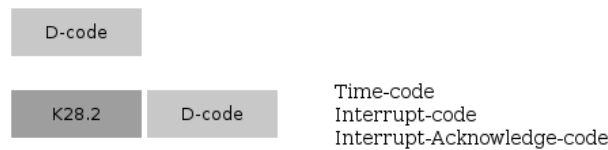


Рис.4.Стек протоколов SpW и GigaSpW

По стандарту GigaSpaceWire, каждый символ данных кодируется как D-символ стандарта 8b/10b (на рис. 5) изображены информационные и служебные символы стандарта gigaSpaceWire. Управляющие символы кодируются как K символы стандарта 8b/10b. Особое кодирование у управляющих символов, распределённых прерываний и меток времени. Данные символы кодируются последовательностью K символа и Ccode как D-символ 8b/10b. Формат пакета не отличается от пакета классического SpaceWire.

**Информационные символы**



**Служебные символы**

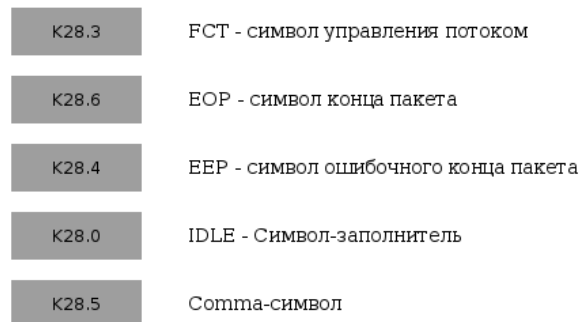


Рис. 5. Канальные символы gigaSpaceWire

В отличие от SpaceWire и GigaSpaceWire в Ethernet используются фреймы (рис. 6) [4]. Протокол Ethernet-SpaceWire, работает на канальном и физическом уровне OSI.

Задача этого протокола – передавать пакеты SpaceWire в фреймах Ethernet. В каждом фрейме, изображённом на рис. 6, содержится информация о типе, номере и замере передаваемого SpW пакета[3]. Поле кредита информирует другую сторону о том, сколько фреймов можно получить.

Для передачи пакетов SpW определено 7 типов фреймов, позволяющие передавать пакеты целиком или по частям, а также служебные символы. Во время передачи данных могут возникать ошибки, поэтому имеется тип фрейма информирующий об ошибках.

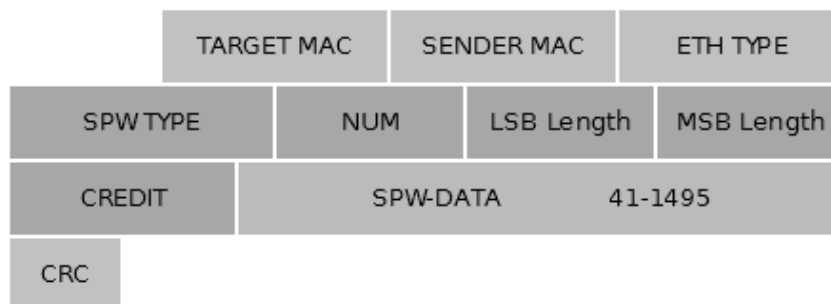


Рис.6.Формат фрейма протокола Ethernet-SpaceWire

В протоколе предусмотрена возможность создавать пользовательские типы пакетов.

В нашей реализации определены специальные фреймы, позволяющие настраивать мост Ethernet-SpaceWire и блок тестирования сети SpW.

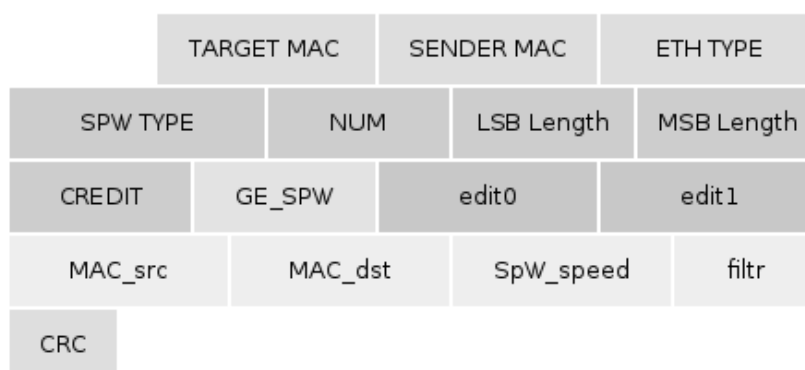


Рис. 7.Фрейм с настройками моста Ethernet-SpaceWire

На рис. 7 изображен фрейм с настройками, который содержит два поля флагов для выбора изменяемых параметров, адрес устройства, адрес устройства назначения, скорость соединения SpW и значение фильтра.

Фрейм с настройками тестера SpW также имеет два поля флагов, затем идёт количество перезапусков группы тестов. Далее идет одиночный тест, в заголовке которого записан его тип, размер, число перезапусков, пауза между перезапусками. Данный тест имеет ограничение на размер в 40 байт. После одиночного теста идет группа тестов, которые выполняются последовательно друг за другом. Количество и размер тестов ограничены размером фрейма Ethernet (рис. 8).

SpaceWire предназначен для передачи данных на небольшие расстояния на скорости до 400 мегабит. Используется тип связи узел-узел или плата-плата из-за небольших расстояний.

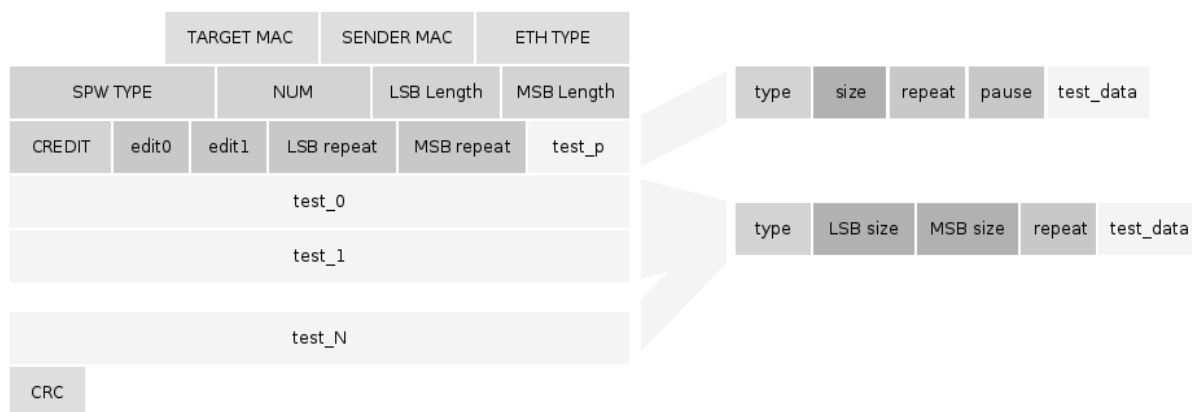


Рис. 8. Фрейм с настройками тестера SpaceWire

GigaSpaceWire предназначен для передачи данных на скорости 1 Gbit/c и больше. Используется тип связи плата-плата или сеть-сеть.

Ethernet-SpaceWire предназначен для передачи данных по устаревшим каналам связи. Используется для настройки и более простой связи с компьютером.

### Библиографический список

1. ECSS Standard ECSS-E-ST-50-12C, "SpaceWire, Links, Nodes, Routers and Networks", Issue 1, European Cooperation for Space Data Standardization, July 2008.
2. Yablokov E., Rozanov V., Vinogradov A. Protocol for Connection Ethernet Interface to SpaceWire Networks Proceedings of the 17th Conference of Open Innovations Association FRUCT, 2015.
3. Yablokov E., GigaSpaceWire – Gigabit Links for SpaceWire Networks / E. Yablokov, Y. Sheynin, E. Suvorova et al. // Proc. of the 5th International SpaceWire Conference, June 2013. P.28-34.
4. IEEE STANDARTS ASSOCIATION, IEEE Standart for Ethernet, New York, USA, December 2012.

УДК 004.896

**В. А. Волков**

студент кафедры аэрокосмических компьютерных и программных систем

**Б. Н. Седов**

научн. сотр. ВКИСТ – научный руководитель

## РЕАЛИЗАЦИЯ МЕТОДОВ УДАЛЕННОГО И АВТОНОМНОГО УПРАВЛЕНИЯ АВТОМОБИЛЕМ НА БАЗЕ COTS-ПЛАТФОРМЫ

Сегодня наблюдается стремительное развитие автономных систем управления автотранспортом, с каждым годом они совершенствуются и становятся все более пригодными к использованию в повседневной жизни. Такой способ управления автотранспортом приведет к снижению возникновения опасных ситуаций на дорогах, следовательно, снижению числа жертв при ДТП. Системы автономного управления условно можно разделить на два вида:

1. Outdoor – системы (на открытой местности), которые применяются для управления транспортными средствами на открытой местности. Разработка outdoor-систем автономного управления является весьма сложной задачей, так как в таких системах нужно учитывать множество факторов: наличие и положение пешеходов, неблагоприятные погодные условия, перекрытые участки дорог и т.д. Однако, стоит заметить, что разработки таких компаний, как Tesla, Audi, уже сейчас делают определенные успехи в достижении поставленных задач.



2. Indoor – системы (внутри помещения), которые применяются преимущественно в складских и промышленных помещениях. Такие системы позволяют заменить человека на монотонных и однообразных работах, что приводит к увеличению эффективности рабочего процесса и снижению вероятности возникновения неблагоприятных ситуаций, возникающих из-за человеческого фактора. Наглядные примеры Indoor систем – роботы-носильщики интернет-магазина Amazon, которые работают в складских помещениях. Автоматизированные склады позволяют справляться с потоком заказов и уменьшают вероятность формирования неверной посылки. Другой пример: – автоматизированные парковки компании Volkswagen, они выполнены в виде башни и хранят произведенные автомобили компании. Такой способ хранения обеспечивает заметную экономию места, а покупатели получают автомобиль с нулевым пробегом, так как от этапа начала производства до конечной выдачи покупателю автомобиль не проезжает ни километра.

На данный момент Indoor-системы автономного управления более развиты и распространены в следствии меньшей сложности требований по сравнению с Outdoor-системами.

Рассмотрим подробнее два существующих вида управления: удаленное и автономное.

Главное их отличие в том, что удаленная система управления требует наличие человека, который непосредственно будет управлять объектом управления либо на самом объекте управления, либо посредством проводного соединения, либо с помощью радиосигнала, либо другим способом. Автономное управление не требует наличие человека в процессе выполнения задачи. Наглядный пример, упомянутый ранее – автоматизированные склады компании Amazon. Оператор вводит входные данные, исполнение задачи происходит в автоматическом режиме

Каждый вид управления имеет свои недостатки и преимущества. Например, без участия человека удаленная система не будет функционировать, а автономная система пока что может успешно существовать только в практически идеальных, не зависящих от внешних факторов условиях (в случае возникновения нештатной ситуации система не может ее обработать).

Проект удаленного и автономного управления автомобилем на базе COTs платформы был выполнен в рамках конкурса World's largest Arduino maker challenge, проведенного компанией Arduino совместно с Microsoft.

Целью проекта являлось создание прототипа многофункциональной платформы на базе COTs платформы автомобиля с применением удаленного и автономного управления, а также с возможностью позиционирования внутри помещения.

Подобное решение может быть использовано в качестве автоматизированной системы доставки небольших посылок внутри помещения.

Обратимся к схеме проекта (рис. 1) и рассмотрим основные детали:

1. COTs платформа – радиуправляемая модель автомобиля в масштабе 1:24.

COTS – коммерческий продукт, готовый к эксплуатации сразу после приобретения.

2. Пульт радиуправления, используется для возможности удаленного управления, пульт поставляется в комплекте с радиуправляемой моделью.

3. Смартфон на базе Windows Phone для возможности удаленного управления с помощью смартфона по технологии Bluetooth.

4. Система определения местоположения. Для отслеживания местоположения платформы и указания пункта назначения спроектирован программный комплекс, центральной частью которого является облачный сервис Microsoft Azure.

5. Вспомогательные датчики. Инфракрасные датчики определения линии для следования по маршруту, а также ультразвуковой датчик для обнаружения препятствий.

6. Модуль связи. Bluetooth-модуль необходим для соединения со смартфоном и для взаимодействия с системой определения местоположения.

В COTs модель автомобиля были внесены существенные изменения и добавления. Распайка контактов (рис. 2) от управляющей платы произведена таким образом, чтобы сигналы, приходящие с радиуправляемого пульта, передавались сначала на Arduino, затем, в зависимо-

сти от выбранного режима управления, либо на двигатели платформы, либо на дальнейшую обработку.

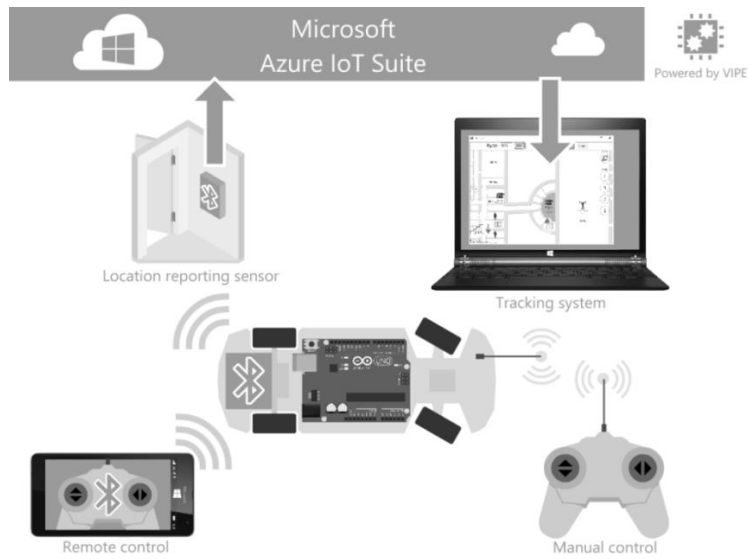


Рис 1. Принципиальная схема проекта

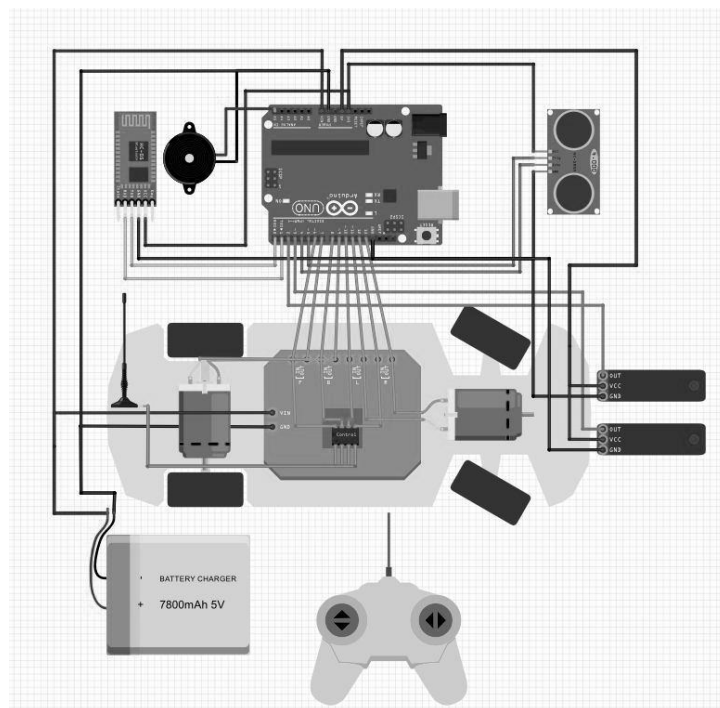


Рис 2. Схематика проекта

Возникла проблема нехватки места внутри модели автомобиля, поэтому было принято решение вывести провода наружу и расположить микроконтроллер Arduino\* и все необходимые для работы датчики снаружи автомобиля. Для этого был спроектирован и напечатан на 3D-принтере корпус.

В ходе тестирования системы возникла проблема быстрого разряда аккумуляторов. Решением проблемы стало замена пальчиковых батареек на съемный аккумулятор, который питает

\* Соммер У. Программирование микроконтроллерных плат Arduino/Freeduino: пер с англ. М.: БХВ-Петербург, 2012. 238 с

как плату Arduino, так и саму модель автомобиля. Это привело к значительному увеличению продолжительности работы платформы и удобству эксплуатации.

Алгоритм движения по линии разработан таким образом, чтобы платформа в случае выхода за пределы направляющей линии возвращалась на намеченную линию и не сбивалась с намеченного маршрута. Таким образом, платформа устойчива к резким изменениям направления линии маршрута, что позволяет не беспокоиться о механических ограничениях платформы.

Была добавлена возможность обнаружения препятствия на пути следования платформы. К примеру, маршрут следования пересекает человек, в этом случае платформа останавливается и как только человек уходит от линии платформа продолжает движение.

Пользователь может изменять режим и способ управления в реальном времени посредством специальной комбинации клавиш, ввод которых с устройства управления (пульт, телефон) приведет к смене режима управления в зависимости от комбинации.

Система определения местоположения состоит из следующих частей:

1. Программный комплекс на базе облачного сервиса для интернета вещей Microsoft Azure.
2. Датчики местоположения.

Датчик местоположения представляют собой Bluetooth-модуль, который сканирует область в пределах видимости с некоторым интервалом. Модуль фиксирует Bluetooth-устройства, входящие в список отслеживаемых (таким устройством является автоматизированная платформа) и отправляет данные в облачный сервис Azure. Облачный сервис посылает информацию о прибытии автомобиля в специализированный программный комплекс, который выводит информацию о местоположении платформы. Каждому Bluetooth-модулю соответствует свое заранее определенное местоположение. При обнаружении отслеживаемого устройства формируется специализированная ссылка, позволяющая использовать сервис навигации внутри университета и наглядно отобразить местоположение платформы.

Таким образом, система определения местоположения позволяет отслеживать текущее состояние платформы, на каком отрезке пути она находится, а также уведомлять о прибытии в конечный пункт назначения.

Был разработан и успешно протестирован прототип системы (рис. 3), реализующий методы удаленного и автономного управления автомобилем на базе COTS-платформы с применением микроконтроллера Arduino Uno и облачного сервиса для интернета вещей Microsoft Azure.



Рис. 3. Созданный прототип

Планируются дальнейшие работы над прототипом, в частности улучшение навигации внутри помещения путем применения видеоаналитики, что может привести к отсутствию необходимости в нанесении маршрутов. Также планируется ряд механических (или других) решений для преодоления лестниц и всевозможных препятствий на пути следования.

Несмотря на недостатки, платформа имеет очень большое преимущество – низкую себестоимость в сравнении с более сложными аналогами, а также широкую доступность элементной базы.

## УДК 656.021.2

**К. В. Гоголев**

студент кафедры системного анализа и логистики

**Н. Н. Майоров**

канд. техн. наук, доц. – научный руководитель

### МЕТОДЫ ПРОГНОЗИРОВАНИЯ ЗАГРУЖЕННОСТИ ГОРОДСКИХ ПРИЧАЛОВ САНКТ-ПЕТЕРБУРГА

В Санкт-Петербурге ежегодно увеличивается количество перевозимых по воде пассажиров. Это говорит о востребованности данного вида транспорта. Однако, констатируя положительную динамику в развитии водного транспорта, нельзя не сказать о проблемах, препятствующих дальнейшему росту пассажирских водных перевозок.

Одной из основных является недостаточное развитие и использование инфраструктуры внутреннего водного транспорта.

С целью монополизации причального фронта в наиболее востребованных при организации пассажирских перевозок внутренним водным транспортом местах и повышения эффективности использования городского имущества Комитет по транспорту разработал проект «Городские причалы Санкт-Петербурга».

В 2016 г. запланировано выполнить работы по монтажу и обустройству 15 причалов, состоящих из железобетонных понтонов, по следующим адресам:

- Английская набережная (напротив Александровского сада), спуск №8. Причал монтируется из 4 понтонов;
- Адмиралтейская набережная (напротив Адмиралтейства), спуск №10. Причал монтируется из 4 понтонов;
- Мытнинская набережная (правый берег Кронверкского пролива, в 90 метрах ниже по течению Кронверкского моста). Причал монтируется из 2 понтонов;
- Дворцовая набережная (напротив Летнего сада), спуск №22. Причал монтируется из 2 понтонов;
- Петровская набережная (напротив перекрестка Петровской наб. с Мичуринской ул.), спуск №2. Причал монтируется из 2 понтонов;
- Арсенальная набережная (напротив пл. Ленина), спуск №5. Причал монтируется из 2 понтонов;
- Университетская набережная (в створе Менделеевской линии), спуск № 5. Причал монтируется из 2 понтонов;
- Набережная Макарова, д.3, соор. 5, лит. Г5 – стационарный причал, состоящий из 2 причальных эстакад, соединенных переходным мостом, длина 74,5 метров;
- Южная дорога д.12, соор. 1;
- Приморский проспект (правый берег Большой Невки, ниже 3-го Елагиного моста). Причал монтируется из 2 понтонов;

- Набережная Мартынова (у 2-го Елагина моста). Причал монтируется из 4 понтонов;
- Набережная реки Мойки «Почтамтский мост – Поцелуев мост», лит. Б, спуск № 17. Причал монтируется из 1 понтона.
- Смольная набережная (напротив дома № 6 по ул. Смольного). По данному адресу Причал организуется без установки понтонов, с использованием существующего спуска набережной (необходимо осуществить оборудование спуска);
- Свердловская набережная (напротив гостиницы «Охтинская») – по данному адресу причал организуется без установки понтонов, с использованием существующего спуска набережной (необходимо осуществить оборудование спуска);
- Синопская набережная (напротив д.28). По данному адресу причал организуется без установки понтонов, с использованием существующего спуска набережной (необходимо осуществить оборудование спуска [1]).

Существует несколько методов прогнозирования, применяемых для оценки загруженности городских причалов Санкт-Петербурга:

- 1) метод средних величин;
- 2) линии тренда;
- 3) метод, основанный на оценке районов тяготения пассажиров;
- 4) имитационное моделирование.

Рассмотрим каждый из данных методов отдельно.

Метод средних величин основывается на показателе средней величины, который дает обобщенную характеристику варьирующего признака однородных единиц совокупности.

Статистическая обработка методом средних величин заключается в замене индивидуальных значений варьирующего признака  $x_1, x_2, x_3 \dots x_n$  некоторой уравновешенной средней величиной  $x$ .

Основываясь на статистике швартовок к городским причалам в 2015 г. (табл. 1), можно сказать, что среднее количество швартовок к одному причалу в день составило – 4 причаливания.

Таблица 1

Статистика швартовок за навигацию 2015 г.

Адрес причала	Месяц							Общее количество
	май	июнь	июль	август	сентябрь	октябрь	ноябрь	
Адмиралтейская наб.	56	116	96	104	3	29	14	418
Английская наб.	50	656	915	983	434	9	0	3047
Арсенальная наб.	24	40	27	34	5	1	0	131
Дворцовая наб.	16	243	271	292	256	4	0	1082
Мытнинская наб.	16	205	252	244	253	42	20	1032
Петровская наб.	4	4	2	4	7	4	0	25
Университетская наб.	0	6	9	17	8	1	0	41
Приморский пр.	3	2	18	9	5	0	0	37
Общее количество	169	1272	1590	1687	971	90	34	5813

На основании средней величины можно сделать прогноз, что общее количество швартовок в навигацию 2016 г. достигнет числа в 11040 швартовок, так как всего причалов 15, а дней навигации – 184.

Линии тренда. На основе статистических данных (см. табл. 1), можно получить законы распределения, которые будут справедливы для причалов в навигацию 2016 г. Рассмотрим данный метод на примере причала по адресу Английская набережная (рис. 1).

Аппроксимировав статистические данные полиномом третьего порядка, получим коэффициент достоверности аппроксимации  $R^2=0,9442$ , чем ближе  $R^2$  к 1, тем точнее модель описывает имеющиеся данные. По линии тренда можно построить таблицу с вероятным количеством швартовок для конкретных причалов. Так, на примере рассмотренного причала можно прогнозировать его загруженность на 2016 г. в районе 3116 швартовок.

Метод, основанный на оценке районов тяготения пассажиров. Основными пользователями городских причалов являются перевозчики, организующие водные прогулки для туристов, поэтому оценку инфраструктуры около причалов следует производить по следующим критериям:

- наличие гостиниц и ресторанов(кафе);
- наличие культурных центров;
- рабочие центры.

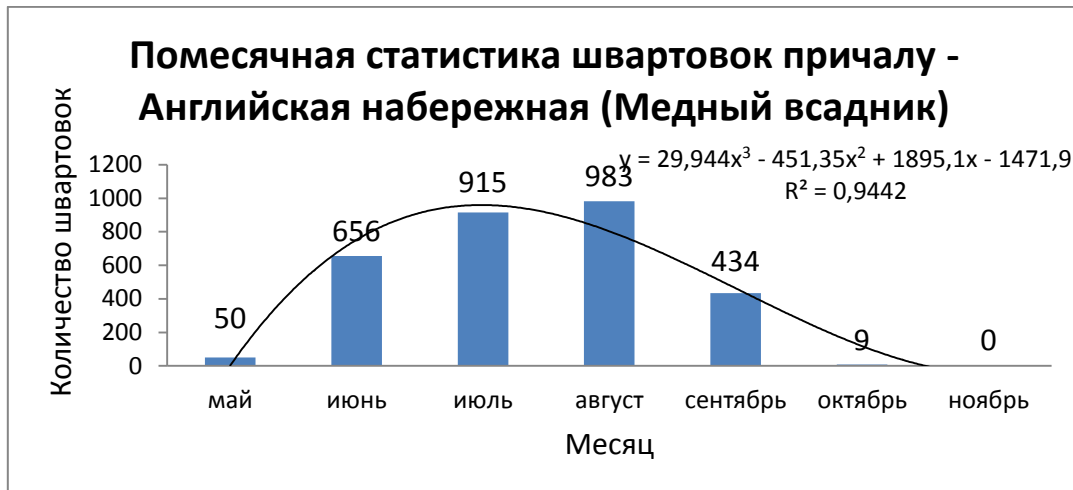


Рис. 1. Аппроксимация статистических данных

Проведя такой анализ и опираясь на статистику 2015 г., можно прогнозировать загруженность причалов на 2016 г.

Имитационное моделирование. Причальный фронт является частным случаем сложных динамических систем, включающих в свой состав городские причалы и причалы, работающие на правах аренды и координирующих свою коммерческую деятельность на реках и каналах Санкт-Петербурга.

Для того чтобы достичь оптимальности работы всего причального фронта в целом, необходимо проводить исследования всех элементов данной системы: причалы, перевозчики, инфраструктура вокруг причалов.

Причал является классической системой массового обслуживания (СМО).

Работа любой СМО заключается в обслуживании поступающих в нее требований, или заявок. Заявками в данном случае являются потенциальные пользователи причалами, т. е. суда. Заявки поступают в систему по расписаниям движения судов, указанным компаниями перевозчиков. Обслуживание поступившей заявки продолжается какое-то время, после чего система освобождается для обслуживания очередной заявки (рис. 2) [2].

Система массового обслуживания включает четыре основных элемента: входящий поток, очередь, обслуживающее устройство(причалы), выходящий поток.

Работу СМО можно прогнозировать, учитывая ряд допущений относительно протекания процессов обслуживания.

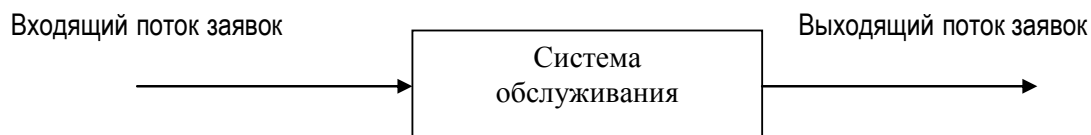
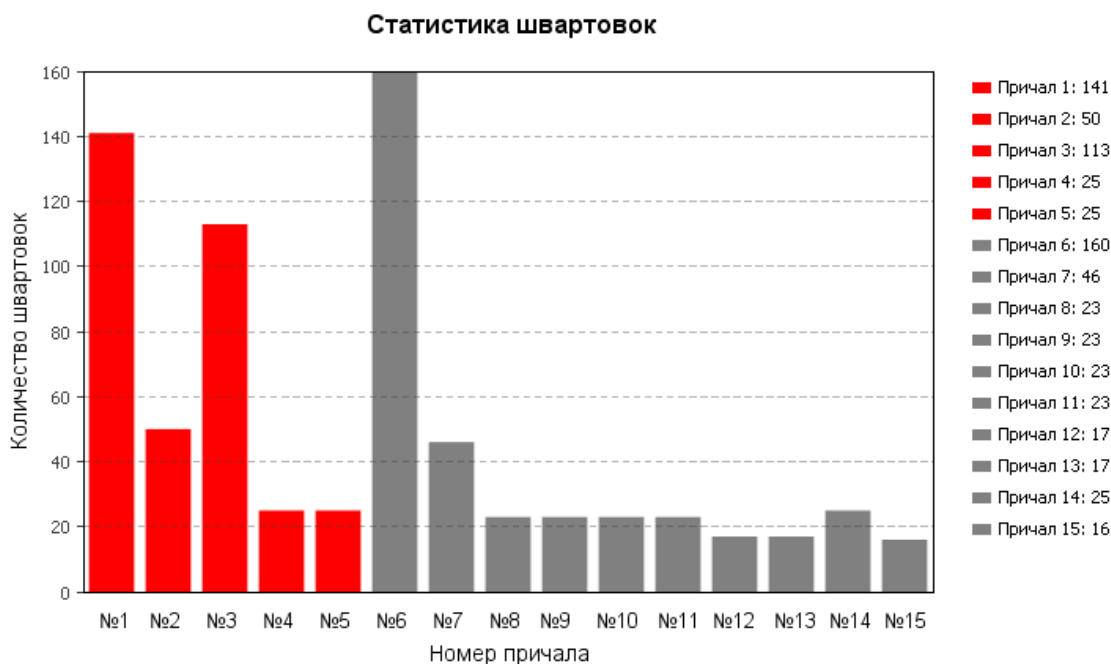


Рис. 2. Обобщенная схема СМО

Имитационная модель СМО состоит из трех взаимосвязанных модулей:

1. Моделирование поступления заявки в систему (появление судна в системе).
2. Моделирование всевозможных перемещений судов по маршрутной сети.
3. Моделирование обслуживания судов на обслуживающих устройствах (причалах).

Прогноз загрузки причалов будет производиться на основе данных, полученных с помощью третьего модуля, в котором собирается статистика по количеству швартовок (рис. 3).



**Общее количество швартовок за сутки: 727**

**Количество швартовок к городским причалам за сутки: 354**

Рис. 3. Статистика загрузки причалов

Таким образом на основании представленных методов прогнозирования можно оценить эффективность использования городских причалов (табл. 2).

Таблица 2

Итоги оценки эффективности использования городских причалов

Год	2015	2016		
Критерий оценки	Статистика за навигацию 2015 г.	Годовой прогноз по среднему количеству швартовок к одному причалу в навигацию 2015 г.	Годовой прогноз исходя из законов распределения и схемы районов тяготения пассажиров	Прогноз по результатам моделирования при максимальной загрузке причалов (5 городских причалов, 122 дня из 184)

Год	2015	2016		
Количество швартовок	5 813	11 040	6 303	43 188

Сравнив результаты моделирования, можно утверждать, что на сегодняшний день городские причалы загружены, к сожалению, на 13,5% от реально возможной загруженности. Этот факт является ключевым для дальнейшего рассмотрения расходов для реализации проекта и доходов от его функционирования.

### Библиографический список

1. СПб ГКУ «Агентство внешнего транспорта» [Электронный ресурс] // Водный транспорт – Режим доступа: <http://avt.spb.ru/>
2. Майоров Н.Н., Фетисов В.А. Моделирование транспортных процессов: учеб. пособие. СПб. ГУАП, 2011. 165 с.
3. Фетисов В.А. и др. Моделирование транспортных процессов. // Методические указания к выполнению лабораторных работ. СПб.: ГУАП, 2013. 31 с.

УДК 004.453

**С. С. Горкалов**

студент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**В. Г. Никитин**

канд. техн. наук, доц. – научный руководитель

## ПРОЕКТИРОВАНИЕ СИСТЕМЫ УПРАВЛЕНИЯ ОПТИЧЕСКОЙ СТАНЦИИ НА ОСНОВЕ ТРЕХМЕРНОЙ ТВЕРДОТЕЛЬНОЙ МОДЕЛИ И ФИЗИКО-МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ

В современном мире существуют задачи проектирования сложных систем. В силу специфики проектирования невозможно учесть все нюансы. Для предотвращения ошибок при проектировании были разработаны вычислительные системы, позволяющие существенно упростить проектирование различных механических систем с помощью физико-математического моделирования. Эти системы значительно ускоряют и уменьшают затраты на этапе макетирования. Они применяются в приборостроении, в том числе в задачах автоматического управления. Наиболее перспективной и инновационной считается САПР SolidWorks, в которой была создана виртуальная конструкторская модель системы. Для разработки системы автоматического управления и проверки физических процессов устройства, разработанного в SolidWorks, необходимо провести физическое моделирование. Для выполнения данной задачи выбран программный комплекс имитационного моделирования Simulink с библиотекой SimMechanics фирмы MathWorks.

Основное достоинство метода заключается в том, что трехмерное моделирование существенно превосходит линейное проектирование. Программа SolidWorks позволяет создать трехмерную модель, предоставляя наиболее подробное описание свойств объекта (масса, объем, моменты инерции), позволяя работать в виртуальном объемном пространстве, благодаря чему на самом высоком уровне происходит приближение компьютерной модели к облику будущего изделия. Процесс построения трехмерной модели основывается на формировании объемных геометрических элементов и выполнении разных операций между ними. Модель создается из шаблонных элементов (блоков), она подлежит редактированию путем добавления (удаления) данных блоков или посредством изменения их характерных параметров. Используемая в среде имитационного моделирования Simulink (Matlab) библиотека SimMechanics предоставляет среду



для трёхмерного моделирования многотельных механических систем, что в совокупности даёт мощный аппарат для проектирования и моделирования сложных мехатронных систем.

Рассмотрим пример проектирования системы управления оптической станции на основе ее трехмерной твердотельной модели. Оптическая станция состоит из двух частей: станции обнаружения и станции слежения. Станция обнаружения, представляет из себя вращающуюся с постоянной скоростью платформу, состоящую из 6-ти объективов. Область контроля:  $360^\circ$  по азимуту,  $-10..20^\circ$  по углу места. Станция обнаружения, сканируя область вокруг станции, выдаёт полноэкранное изображение за 1 с, таким образом вращается со скоростью  $360^\circ/\text{сек}$  с темпом выдачи данных на каждый градус. Станция слежения, представляет из себя вращающиеся объективы, состоит из 3-х объективов. Область контроля:  $360^\circ$  по азимуту,  $-20..50^\circ$  по углу места. Станция слежения последовательно наводится на все объекты, которые были замечены станцией обнаружения и сторонними источниками, для уточнения угловых координат и создания траектории объектов.

Получена упрощенная трехмерная твердотельная модель оптической станции в САПР SolidWorks2012 с необходимыми степенями свободы для вращения подвижных частей и древа модели (рис.1).

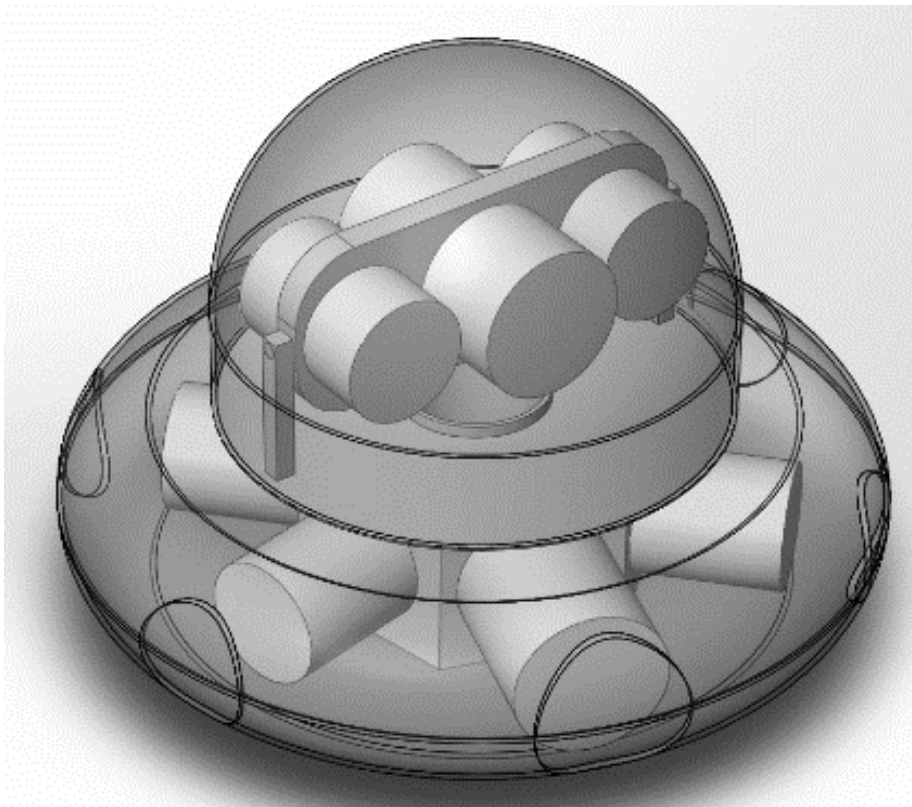


Рис. 1. Упрощенная модель оптической станции

Модель содержит в себе следующую информацию: положения центров масс деталей, ориентацию в пространстве каждой детали, массу деталей, форму деталей, тензоры инерции деталей, сопряжения деталей между собой. Используя CAD-транслятор, который обеспечивает создание динамических моделей механизмов в среде SimMechanics, на основе их твердотельных моделей, разработанных в CAD-системах (в нашем случае SolidWorks). При применении CAD-транслятора тензоры моментов инерции и присоединительные размеры передаются из CAD-системы в SimMechanics без изменений. При этом работоспособность моделей проверяется в CAD-системе посредством установления правильных связей между деталями механизма. После

экспорта в рабочую среду Simulink мы получаем следующую схему (рис.2).

Получена реализация системы управления в САПР Simulink (рис. 3). Выходной сигнал с блока *Скорость* поступает на блок *Обнаружение*, который математически описывает работу электродвигателя, используемого в системе.

Выходной сигнал с блока *Обнаружение* проходит через блок *Оптическая станция*. Сигнал с блока *Joint Sensor* проходит через демультиплексор, преобразующий входной сигнал в сигналы: *Угол*, *Скорость*, *Ускорение*, *Момент*. Сигнал *Скорость* подается на блок *Обнаружение* для реализации обратной связи. Подобным образом описывается работа по каналам азимута и угла места.

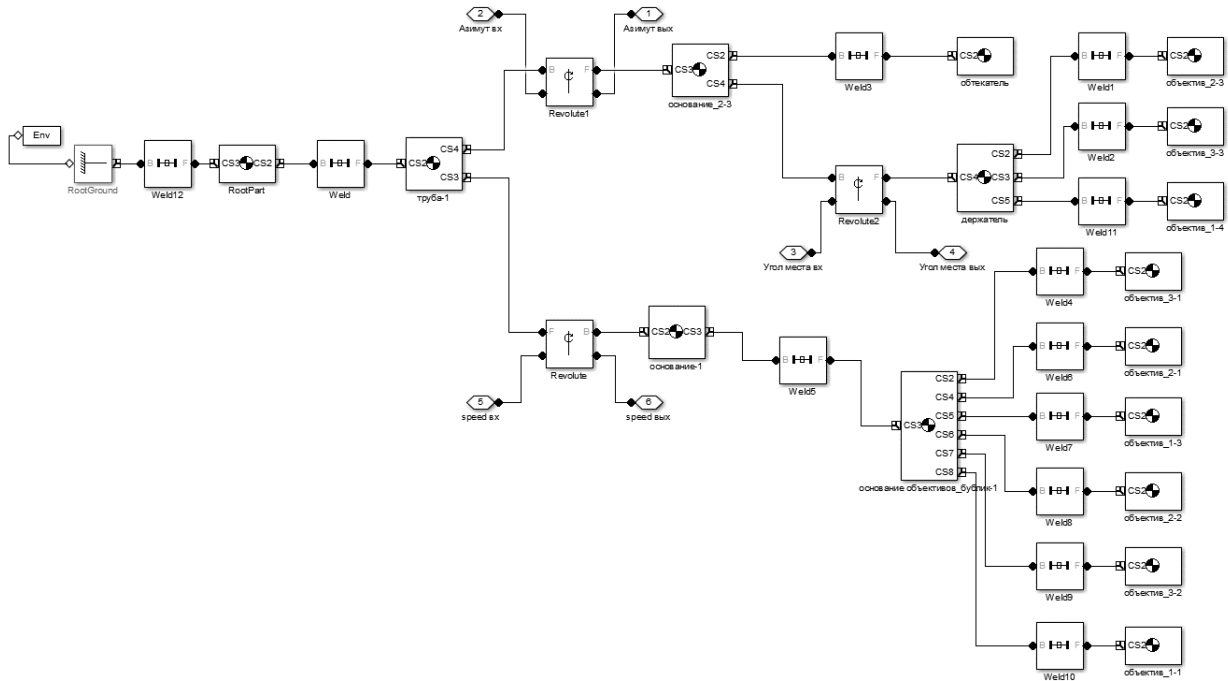


Рис. 2. Структурная схема блока «Оптическая станция»

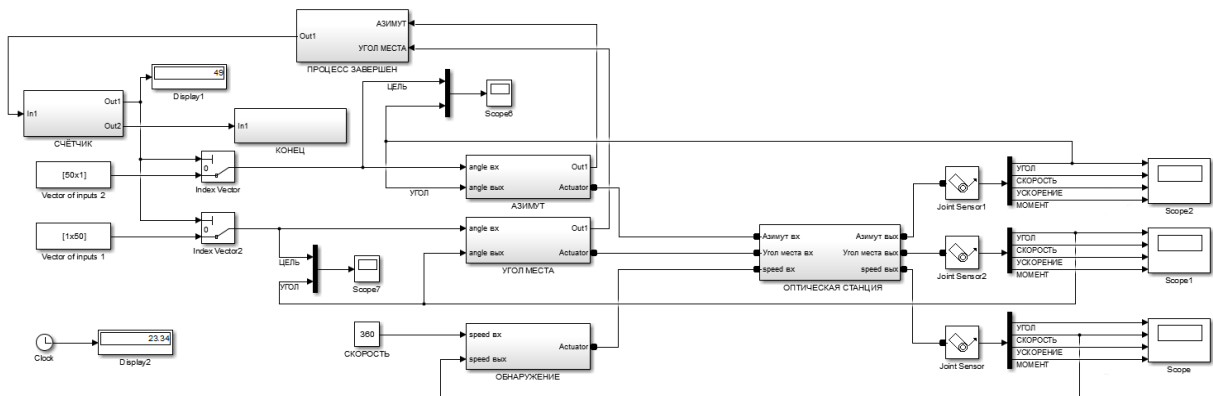


Рис. 3. Система управления оптической станцией

Результатом моделирования являются графики переходных процессов по каналам угла, скорости, ускорения и момента. Ниже приведены графики описывающие работу системы по каналу обнаружения (рис. 4).

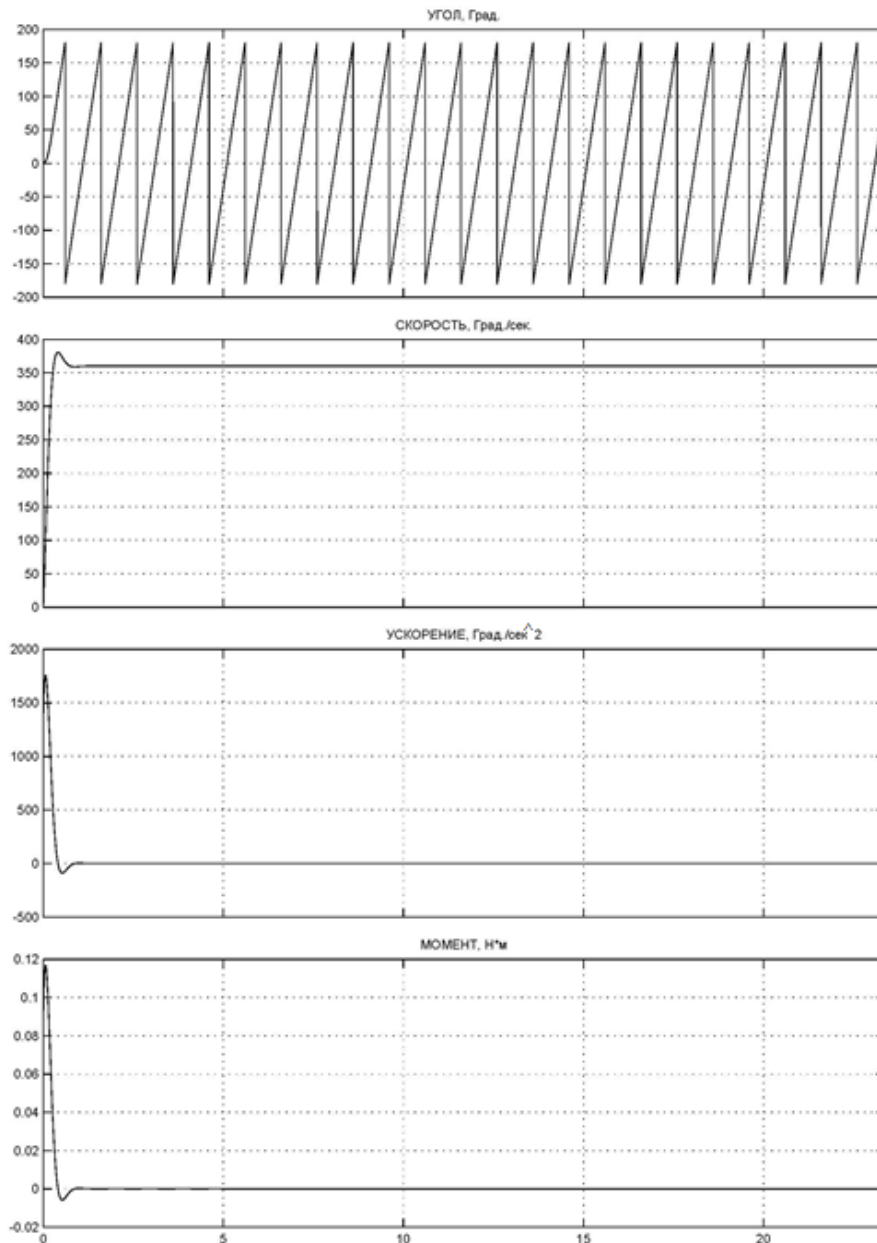


Рис. 4. Графики переходных процессов по углу, скорости, ускорению и моменту канала обнаружения

По графику скорости можно сказать, что перерегулирование равно 5%, переходный процесс длится 1,5 с. На основе проведенного исследования результаты которого приведены на графиках (см. рис. 4.) можно сделать вывод, то полученная система управления оптической станции удовлетворяет требованиям по динамическим характеристикам.

Предложенный подход моделирования на основе полученных динамических характеристик позволяет вносить изменения в конструкцию изделия до этапа производства и значительно ускоряет этап макетирования. Разработанная система создана на базе АО «НПК ПЕЛЕНГАТОР».

#### Библиографический список

1. Бесекерский В.А. Теория систем автоматического управления (4-е издание). Спб.: Профессия, 2003. 752 с. .
2. Блинов О.В., Кузнецов В.Б. Исследование механических систем в среде SimMechanics (MatLab) с использованием возможностей программ трехмерного моделирования. Методические указания.: Иваново, 2012. 19 с.

3. Моделирование мехатронных систем в среде Matlab (Simulink/ Simmechanics): учеб. пособ. / В.М. Мусалимов, Г.Б. Заморюев, И.И. Калапышина, и др. СПб: НИУ ИТМО, 2013. 114 с.

УДК 656.614.2

**А. В. Гунягина**

студентка кафедры системного анализа и логистики ГУАП

**Н. Н. Майоров**

доц., канд. техн. наук – научный руководитель

## ВОПРОСЫ РАЗРАБОТКИ ИМИТАЦИОННОЙ МОДЕЛИ МОРСКОГО ПАССАЖИРСКОГО ПОРТА

Главной задачей для качественной и бесперебойной работы «Пассажирского порта Санкт-Петербурга «Морской фасад» является слаженная работа множества подразделений и служб, которые в случае возникновения проблемных ситуаций должны быстро принимать меры по их ликвидации. Численность подразделений и служб может меняться в зависимости от суточной загруженности вокзала. Целью работы является моделирование пассажирских потоков и работы служб «Пассажирского порта Санкт-Петербурга «Морской фасад» для формирования системы принятия решения о качественной работе порта и количестве необходимого персонала для обслуживания пассажиров.

Исторические сведения и более подробная информация о строительстве пассажирского порта даны ранее в статье «Вопросы развития морских пассажирских перевозок в Балтийском бассейне на примере города Санкт-Петербург».

На сегодняшний день метод имитационного моделирования является одним из самых мощных и наиболее эффективных методов исследования процессов. Сущность этого метода состоит в том, чтобы создать в компьютерной программе модель, имитирующую процесс функционирования системы с целью получения статистических характеристик моделируемой системы. Используя результаты имитационного моделирования, можно описать поведение системы, оценить влияние различных параметров системы на ее характеристики, выявить преимущества и недостатки предлагаемых изменений, прогнозировать поведение системы.

Системы массового обслуживания (далее СМО) отличаются высокой наглядностью отображения моделируемых объектов и вследствие этого сравнительной простотой перехода от реальных объектов к соответствующим СМО (рис. 1) [1].

Основная цель разработки автоматизированной системы имитационного моделирования СМО – реализация современных подходов к проведению имитационного моделирования [1].

Работа любой СМО заключается в обслуживании поступающего на нее потока требований, или заявок. Заявками в данном случае являются туристы.

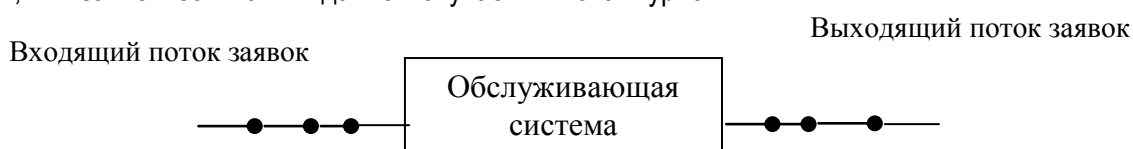


Рис.1. Обобщенная схема СМО

Система массового обслуживания включает несколько основных элементов: входящий поток, очередь для прохода контрольно-пропускного пункта, обслуживающее устройство (рамка металлоискателя и рентгеновский сканер), очередь для прохода секторов пограничного контроля, обслуживающее устройство (металлодетекторы и интроскопы) и выходящий поток. С каждым из них связан ряд возможных допущений относительно протекания процессов обслуживания.

Для оптимизации работы актуально применение имитационного моделирования в программе AnyLogic. AnyLogic – программное обеспечение для имитационного моделирования сложных систем и процессов. Программа обладает графической средой пользователя и позволяет использовать язык Java для разработки моделей. Модели AnyLogic могут быть основаны на любой из основных методов имитационного моделирования: дискретно-событийное моделирование, системная динамика, и агентное моделирование. [2]

Дискретно-событийное моделирование – это вид имитационного моделирования. В дискретно-событийном моделировании функционирование системы представляется как хронологическая последовательность событий. Событие происходит в определенный момент времени и знаменует собой изменение состояния системы.

Дискретно-событийное моделирование работает в низком и среднем диапазонах. Что же касается агентного моделирования, то оно может применяться практически на любом уровне и в любых масштабах. Агенты могут представлять пешеходов, автомобили или роботов в физическом пространстве, клиента или продавца на среднем уровне, или же конкурирующие компании на высоком. [2]

Основное средство процессного моделирования в AnyLogic – это библиотека Enterprise Library.

В эту библиотеку вошли объекты для определения «потока» процесса (process workflow):

- PedSource создает пешеходов (обычно используется в качестве начальной точки блок-схемы, формализующей поток пешеходов). Создает пешеходов любых подклассов базового класса Ped через случайные промежутки времени. Пешеходы прибывают согласно интенсивности, равной 1000/час.

- PedSink удаляет поступивших в объект пешеходов из моделируемой среды (обычно объект используется в качестве конечной точки блок-схемы, формализующей поток пешеходов). PedSink автоматически ведет подсчет пешеходов.

- PedWait заставляет пешеходов перейти в заданное место (pedArea) и ожидать там в течение определенного периода времени. Пешеходы находятся в режиме ожидания в течение заданного промежутка времени, отсчитанного относительно достижения точки ожидания. Точность достижения места ожидания равна 0,1 м.

- PedService направляет поток пешеходов через группу сервисов и очередей, заданных в объекте PedServices.

- PedGoTo заставляет пешеходов перейти в заданное место моделируемого пространства, которое задано линией. Переход будет считаться выполненным, когда пешеход пересечет заданную линию. Пешеходы будут искать путь к заданному транзиту в пределах текущего этажа. Режим выбора пути к заданной линии автоматический, т.е. путь вычисляется библиотекой автоматически. Точность достижения цели равна 0,25 м. Тайм-аут определения блокировки равен 5 с.

- PedSelectOutput является блоком принятия решения пешеходной библиотеки. Агент, вошедший в блок PedSelectOutput, будет перенаправляться в один из пяти выходных портов в зависимости от заданных для этих портов коэффициентов предпочтения.

- PedEscalator задаёт движение и направление эскалаторов.

Все объекты гибкие и настраиваемые: параметры могут изменяться динамически, действия могут зависеть от атрибутов заявок и т.д.

За основу для моделирования в данной статье был взят Пассажирский порт Санкт-Петербурга «Морской фасад». Морской вокзал – сложный комплекс взаимодействия различных служб, участвующих в одном единственном комплексе по обслуживанию и перевозке пассажиров. Работа вокзала зависит от множества параметров, таких как: наличие человеческого фактора; прохождение паспортного и пограничного контроля; состояние информационных систем; продолжительность разгрузки/загрузки пассажиров и др. Нарушение работы одного из факторов приводит к тому, что вызывает затруднение движения пассажиропотока и нарушение целостности работы всей системы, а это ведет к потере в пропускной способности и экономическим поте-

рям. В связи с этим возникает задача исследования, планирования и оптимизации работы морского вокзала, которая достаточно трудно решается традиционными алгоритмическими методами. Решение этой задачи позволит ответить на вопросы: как повлияет изменение нагрузки на работу всех систем вокзала в комплексе; что произойдет, если в несколько раз увеличится пассажиропоток; как повлияет внедрение новых служб или новый режим работы на пассажиропоток? Помочь ответить на эти вопросы может создание специализированной имитационной модели, которая прогнозирует основной бизнес-процесс деятельности вокзала – обслуживание пассажиров. Для создания имитационной модели пассажиропотока в данной работе в качестве примера был взят пассажирский паром «Princess Anastasia», пассажироместимость которого составляет 2000 человек.

Моделируя работу вокзала, можно предварительно спрогнозировать места в системе, которые могут быть источниками задержек, и таким образом заранее избежать сбоев.

С помощью имитационного моделирования можно проводить эксперименты для оценки работы системы, меняя такие параметры, как расписание, график прибытия пассажиров, правила их обслуживания и маршруты перемещения по терминалу, количество оборудования в зонах регистрации, расположение магазинов, таможни и т.д.

В любой системе периодически возникают отклонения, такие как задержка парома или выход из строя какого-либо оборудования, поэтому необходимо оперативно принимать решения о перераспределении ресурсов, об изменении графика работы или порядка обслуживания.

На рис. 2 представлена логическая цепь движения пассажиров в порту в программе графоанализатор.

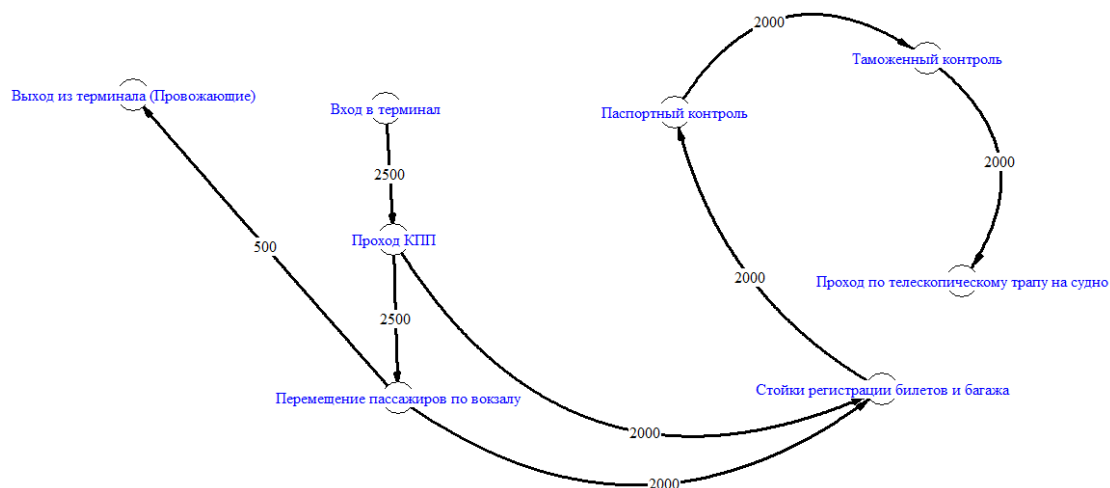


Рис. 2. Логическая цепь движения пассажиров в порту

В каждой из этих подсистем пассажир затрачивает разное время на обслуживание. В связи с этим математическая модель пропускной способности приобретает следующий вид:

$$\sum_{i=1}^n t_i$$

–  $\sum_{i=1}^n t_i$  – суммарное время обслуживания, где  $n$  – кол-во операций;  
 –  $i$  – технологические этапы обработки пассажира.

Суммарное время обслуживания пассажиров можно представить в развёрнутом виде:

$$\sum_{i=1}^n t_i = t_1 + t_2 + t_3 + t_4 + t_5 + t_6 + t_7 + t_8 + t_9, \text{ где}$$

- $t_1$ . – время на проверку пассажира на входе в порт;
- $t_2$ . – время на проверку багажа пассажира при повороте;

- $t_3$  . – свободное время, проведенное в зале ожидания;
- $t_4$  . – время на выдачу билета на стойке регистрации;
- $t_5$  – время на ожидание паспортного контроля;
- $t_6$  – время на прохождение паспортного контроля;
- $t_7$  . – время на ожидание таможенного контроля;
- $t_8$  . – время на прохождение таможенного контроля;
- $t_9$  – время, затрачиваемое пассажиром на проход по телескопическому трапу.

После разработки алгоритма выполняется важный этап – разработка структурной схемы самой имитационной модели (рис. 3) и установка взаимосвязи между элементами, составляющими сложную техническую систему.

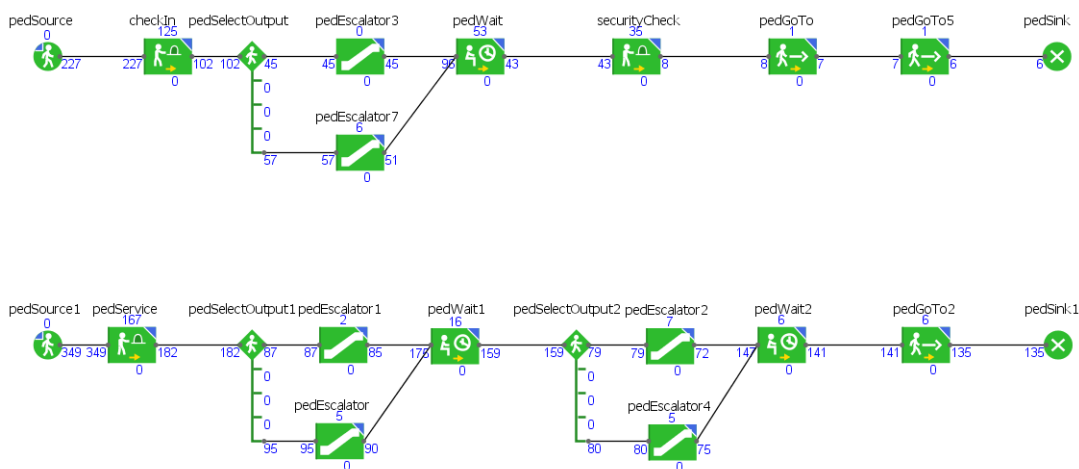


Рис. 3. Структурная схема имитационной модели работы пассажирского порта

В результате мы получаем 3D-визуализацию процессов, происходящих в системе (Рис. 4), плотность потока в зависимости от загруженности очереди (рис. 5) и статистику количества людей при перемещении в пассажирском порту (рис. 6).



Рис. 4. 3D-визуализация процессов происходящих в системе



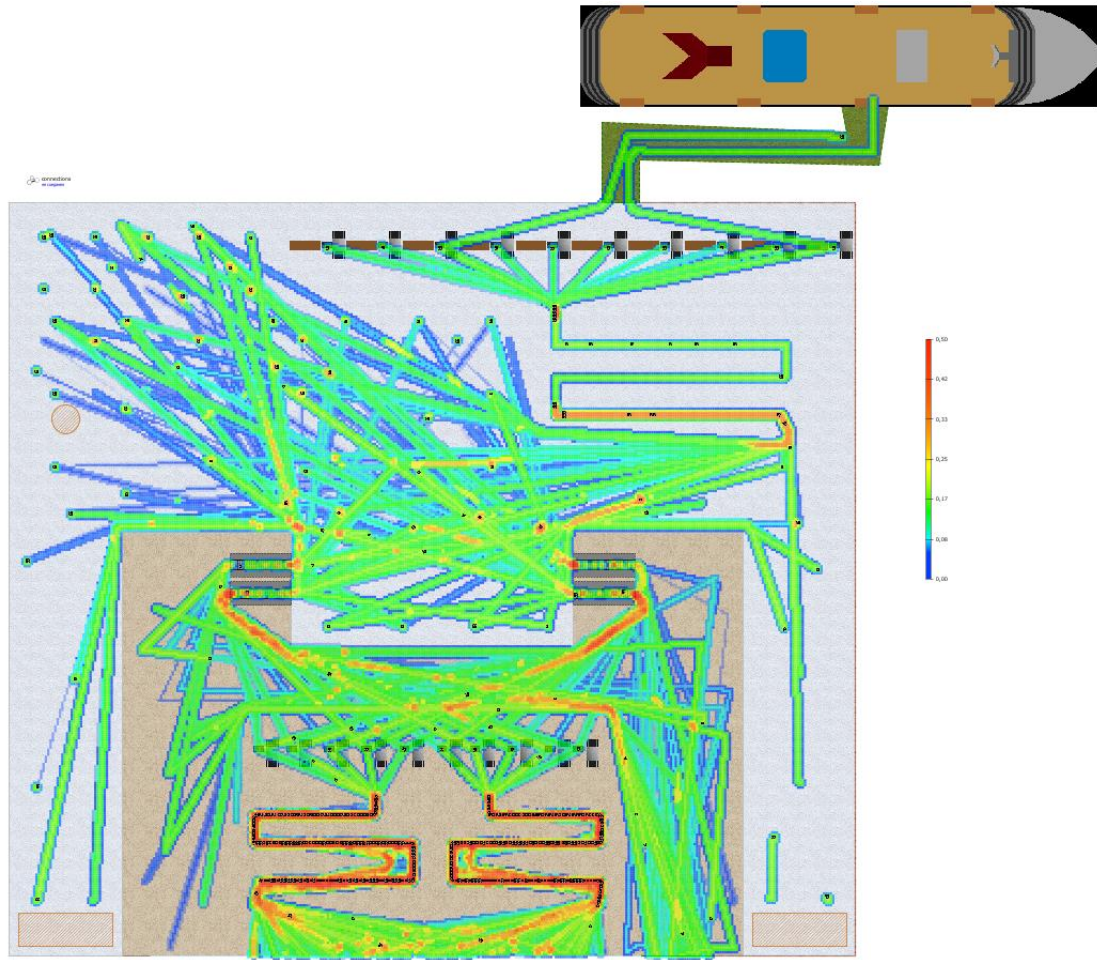


Рис. 5 Плотность потока в зависимости от загруженности очереди

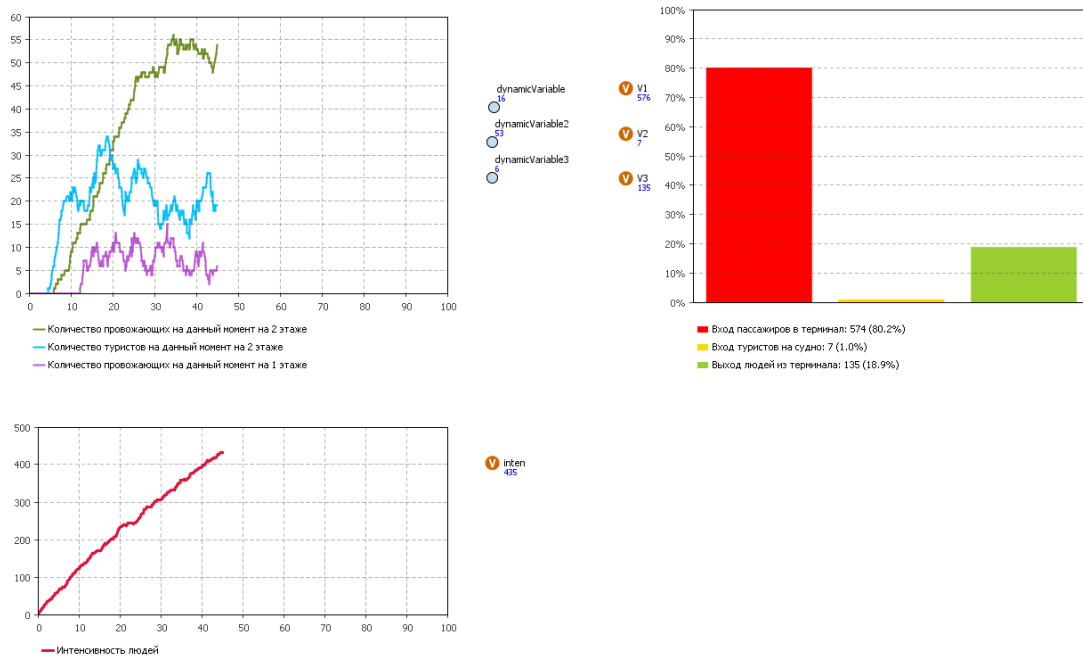


Рис. 6 Статистика количества людей при перемещении в пассажирском порту

Планирование эвакуации. AnyLogic позволяет тестировать различные варианты эвакуации, отражая агрессивное поведение людей в экстренной ситуации. Модели также помогут вы-



явить места, потенциально удобные для совершения правонарушений, в том числе терактов, а также спрогнозировать потенциальный ущерб от таких действий.

В результате создана математическая имитационная модель пассажиропотоков с использованием агентного моделирования в среде AnyLogic. В модели представлена работа «Пассажирского порта Санкт-Петербурга «Морской фасад» и реализована работа всех подразделений терминала.

Практическая значимость: разработанная имитационная модель не только обладает высокой точностью моделирования, но и позволяет всего за считанные минуты получать прогноз пассажиропотока на несколько часов вперед. Имитационная модель учитывает особенности всех важнейших элементов вокзала, влияющих на пассажиропоток.

### Библиографический список

1. Даненова Г. Т. «Применение имитационных моделей для исследования систем массового обслуживания».
2. Карпов Ю. Г. Имитационное моделирование систем. Введение в моделирование с AnyLogic 5. СПб: БХВ-Петербург, 2006. 400 с.
3. Майоров Н. Н., Фетисов В. А., Таратун В.Е. Моделирование транспортных процессов Методические указания к выполнению лабораторных работ. СПб.: ГУАП, 2013.

УДК 004.93'1, 004.932.2, 681.7

**Н. А. Деготинский**

магистрант кафедры аэрокосмических компьютерных и программных систем

**В. Р. Луцке**

д-р. техн. наук, профессор – научный руководитель

## ИЗМЕРЕНИЕ ГЛУБИНЫ ПО ДЕФОКУСИРОВКЕ ИЗОБРАЖЕНИЯ

Введение

Задача определения удаленности объектов на изображении давно известна в компьютерном зрении. Извлечение из изображения информации о «глубине» объектов позволяет решить такие практические задачи, как, в частности: оценка качества изображения и отдельных его элементов, выделение сфокусированных элементов изображения, восстановление фокуса в дефокусированных (не находящихся в фокусе) зонах.

Для решения задачи в настоящее время могут использоваться методы на основе анализа величины углового рассогласования объектов, методы, анализирующие форму движения объекта, методы определения глубины на основе информации о фокусе и дефокусе изображения. Недостатком этих методов является необходимость наличия серии изображений для анализа, либо пары камер [1], что не всегда удобно или допустимо. Кроме того, перед применением большинства методов необходимо произвести трудоемкий процесс калибровки камеры для определения ее оптических параметров.

В настоящей статье рассматривается альтернативный метод оценки удаленности объектов на изображении. Метод позволяет рассчитать относительное расстояние до объектов, используя информацию о дефокусировке единственного изображения. Метод не требует наличия серии снимков, а также проведения операции калибровки, и для его успешного применения достаточно одного снимка с обычной камеры. Ниже приведено краткое описание основных принципов его работы.

Описание модели дефокусировки изображения

Для начала представим небольшой теоретический минимум, показывающий основы формирования изображения на сенсоре камеры. Предполагается, что дефокусировка подчиняется модели тонкой линзы (рис. 1).

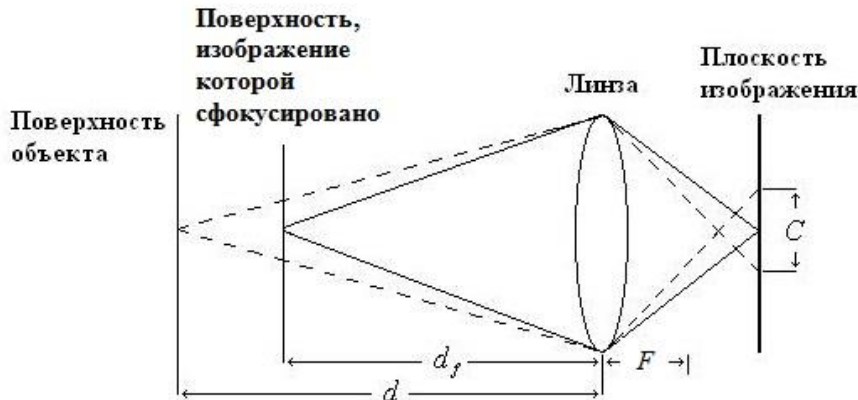


Рис. 1. Формирование изображения при помощи тонкой линзы

Для получения сфокусированного изображения необходимо, чтобы все испускаемые им лучи, проходя через линзу, пересеклись в одной из точек изображения. В случае, когда изображение не лежит на линии фокуса, лучи, проходящие через линзу, ложатся на разные точки сенсора камеры, в результате чего изображение получается смазанным [2].

Величина диаметра пятна рассеивания (CoC – Circle of Confusion) характеризует диаметр размытия и может быть рассчитана при помощи формулы:

$$C = \frac{d - d_f}{d} \cdot \frac{f^2}{N(d_f - f)},$$

в которой  $f$  – фокусное расстояние,  $N$  – число диафрагмы;  $d_f$  – расстояние до объекта, находящегося в фокусе,  $d$  – расстояние до объекта, находящегося не в фокусе.

Из этой формулы видно, что диаметр пятна рассеивания представляет собой нелинейную функцию, возрастающую с увеличением расстояния от камеры до объекта. Размытие же изображения может быть промоделировано как свертка исходного изображения с функцией рассеивания точки (PSF – Point Spread Function). В качестве ядра свертки обычно используется функция Гаусса:

$$g(x, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}},$$

где  $x, y$  – декартовы координаты пикселя, а между параметром  $\sigma$  и диаметром  $C$  пятна размытия имеется прямо пропорциональная зависимость [3]. Оценка удаленности же объекта сводится к оценке неизвестного параметра  $\sigma$  [3].

Принцип работы описываемого метода

Алгоритм работает в три основных этапа:

1. Выделение границ.

В качестве ключевых точек, в которых будут производиться вычисления величины дефокуса, выбираются граничные (угловые) точки изображения. Угловые точки предлагается находить при помощи детектора контуров Canny. Данный детектор дает в результате границу без разрывов, является довольно точным и устойчивым к шумам [4].

2. Расчет величины дефокусировки.

Вычисление величины дефокусировки на границах в свою очередь состоит из следующих шагов.

Шаг 1. Вычисление градиента на границах.

Модель идеального края может быть представлена как

$$f(x) = Au(x) + B,$$

где  $f(x)$  – функция Хевисайда,  $A$  – амплитуда края,  $B$  – его линейное смещение. Отметим, что сама граница находится в точке  $x = 0$ . Тогда модель смазанной границы может быть записана как

$$i(x) = f(x) \otimes g(x, \sigma),$$

где  $\otimes$  – оператор свертки.

Используя свойство дифференцирования оператора свертки, градиент граничной точки можно вычислить по следующей формуле:

$$\nabla i(x) = A \otimes g(x, \sigma) = \frac{A}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

Шаг 2. Применение размытия к граничным точкам и повторный расчет градиента

После проведения повторного размытия граничных точек Гауссианом с параметром  $\sigma_0$  модель повторно размытого края будет выглядеть следующим образом:

$$r(x) = i(x) \otimes g(x, \sigma_0) = f(x) \otimes g(x, \sigma) \otimes g(x, \sigma_0).$$

Далее, с использованием свойства о свертке двух Гауссианов запишем градиент повторно размытого края:

$$\nabla r(x) = \frac{A}{\sqrt{2\pi(\sigma^2 + \sigma_0^2)}} e^{-\frac{(x-(\mu_1+\mu_2))^2}{2(\sigma^2 + \sigma_0^2)}}.$$

Шаг 3. Вычисление неизвестной величины дефокусировки

Сначала необходимо найти отношение модулей рассчитанных градиентов

$$R = \frac{|\nabla i(x)|}{|\nabla r(x)|} = \sqrt{\frac{\sigma^2 + \sigma_0^2}{\sigma^2}} e^{-\left(\frac{x^2}{2\sigma^2} - \frac{x^2}{2(\sigma^2 + \sigma_0^2)}\right)}.$$

Затем, используя свойство получения максимального градиента будет максимальным на границе ( $x = 0$ ), отношение принимает вид

$$R = \sqrt{\frac{\sigma^2 + \sigma_0^2}{\sigma^2}}.$$

Из полученного отношения находим искомое значение дефокусировки  $\sigma = \sqrt{\frac{\sigma_0}{R^2 - 1}}$ . При

этом видно, что полученная оценка дефокусировки зависит только от параметра повторного смаза  $\sigma_0$  и величины отношения градиентов  $R$ .

3. Распространение рассчитанных величин на все изображение

Последним шагом является интерполирование полученных значений в граничных точках на все изображение. Для этого подходит ряд методик, одной из которых является применение интерполирующего (материрующего) Лапласиана. Матированием называют восстановление переднего плана изображения путем предоставления подсказок восстанавливающей системе в виде разметки. Данная методика подробно описана в работе [5]. В нашем случае в качестве разметки будет выступать разреженная карта относительной глубины, составленная из рассчитанных на

предыдущем этапе оценок дефокусировки на границах. Задача нахождения полной карты относительной глубины может быть записана как задача минимизации следующей функции [3]:

$$E(d) = d^T L d + \lambda (d - \bar{d})^T D (d - \bar{d}),$$

где  $\bar{d}$  – разреженная карта относительной глубины, записанная в векторной форме;  $d$  – полная карта глубины в векторной форме;  $L$  – матрица матирующего Лапласиана;  $D$  – диагональная матрица, элементы  $D_{ij}$  которой равны единице, если пиксель исходного изображения с индексом  $i$  лежит на границе;  $\lambda$  – параметр, отвечающий за баланс между точностью и гладкостью результатов интерполяции;  $L$  – симметричная матрица размера  $N \times N$ , где  $N$  – размерность изображения, представленного в векторной форме. Элементы матрицы  $L_{ij}$  задаются следующим образом:

$$L_{ij} = \sum_{k|(i,j) \in \omega_k} \left( \delta_{ij} - \frac{1}{|\omega_k|} \left( 1 + (I_i - \mu_k)^T \left( \Sigma_k + \frac{\varepsilon}{|\omega_k|} U \right)^{-1} (I_i - \mu_k) \right) \right).$$

где  $\delta_{ij}$  – дельта Кронекера, равна единице при равенстве индексов и нулю в противном случае;  $\mu_k$  – матрица средних значений цветов пикселей в окне  $\omega_k$ ,  $\Sigma_k$  – матрица ковариации окне  $\omega_k$ ,  $I_i$  – пиксель с индексом  $i$ ;  $|\omega_k|$  – размер окна;  $U$  – единичная матрица;  $\varepsilon$  – параметр регуляризации. Задача минимизации функции (10) может быть выполнена при помощи решения следующей разреженной системы линейных уравнений

$$(L + \lambda D)d = \lambda D\bar{d}$$

в результате решения которой находится вектор  $d$ . «Вытянув» по столбцам вектор  $d$  в матрицу, получается искомая полная карту глубины.

Примеры работы метода

Результаты работы алгоритма для каждой из представленных ниже фотографий показаны с помощью трех изображений (рис.2): исходного изображения а; контурного – б, демонстрирующего оценки величин дефокусировки границ величиной яркости контурной линии; и итогового в, представляющего построенную плотную карту глубины. Самым темным цветом обозначены объекты, находящиеся ближе всего к камере. На рис. 2, в видны незначительные погрешности оценки дальности на мелких деталях, но в целом алгоритм достаточно точно определил относительную удаленность объектов от камеры. Ошибка определения ободка на столбе, вероятно, связана с отражением в нем поверхности земли.

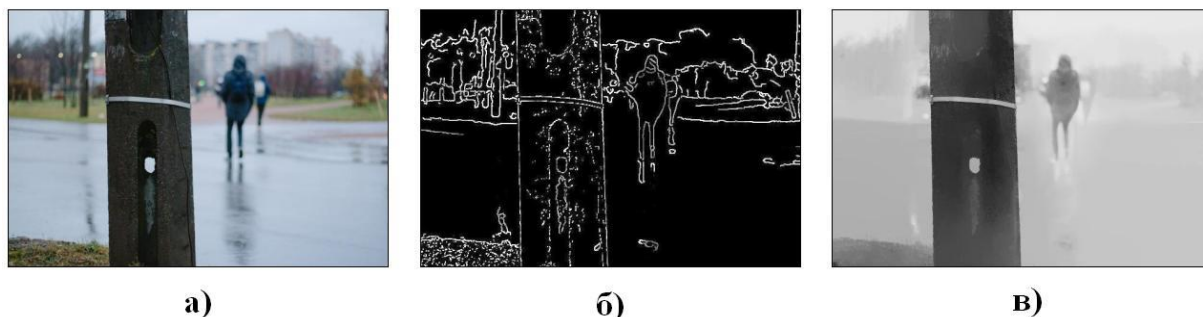


Рис. 2. Результаты анализа сцены «Переход»

На рис. 3, в. также видны только небольшие погрешности измерения дальности мелких деталей в правой части фотографии. Метод не идеален и подобные ошибки могут быть вызваны тенями либо некорректной интерполяцией.

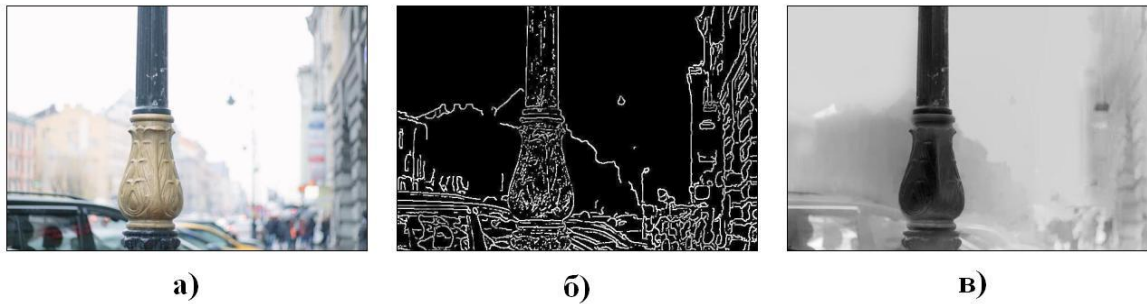


Рис. 3. Результаты анализа сцены «Невский Проспект»

### Библиографический список

1. Chaudhuri S., Rajagopalan A.N. Depth from Defocus: A Real Aperture Imaging Approach. Springer, 1999. 172 p.
2. Ray S. F. Applied Photographic Optics: Lenses and Optical Systems for Photography, Film, Video, Electronic and Digital Imaging. Oxford: Focal Press, 2002. 89 p.
3. Zhuo S., Sim T. On the recovery of depth from a single defocused image // Lecture Notes in Computer Science. 2009. V. 5702. P. 889–897
4. Canny J. A Computational Approach To Edge Detection // IEEE Trans. Pattern Analysis and Machine Intelligence. 1986. P. 679–698.
5. He K., Sun J., Tang X. Fast Matting Using Large Kernel Matting Laplacian Matrices // Computer Vision and Pattern Recognition (CVPR) IEEE Conference. 2010. P. 2165 – 2172.

### УДК 629

#### **А.Ю. Дикан**

студентка кафедры аэрокосмических измерительно-вычислительных комплексов

#### **Н.С. Коньшев**

студент кафедры аэрокосмических измерительно-вычислительных комплексов

#### **М.Е. Тихомиров**

доц. – научный руководитель

## СИСТЕМА ОПРЕДЕЛЕНИЯ ЦЕНТРА МАСС САМОЛЕТА В ПОЛЕТЕ

Положение центра тяжести механической системы играет важную роль в любой отрасли техники, а особенно важно для самолета, имеющего шесть степеней свободы. Определение центра тяжести самолета с ошибкой в 2% длины самолета может привести к снижению экономичности, к непоправимой катастрофе как при эксплуатации летательного аппарата (ЛА), так и при его постройке и испытаниях [1].

Актуальность рассматриваемой темы заключается в том, что в современном мире особую важность приобретает безопасность полета. Одними из важнейших факторов обеспечения безопасности, устойчивости и балансировки самолета являются положение центра тяжести и центровка самолета.

Во время полета происходит выработка топлива, сброс различных грузов, парашютистов, иногда меняется размещение грузов на борту самолета. Все это может негативно повлиять на балансировку, а также на устойчивость и управляемость самолета.

Равновесие, балансировку, управляемость и устойчивость можно обеспечить только если точно определено положение центра тяжести ЛА и это положение поддерживается в полете в заданных пределах [2]. Существуют различные методы определения положения центра масс на Земле. Центровка самолета, определенная методами взвешивания на Земле, расчетным методом при изменившейся нагрузке и определенная по центровочным графикам, не учитывает ре-

альное перераспределение масс по самолету. Это относится в первую очередь к расходованию авиационного топлива. При этом современные топливомерные системы вычисления положения центровки в полете не отличаются достаточной точностью определения текущей центровки [3].

Учитывая вышеизложенное, была разработана новая концепция определения положения центра масс самолета в полете, которая основывается на измерении отдельных параметров движения фюзеляжа в точках, распределенных вдоль продольной оси самолета.

Самолет в полете постоянно обладает вращательным движением относительно центра масс, обусловленным короткопериодическим и длиннопериодическим движением под действием внешних возмущений (турбулентность атмосферы), перемещения центра масс по длине фюзеляжа, а также управляющим воздействием от экипажа и/или системы автоматического управления (САУ). Амплитуды этих колебаний могут быть достаточно малыми, однако, чувствительность современных датчиков ускорения и угловой скорости позволяет измерять параметры малых движений.

Предлагаемый метод измерения положения центра масс требует измерения параметров движения в нескольких точках, распределенных вдоль длины фюзеляжа. В этих точках должны измеряться текущие значения нормального ускорения  $a_{yj}$ , а также местная угловая скорость  $\omega_{zj}$  вращения шпангоутов, на которую закреплены измерительные модули.

Измерение угловых скоростей необходимо для оценки величины ускорений, вызванных деформациями упругого фюзеляжа. Данные ускорения и угловые скорости воспринимаются измерительными модулями и учитываются для более точного вычисления точки, относительно которой осуществляются продольные колебания самолета, т. е. точки центра масс.

Вблизи от центра масс располагается измерительный модуль, который измеряет, собственно, угловую скорость продольного вращения самолета, а также величину ускорения самолета в центре масс при маневрировании. Значение данного ускорения должно быть учтено при обработке информации в системе измерения положения центра масс.

В системе определения центра масс самолета в полете предусматривается размещение шести измерительных модулей, распределенных по длине фюзеляжа. В данной работе примером служит пассажирский ближние, среднемагистральный самолет МС-21 («Магистральный самолет XXI в»), новое наименование Як-242, точнее, его модификация МС-21-200. Длина фюзеляжа данного самолета составляет 37 м. Каждый из 6 модулей располагаем с шагом 7,15 м (рис.1). Планируемое расположение модулей:

1. В переднем техническом отсеке, на верхней полке авионики, или у шпангоута №8 у пола, в отсеке кухни.
2. На задней стенке багажного отсека под полом кабины, у шпангоута №20.
3. У задней стенки центрального топливного бака, у шпангоута №30.
4. На передней стенке в заднем багажном отсеке.
5. У задней стенки заднего багажного отсека.
6. Базовый модуль располагаем в техническом отсеке под полом кабины, на верхней полке авионики.

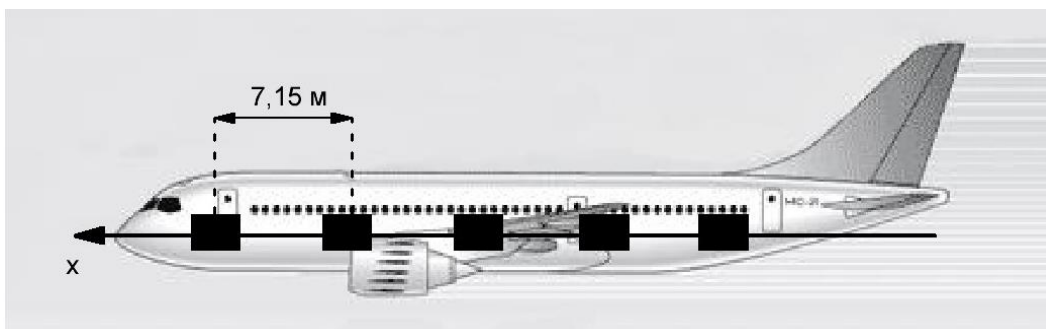


Рис.1. Размещение измерительных модулей на борту летательного аппарата.

Результат измерения ускорений в  $i$ -й точке представляют собой алгебраическую сумму из ускорений движений центра масс при маневрировании  $a_{y.ц.м}(t)$ , ускорения  $a_{y.ж}(t)$  от колебательного движения при вращении относительно центра масс для гипотезы абсолютно жесткого фюзеляжа, и ускорения  $a_{y.упр}(t)$   $i$ -го сечения фюзеляжа, вызванных упругими колебаниями фюзеляжа при изгибах. Данные ускорения приведены в выражении (1.1):

$$a_{y_i}(t) = a_{y.ж}(t) \pm a_{y.ц.м}(t) \pm a_{y.упр}(t), \quad (1.1)$$

где  $a_{y.ж}(t)$  – ускорения в  $i$ -й точке, если считать самолет абсолютно жестким;  $a_{y.ц.м}(t)$  – ускорение центра масс, при наличии нормальной перегрузки (находится ближе к центру масс);  $a_{y.упр}(t)$  – ускорения, вызванные упругими колебаниями фюзеляжа.

На рис. 2 показаны эпюры распределения ускорений, которые являются следствием влияния колебаний измерительных модулей при короткопериодическом движении при гипотезе фюзеляжа как абсолютно жесткого тела. На основе решения задачи о подобии треугольников, на основе этой эпюры можно вычислить текущее положение центра масс самолета.

При наличии ускорения движения центра масс при маневрировании самолета, измеренное соответствующим модулем значение  $a_{y.ц.м}(t)$  следует вычесть из каждого  $a_{y_i}(t)$ .

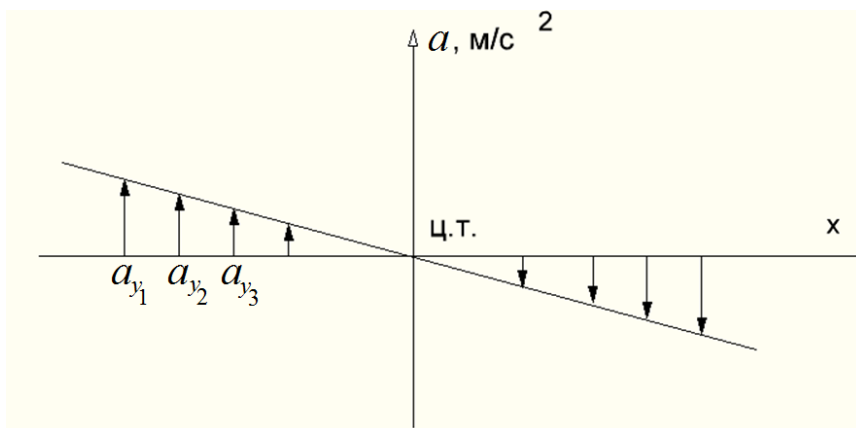


Рис.2. Графическое изображение эпюр самолета.

Датчики угловых скоростей, расположенные в модулях, распределенных по фюзеляжу, измеряют значения  $\omega_{z_i}(t)$ , соответствующего угловой скорости вращения шпангоута силового набора фюзеляжа в месте установки конкретного модуля. Измерение угловых скоростей необходимо для вычисления параметров изгиба фюзеляжа при упругих деформациях [4]. Для вычисления формы изогнутого фюзеляжа и по этим данным – значения нормального ускорения, вызванного упругими колебаниями, используем решение определенного интеграла (1.2), если мы проинтегрируем эту функцию от  $t_1$  до какого-то времени  $t_2$  по  $dt$  при начальном условии мы получим угол  $\phi_0$ , где  $\phi(t)$  – угол поворота шпангоута из-за упругих свойств (1.2) :

$$\phi(t) = \phi_0 + \int_{t_1}^{t_2} \omega_{z_i}(t) dt. \quad (1.2)$$

Так, измерив в каждом сечении значение угловой скорости, можем рассчитать угол поворота шпангоута и местный угол касательной к продольной строительной оси самолета. На этой основе получаем параметры деформаций фюзеляжа, и с учетом координат установки измерительного модуля – значение нормального ускорения в данной точке, вызванное упругой дефор-



мацией фюзеляжа. Определенное таким образом значение  $a_{y.ynp}(t)$  должно вычитаться из значения ускорения  $a_{y_i}(t)$ . Таким образом, выражение для вычисления значения примет вид

$$a_{y.ж_i}(t) = a_y(t) - a_{y.ц.м}(t) - a_{y.ynp}(t). \quad (1.3)$$

Определенное таким образом значение  $a_{y.ж_i}(t)$  в каждой точке установки измерительных модулей используется в вычислении текущего положения центра масс самолета.

При разработке аппаратной части макета системы измерения положения центра масс была разработана функциональная схема измерительного модуля (рис.3), выбрана элементная база. Функциональная схема представлена на рисунке. В качестве датчика линейных ускорений выбран МЭМС-датчик, акселерометр типа LMS303DLH фирмы STMicroelectronics. При выборе акселерометра анализировались его технические характеристики. В качестве датчика угловой скорости  $\omega_{z_i}(t)$  применена МЭМС типа A3G4250D фирмы STMicroelectronics.

Сбор данных от датчиков по интерфейсу I<sup>2</sup>C осуществляется микроконтроллером. С учетом сравнительно малой стоимости, габаритов и универсальных характеристик был выбран микроконтроллер типа STM32F100 фирмы STMicroelectronics.

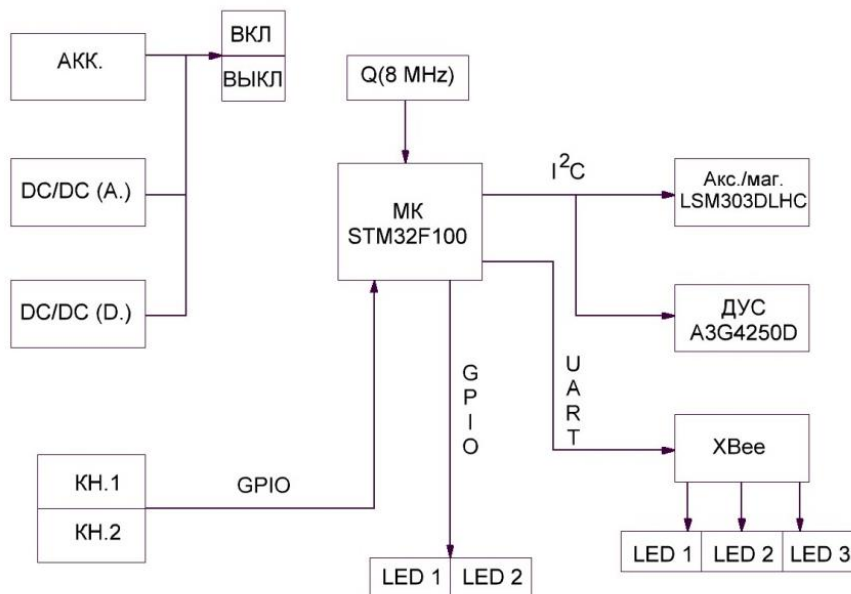


Рис.3. Функциональная схема макета измерительного модуля

Электропитание потребителей макета измерительного модуля осуществляется от аккумулятора двумя преобразователями напряжения питания, один – для аналоговой части схемы (DC/DC (A)), другой – для преобразования цифровой части (DC/DC (D)).

Для стабилизации частоты работы микроконтроллера используется кварцевый резонатор Q с частотой 8 МГц.

Микроконтроллер по интерфейсу GPIO связан с индикаторами LED1, LED2, выполняющими служебные функции на макете измерительного модуля. Индикаторы необходимы для технической отладки схемы и устранения неполадок во время работы.

В целях снижения массы макетной системы и отказа от кабельных соединений между модулями может быть применен современный радиомодуль XBee. По этому каналу передается информация от датчиков, обработанная микроконтроллером измерительного модуля. Радиомодуль также оснащен служебными индикаторами для обнаружения возможных неполадок в схеме.



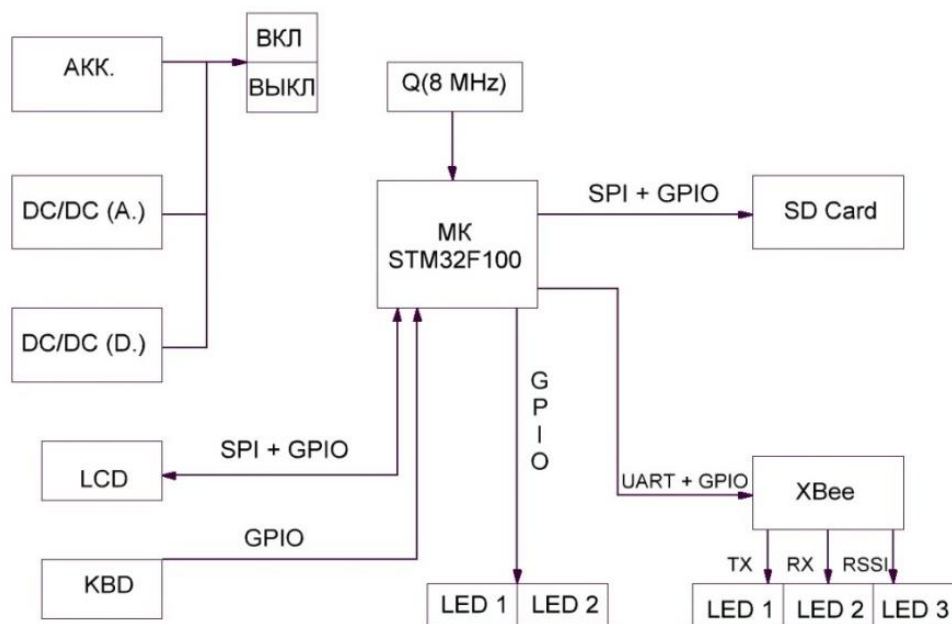


Рис.4.Функциональная схема базового модуля

Основным элементом системы измерения центра масс является базовый блок (рис.4). Для оптимизации работы схемы и индикации в блоке присутствуют дисплей и клавиатура (LCD, KBD), связанные с микроконтроллером интерфейсами SPI и GPIO. Хранилищем памяти в схеме служит SD-карта, с помощью которой станет возможна считывание и обработка информации на вычислительной машине (компьютере).

На схеме присутствуют индикаторы, помогающие обнаружить проблемы при отладке работы радиомодуля. Используются соответственно для выводов приема, передачи и усиления поступающих сигналов.

Электроснабжение макета базового модуля осуществляется аналогично электроснабжению измерительного модуля от аккумулятора и преобразователей напряжений. Использование аккумуляторов, а не связей с бортовой системой электроснабжения самолета обусловлено разработкой макетного варианта рассматриваемой системы для проведения летных исследований на борту воздушного судна – летающей лаборатории.

Выводы:

1. Создание автономной системы определения положения центра масс самолета в полете следует признать возможным.
2. Предлагаемые технические решения макетов измерительных модулей и базового модуля реализуемы на практике.
3. Предложения о проведении соответствующей научно-исследовательской работы могут быть представлены в профильные предприятия и организации Министерства промышленности и торговли РФ.

#### Библиографический список

1. И.В.Остославский, И.В.Стражева О формировании контура управления самолетом. М.: Наука, 1960. 95 с.
2. Г.М.Кошин, Г.И.Федоренко Автоматическое управление продольным движением упругого самолета. М.: Машиностроение, 1974. 310 с.
3. Roy Langton, Chuck Clark, Martin Hewitt, Lonnie Richards Aircraft Fuel Systems 345 с.
4. К.С.Колесников, В.Н.Сухов Упругий летательный аппарат как объект автоматического управления. М.: Машиностроение, 1974. 265 с.

УДК629.7.04

**Р. П. Дуркин**

студент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**А. П. Григорьев**

ассистент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**В. Я. Мамаев**

доц., канд. техн. наук – научный руководитель

### РАЗРАБОТКА УЧЕБНОГО СТЕНДА ДЛЯ ИССЛЕДОВАНИЯ ТЕХНОЛОГИЙ БЕСПРОВОДНОЙ ПЕРЕДАЧИ ДАННЫХ

Беспроводные системы по сбору данных на борту летательного аппарата в настоящее время не нашли широкого применения, но намечаются тенденции к актуализации их использования за счет потенциальной возможности организации простоты контроля и обслуживания[1].

Для исследования технологий беспроводной передачи данных необходима разработка учебного стенда, при проектировании которого должны решаться следующие частные задачи:

- анализ электромагнитной совместимости[2];
- выбор оптимального стандарта беспроводной передачи данных для реализации в рамках учебного стенда;
- выбор беспроводных датчиков, работающих по выбранному стандарту, не создающих возможных помех для других устройств на летательном аппарате [2];
- подбор элементов, реализующих прием, обработку и индикацию информации.

При реализации стенда должны быть выбраны такие беспроводные датчики, частотный диапазон которых не будет «пересекаться» с рабочими частотами различных систем-потребителей летательного аппарата (табл. 1).

Частотные характеристики различных систем

Частоты, МГц	Системы-потребители
124.800	Частота службы авиационной безопасности и управления полетами Московского центра управления воздушным движением
129.800	Авиадиспетчеры
130.000	Гражданская авиация
131.900	Диапазон для воздушной, подвижной и навигационной служб
163-165	Внутриаэродромная и технологическая связь
960-1215	Частоты, выделенные для аэродромных DME маяков
1575	GPS
1227	
1602	ГЛОНАСС
1615	

Существует множество стандартов беспроводных сетей, однако их условно можно разделить на три группы (рис. 1):

- WPAN – беспроводная персональная сеть;
- WLAN – беспроводная локальная сеть;
- WMAN – беспроводная сеть масштаба города.

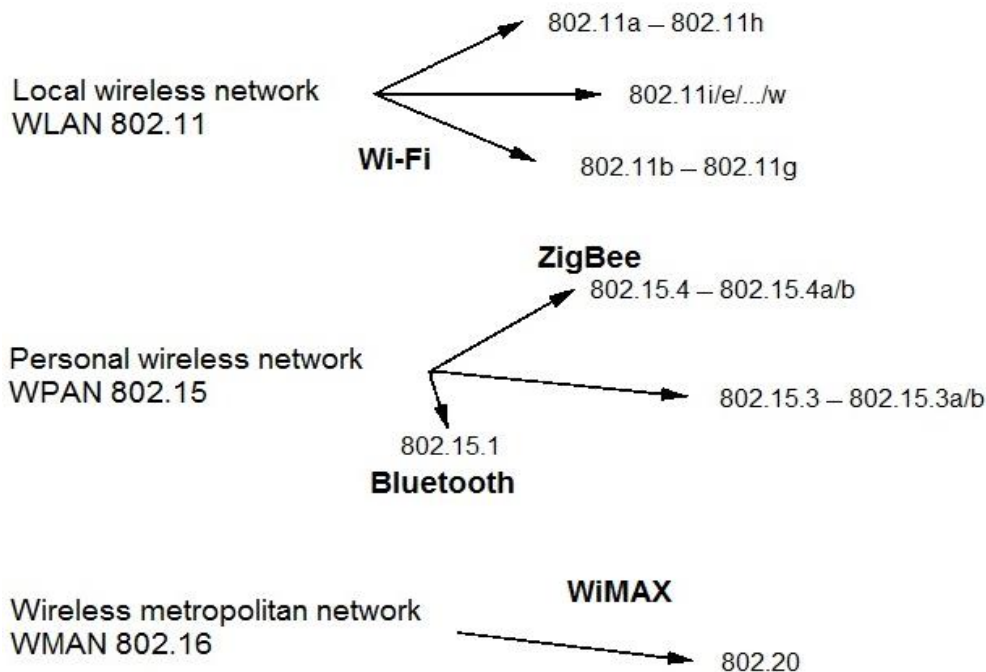


Рис. 1. Основные стандарты беспроводных сетей

Для беспроводных датчиков автономность является одним из главных параметров. На этот параметр влияет скорость передачи данных от датчика к приемнику, чем она ниже, тем ниже энергопотребление, следовательно, при разработке учебного стенда следует акцентировать внимание на стандартах группы WPAN, так как они рассчитаны на низкоскоростные сети.

WPAN (Wireless Personal Area Network) – беспроводная сеть, предназначенная для организации беспроводной связи между различного типа устройствами на ограниченной площади. Стандарты, определяющие методы функционирования сети WPAN разработаны рабочей группой IEEE 802.15.

При анализе стандартов беспроводных сетей (в табл. 2) было выявлено, что наиболее предпочтительными являются стандарты 802.15.4 и ZigBee.

Стандарт ZigBee предусматривается под достаточно небольшие скорости передачи данных, частота данного стандарта не совпадает с авиачастотами частотами спутников GPS и ГЛОНАСС.

Сравнение стандартов беспроводных сетей

Стандарт	802.15.3	802.15.4 и ZigBee	Bluetooth (802.15.1)
Скорость передачи, Мб/с	55-480	0.025-0.25	1-3
Диапазон, м	До 100	120 (в помещении – до40)	10-30
Частота, ГГц	2.4		
Описание	Разрабатывался как высокоскоростной стандарт WPAN-сетей для высокотехнологичных бытовых устройств (предназначенных, как правило, для	802.15.4 и ZigBee часто отождествляются, так как в основе стандарта ZigBee лежит стандарт 802.15.4. Предусматривают большую область применения,	На сегодняшний день хорошо развит и применяется для связи мобильных телефонов, КПК, периферии. Однако он не рассчитан на сети с низким энер-

	передачи мультимедийных данных).	а не «заточенность» под конкретные цели.	гопотреблением
--	----------------------------------	--	----------------

В рамках разработки учебного стенда (рис. 2) [4] предлагается заменить термометр ТВ-19 (рис. 3), предназначенный для измерения температуры воздуха в кабине или отапливаемых отсеках летательного аппарата на беспроводной измеритель температуры схожий по своим характеристикам. Термометр ТВ-19 (см. рис. 3, а) используется на борту АН-26 [3] (состоит из указателя и двух датчиков).

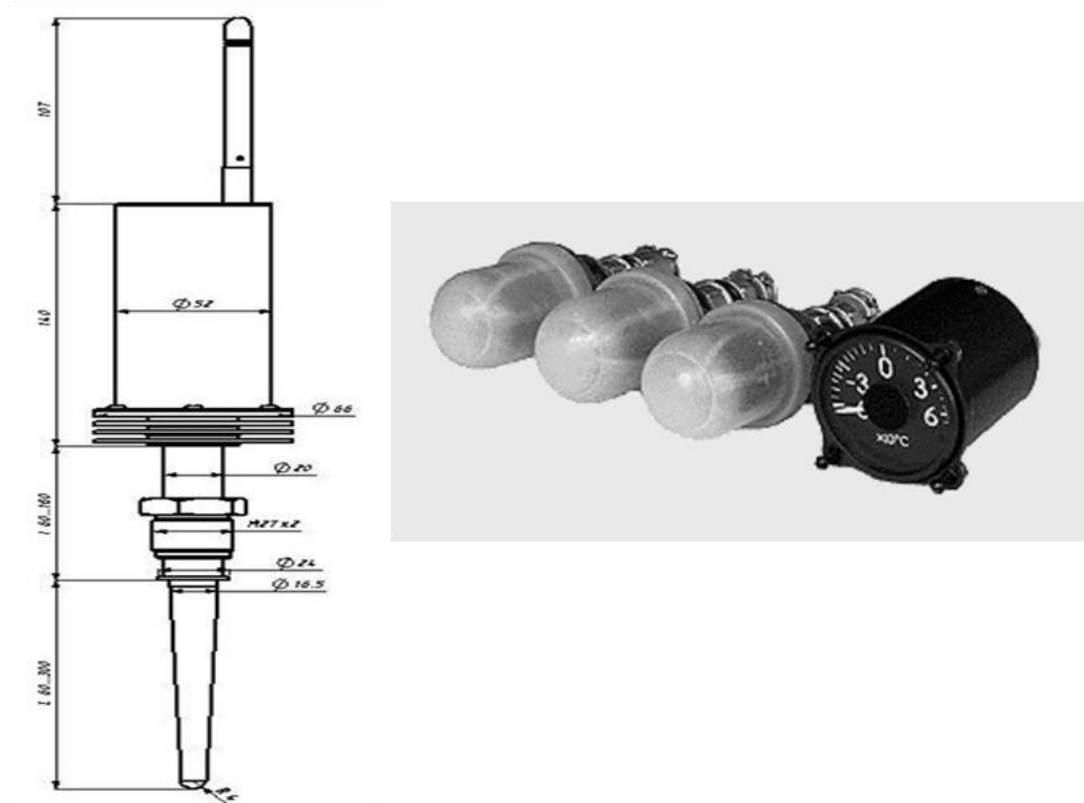


Рис. 3. Измерители температуры: а – термометр ТВ-19; б – беспроводной преобразователь температуры ПТ 9212/ZB

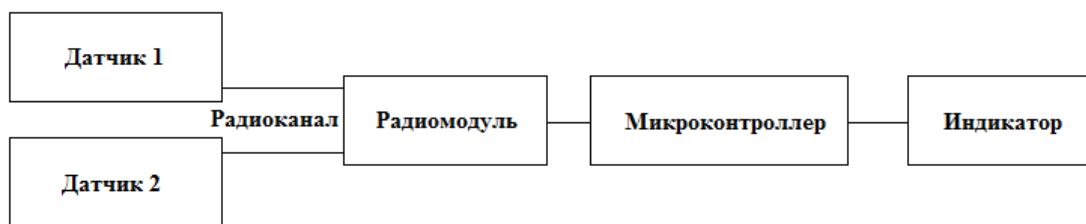


Рис.2. Структурная схема учебного стенда

Для решения данной задачи выбраны преобразователи температуры ПТ 9212/ZB (см. рис. 2, б), предназначенные для измерения и непрерывного преобразования температуры в цифровой сигнал по радиоканалу ZigBee.

Внешний вид комплектующих стенда представлен на рис. 4.

Беспроводные технологии в области систем передачи и сбора данных в перспективе могут быть адаптированы к различным областям применения [1]. Примерами таких технологий являют-

ся сенсорные сети и интеллектуальные системы обслуживания. Беспроводная сенсорная сеть – это распределенная, самоорганизующаяся сеть множества датчиков (рис. 5), объединенных между собой посредством радиоканала. Беспроводные сенсорные сети можно применить для измерения и контроля таких параметров как вибрация, температура, давление и т.п.

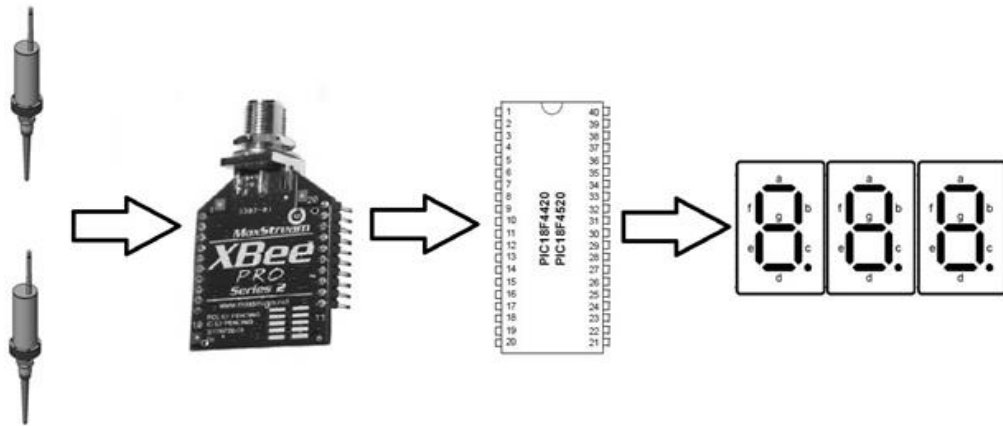


Рис. 4. Внешний вид комплектующих учебного стенда

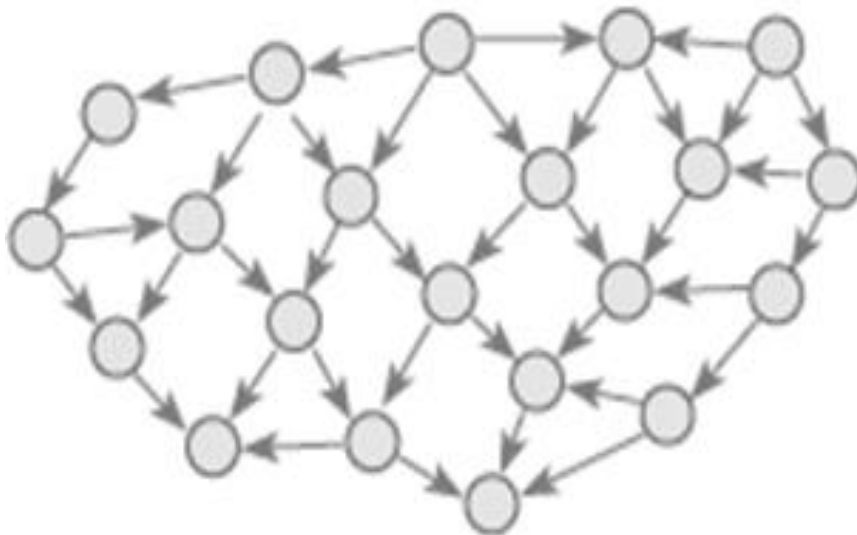


Рис.5. Беспроводные сенсоры

Интеллектуальная система обслуживания на основе технологий беспроводных датчиков и генерации электричества из тепла может заблаговременно предупредить о чрезмерном износе деталей без необходимости разборки двигателя и тщательного изучения его узлов.

#### Библиографический список

1. Шахнович И.А. Современные технологии беспроводной связи. М.: Техносфера, 2006. 288 с.
2. ГОСТ Р 50397-2011 Совместимость технических средств электромагнитная. Термины и определения. Взамен ГОСТ Р 50397-92; введ. 08.12.2011. М.: Стандартинформ, 2011. 61 с.
3. А. С. Альбац, В. Г. Бабий, А. В. Баркар и др. Самолёт Ан-26. Техническое описание / 2-е изд. М.: Авиаэкспорт, 1970. 302 с. (Книга 2. Конструкция самолёта).
4. Волочич Г. И. Схемотехника аналоговых и аналого-цифровых электронных устройств. – М.: Издательский дом "Додека XXI", 2005. 528 с.

УДК 004.021

**А. С. Евдокимов**

студент кафедры аэрокосмических компьютерных и программных систем

**К. Н. Рождественская**

ассистент каф 14 – научный руководитель

## ОБЗОР АЛГОРИТМОВ ПОИСКА МЕСТА ПОДКЛЮЧЕНИЯ МЕНЕДЖЕРОВ PLUG-AND-PLAY ДЛЯ СЕТИ SPACEWIRE

SpaceWire-Plug-and-Play (PnP) – технология – это настройка сети SpaceWire без участия человека, т. е. устройства сами по специальному алгоритму исследуют и настраивают друг друга. Для этого необходимо наличие в сети некоторый «умных» устройств, которые будут исполнять алгоритмы PnP. Технологию Plug-and-Play можно разделить на два режима работы: централизованный и децентрализованный. В зависимости от целей и задач будет изменяться количество необходимых менеджеров, которые должны присутствовать и функционировать в сети. Так, при централизованном PnP в сети присутствует только один менеджер, который и осуществляет конфигурацию и администрирование сети. В децентрализованном режиме таких менеджеров несколько. Но в первом и во втором случае необходимо выбрать место подключения менеджеров. Для централизованного режима необходимо, чтобы менеджер располагался в некотором центре сети, которая должна поддерживаться технологией PnP. Данное решение уже было реализовано – это поиск медианы графа. В нем сеть представляется в виде графа и ищется оптимальная точка подключения менеджера – наилучшая по выбранному критерию минимизации суммы расстояний, другими словами центр – это устройство сети, сумма расстояний от него до всех других устройств сети минимальна.

Для децентрализованного PnP задачи стоят несколько сложнее, так как в сети должно присутствовать несколько центров сети:

1. Сеть должна обеспечивать гарантированную работоспособность, т. е. должна выполнять свои функции даже при выходе из строя одного или нескольких узлов коммутации. Одно из главных ограничений к структуре: сеть не должна иметь древовидный характер так как, это неблагоприятно сказывается на ее работоспособности: древовидная структура без дополнительных связей может привести к потере части сети.

2. Также желательно, чтобы регионы были связаны друг с другом как минимум несколькими связями, идущими из разных узлов, дабы при выходе из строя одного или нескольких из них связь между регионами сохранялась. Но не стоит забывать про связность внутри самих регионов, там также должно быть достаточное количество каналов между узлами.

3. Сеть необходимо делить на регионы так, чтобы выход из строя связи(канала) или узла не приводил к потере управления сетью и нарушению ее функционирования.

4. Количество узлов сети может быть произвольным. Некоторые узлы должны обслуживаться одним менеджером в силу поставленных перед ними задач и целей.

5. Сеть должна разбиваться на несколько регионов (в зависимости от количества менеджеров и размеров сети). В случае, если менеджер способен администрировать всю заданную сеть целиком (его способность определяется техническими характеристиками, такими как: объем памяти, производительность, пропускная способность и пр.), то количество регионов равно единице, это случай централизованного администрирования сети. И обратный случай, децентрализованного администрирования, когда исходная сеть разбивается на несколько регионов, каждому из которых сопоставляется один менеджер. Повышение отказоустойчивости может быть реализовано путем добавления дублирующего менеджера, либо же алгоритмически (перераспределение администрируемых устройств в ходе штатной работы сети). Вопросы отказоустойчивости не рассматриваются в данной статье.

6. В каждом регионе количество устройств должно превышать единицу. Очевидно, что регион целесообразно составлять из нескольких устройств, иначе в сети будут возможны регионы, состоящие всего из одного узла, и каждому такому региону будет присвоен менеджер, выполняющий функции PnP, что не эффективно.

Постановка общей задачи при поиске места подключения центров сети при децентрализованном PnP может быть сформулирована следующим образом: необходимо заданную сеть разделить на регионы по количеству устройств, выполняющих функции PnP. Что из себя представляет сеть на выходе и какие требования предъявляются к структуре сети обусловлено поставленными выше задачами.

Для решения описанной задачи удобно представить сеть с точки зрения графа: вершинам графа соответствуют устройства сети, а дугам – каналы связи между устройствами.

Проведен обзор алгоритмов теории графов для решения описанной задачи распределения менеджеров в сети для децентрализованного PnP. Существует ряд решений, которые подходят для поставленной задачи, однако, большинство из них в немодифицированном виде не подходят под все требования к структуре сети. Таким образом, полученное разделение после алгоритма следует скорректировать согласно поставленным выше задачам.

Ниже приведены алгоритмы и методы, которые были рассмотрены для решения задачи распределения менеджеров при децентрализованном PnP.

1. Метод Мальгранжа [1]. Самый простой для реализации на ЭВМ метод, основанный на работе с матрицами. Но он не способен выдать заданное число подграфов. Самым большим минусом данного метода является то, что в итоге деления возможны случаи одиночных подграфов, то есть графов, состоящих из одной вершины, что не приемлемо в рамках нашей задачи.

2. Метод рекурсивного деления пополам [2]. В отличие от предыдущего алгоритма, с помощью данного метода имеется возможность получить заданное число подграфов. Однако он никак не учитывает очень важный для решения поставленной задачи параметр – связность, а потому он тоже не подходит.

3. Матричный метод разбиения [1]. Получаемое разбиение идентично результату метода Мальгранжа. Матричный метод разбиения имеет такие же положительные стороны, как и этот метод (Мальгранжа): он относительно просто реализуется на ЭВМ, так как работает с матрицами. Но и минусы у него те же: нет контроля количества и размера выдаваемых подграфов, а также, существует возможность появления одиночных подграфов. Поэтому этот метод не подходит для решения поставленной задачи по тем же причинам, что и метод Мальгранжа.

4. Далее рассмотрим комбинаторные методы. Комбинаторные методы обычно обеспечивают более сбалансированное разбиение и меньшее информационное взаимодействие полученных подсетей. Однако они имеют тенденцию работать существенно дольше.

4.1. Алгоритм деления с учетом связности [3]. Данный алгоритм обладает длинным рядом плюсов: алгоритм учитывает связность, относительно прост в реализации, а также существует возможность деления на требуемое количество подграфов. Существуют модификации, позволяющие контролировать и размер получаемых подграфов, однако полученное алгоритмом разбиение далеко от оптимального, т. е. качество деления – единственный существенный минус данного алгоритма, из-за которого было принято решение не использовать его для решения поставленной задачи.

4.2. Алгоритм возрастающего графа [3]. Недетерминированный по своей природе алгоритм. Довольно простой метод деления графа, но не самый легкий для реализации на ЭВМ. В чистом виде не эффективен и требует неоднократного повторения деления графа, однако, существуют модификации, позволяющие избавиться от данного недостатка. Данный алгоритм позволяет делить граф на заданное количество частей, а также он учитывает связность.

4.3. Алгоритм Кернигана – Лина [3]. Алгоритм хорош тем, что учитывает связность и при достаточном количестве итераций разбиение близко к оптимальному. Еще одним плюсом данного алгоритма является возможность регулировать количество получаемых подграфов. К сожалению

нию, у алгоритма также есть и минус: время выполнения алгоритма для больших сетей с большой точностью будет велико, поэтому на практике используют модифицированный алгоритм.

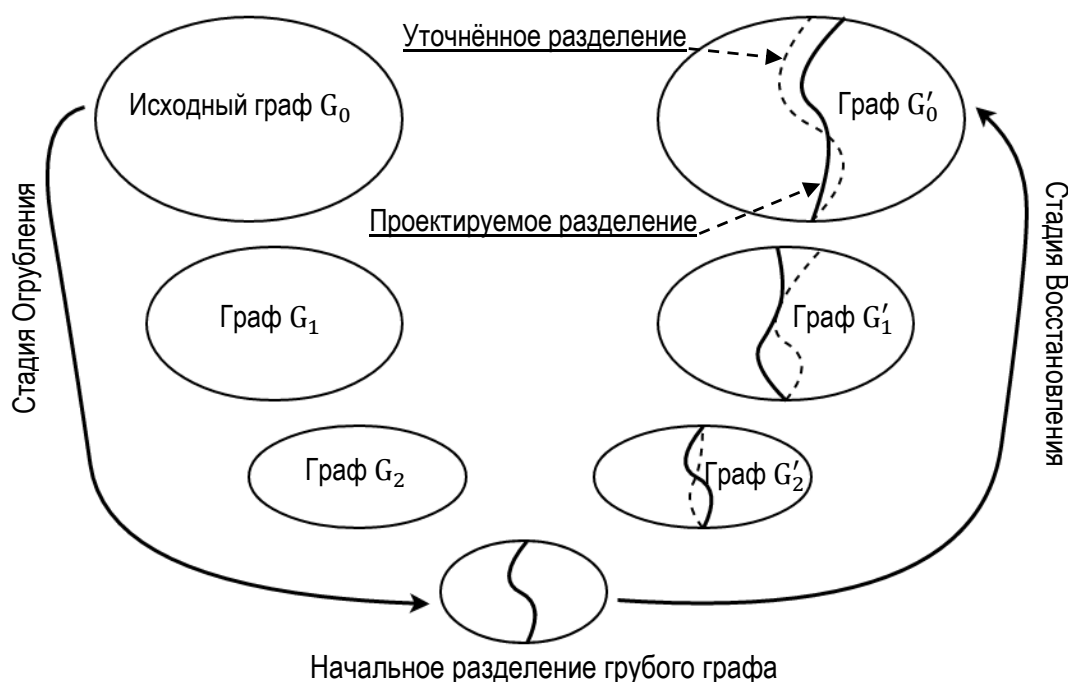
Помимо представленных выше решений, в теории графов существует еще множество различных алгоритмов, например, целый класс геометрических алгоритмов, но они не подходят для решения поставленной задачи. Главной причиной является то, что все эти методы разбиения графа на подграфы не учитывают важный для нас параметр – связность. Также, геометрические методы требуют дополнительной информации для исходной сети: координаты каждой компоненты сети.

Самое качественное из всех рассмотренных выше алгоритмов разбиение способен дать комбинаторный алгоритм Кернигана–Лина, и именно он больше всего подходит для решения поставленной задачи. Но в чистом, не модифицированном виде, алгоритм Кернигана-Лина будет не эффективен, а при большом количестве узлов в сети – слишком затратным. Так как потребуются много итераций, нужно либо огромные вычислительные мощности, либо большое количество времени. Для усовершенствования алгоритма можно применить алгоритм многоуровневой бисекции графа.

Многоуровневый алгоритм Кернигана-Лина состоит из трех стадий [3]:

1. стадия огрубления;
2. стадия разделения;
3. стадия восстановления.

Ниже, на рисунке, показана приблизительная схема работы многоуровневого алгоритма деления графа пополам, на котором отображены все стадии.



Рассмотрим подробнее стадии алгоритма, представленные на рисунке. На стадии огрубления строится последовательность меньших графов, каждый с меньшим количеством узлов. Более грубый граф можно получить, стягивая смежные узлы. Таким образом, ребро между двумя узлами удаляется и создается мультиузел, состоящий из этих двух узлов. Огрубление графа может быть достигнуто различными способами, но в данной статье будет рассматриваться только один из них –случайное паросочетание (RM).

Основная идея случайного паросочетания состоит в том, что узлы посещаются в случайном порядке. Если узел  $u$  не был включен в паросочетание, то алгоритм беспорядочно выбирает один из его смежных узлов, который также не включен в паросочетание. Если такой узел  $v$ . су-



ществует, то алгоритм включает ребро  $(u, v)$ . в паросочетание и маркирует узлы  $u$ . и  $v$ . как посещенные. Если не имеется никакого немаркированного смежного узла  $v$ , то узел  $u$  остается свободным и переходит в следующий граф. Сложность такого алгоритма –  $O(E)$  . .

Необходимость в данной стадии огрубления обуславливается тем, что изначальная сеть может иметь большой размер и именно добавление данной стадии в алгоритм избавит от одного из минусов немодифицированного алгоритма Кернигана-Лина – неэффективность при большом количестве узлов сети.

Далее идет стадия разделения, на ней можно применять все ранее рассмотренные алгоритмы разбиения графа, но, так как из всех прочих алгоов выделен именно алгоритм Кернигана-Лина, то именно он будет описан ниже. Вторая стадия многоуровневого алгоритма – стадия разделения – вычисляет высококачественное деление пополам (сепаратор с маленьким весом)  $P_m$ . для грубого графа  $G_m = (V_m, E_m)$  так, что каждая часть содержит примерно половину узлового веса первоначального графа. Поскольку веса узлов и ребер уменьшенных графов в процессе огрубления вычислялись так, чтобы в целом не потерять существенную информацию о весе узлов и ребер большего графа,  $G_m$  содержит достаточную информацию, чтобы эффективно осуществить сбалансированное разделение.

В качестве алгоритма разделения графа пополам выбран, как уже ранее говорилось, алгоритм Керниган-Лин. Он начинается с начального разделения графа пополам. На каждой итерации он ищет поднабор узлов от каждой части графа, такой, что обмен этими поднаборами ведет к разделению с меньшим сечением. Если такие поднаборы существуют, то обмен выполняется, и это становится разделением для следующей итерации. Алгоритм продолжается, повторяя полный процесс. Если он не может найти два таких поднабора, то алгоритм заканчивается, так как для разделения достигнут местный минимум, и никакое дальнейшее усовершенствование алгоритмом Керниган-Лин не может быть сделано. Он находит локальный минимум для разделения, когда начинает с хорошего начального разделения. Если не известно хорошее начальное разделение, то этот алгоритм повторяется с различными случайно отобранными начальными разделениями и отбирается то разделение, которое дает самый маленький вес сепаратора. Требование многократного повторения вычислений, может быть достаточно обременительным, особенно если граф велик. Однако, так как на вход алгоритма разделения поступает намного меньший грубый граф, многократное выполнение требует очень небольшого времени.

Последняя стадия многоуровневого деления пополам – стадия восстановления. В ходе стадии восстановления разделение  $P_m$  более грубого графа  $G_m$  проектируется назад к первоначальному графу. Даже притом, что  $P_{i+1}$  – местное минимальное разделение  $G_{i+1}$ , проектируемое разделение  $P_i$  может не быть в местном минимуме относительно  $G_i$ . Поскольку  $G_i$  более точный граф, он имеет большее количество степеней свободы, которые можно использовать, чтобы улучшить  $P_i$  и уменьшить вес сепаратора. По этой причине после проектирования разделения используется алгоритм рафинирования разделения.

В качестве алгоритма рафинирования удобно будет выбрать описанный выше алгоритм Кернигана-Лина. Основная идея Керниган-Лин рафинирования состоит в том, чтобы использовать спроектированное разделение  $G_{i+1}$  для  $G_i$  как начальное разделение для алгоритма Керниган-Лина, описанного для стадии разделения. Причина в том, что это проектируемое разделение уже является хорошим разделением; таким образом, Керниган-Лин будет сходиться в пределах нескольких итераций к лучшему разделению.

Таким образом, пропустив исходный граф через все три стадии, на выходе данного многоуровневого алгоритма будет два наиболее связанных внутри себя подграфа. Чтобы получить большее количество подграфов следует один из полученных подграфов подать на вход алгоритма.

В заключение следует заметить, что рассмотренный в данной статье и выбранный в качестве подходящего для решения поставленных задач многоуровневый алгоритм разбиения

Керниган-Лина имеет неоспоримое преимущество перед всеми другими рассмотренными алгоритмами, но и с помощью него невозможно получить идеальный результат. Изъяном полученного на выходе разбиения является тот факт, что хоть и подграфы максимально связаны внутри себя, между собой количество связей у них минимально, поэтому выбранный алгоритм следует применять на уровне проектирования сети, и в случае критически малой связности между регионами будет существовать возможность добавления новых связей между ними. Место подключения менеджера в регионе следует выбирать также, как и при централизованном PnP – с помощью поиска медианы графа.

#### **Библиографический список**

1. Волков К. Н., Емельянов В. Н. Вычислительные технологии в задачах механики жидкости и газа: вычислительные методы и программирование. М.: Физматлит, 2013. 465 с.
2. Волченская Т.В., Князьков В.С. Компьютерная математика: Часть 2. Теория графов: учеб. пособие. Пенза: Пенз. ун-т, 2002. 301 с.
3. Бувайло Д.П., Толоч В.А. Быстрый высокопроизводительный алгоритм для разделения нерегулярных графов// Вісник Запорізького державного університету. № 2, 2002.

**УДК 629.735.33**

**И. А. Иванов**

магистрант кафедры аэрокосмических компьютерных и программных систем

**Н. А. Шехунова**

проф., канд. техн. наук – научный руководитель

### **ИСПОЛЬЗОВАНИЕ ОБЛАЧНЫХ ТЕХНОЛОГИЙ В АВТОМАТИЗИРОВАННЫХ СИСТЕМАХ ОБРАБОТКИ ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ**

Аннотация. На сегодняшний день облачные технологии являются наиболее многообещающими за счет своей гибкости, эффективности и экономической выгоды. Их использование значительно изменило традиционный подход к доставке, хранению, управлению и интеграции приложений. Это позволило более крупным структурами обслуживать несколько групп пользователей в пределах одного облака, но при этом избавило от необходимости иметь мощную систему и значительно снизило затраты на обслуживание.

В статье делается обзор облачных информационных технологий, рассмотрены основные плюсы и минусы их использования. Перечислены некоторые цели и перспективы использования облачных вычислений в автоматизированных системах обработки экспериментальных данных.

Впервые об идее облачных вычислений высказался Джон Маккарти еще в 1960 г. Он предположил, что когда-нибудь компьютерные вычисления будут производиться с помощью «общенародных утилит». Однако только в 2007 г. эта идея получила популярность. Под облачными вычислениями (от англ. cloud computing) было принято понимать предоставление пользователю вычислительных ресурсов (например, серверов, устройств хранения данных, приложений) в виде интернет-сервиса. На практике облако вычислений – это тысячи различных серверов, размещенных в дата-центрах, обеспечивающих работу десятков тысяч приложений, которые одновременно используют миллионы пользователей по всему миру.

Облачные вычисления обладают рядом функциональных характеристик, которые и формируют их уникальность:

- самообслуживание по требованию (self service on demand) позволяет потребителям определять и изменять вычислительные потребности без взаимодействия с поставщиком;
- универсальный доступ по сети позволяет получать услуги по сети передачи данных вне зависимости от используемого устройства;

- объединение ресурсов (resource pooling) позволяет обслуживать большее число потребителей благодаря динамическому распределению мощностей;
- эластичность позволяет расширять и сужать спектр услуг в любой момент времени без взаимодействия с поставщиком;
- учет потребления позволяет унифицировать потребляемые ресурсы с использованием определенного уровня абстракции: объем хранимых данных, пропускная способность, количество пользователей, количество транзакций.

Исходя из характеристик становится понятна популярность облачных технологий: возможности их применения могут быть очень разнообразны и позволят экономить на инфраструктуре и обслуживании систем, а аппаратное обеспечение может быть сильно упрощено при обработке данных и хранении информации в удаленных центрах данных. К тому же такой подход позволяет полностью стандартизировать программное обеспечение под любую операционную систему (Windows, Linux, MacOS и т.п.). Облачные технологии значительно облегчают обеспечение доступа к данным как для клиентов, так и для администраторов, находящихся вдалеке от системы, но имеющих возможность подключиться через Интернет.

К тому же стоит отметить, что современные облачные технологии уже далеко вышли за границы сетевого и серверного оборудования, постепенно проникают на рынок встраиваемых систем (embedded cloud) и становятся причиной масштабной реструктуризации любого вида сервисов. Внедрение встраиваемых систем позволило управлять работой устройств, сбором данных и обеспечением интерактивных возможностей посредством подключения к компьютерной сети.

Подобная модель хранения и вычисления данных очень перспективна в автоматизированных системах обработки данных. Вычислительные системы, осуществляющие машинную обработку результатов измерения параметров объекта (или явления), могут требовать больших ресурсов. Однако далеко не всегда точно можно определить, каких и на какой срок. К тому же, подобные системы обеспечивают обмен информацией с экспериментатором в процессе функционирования и берут на себя задачи по представлению данных.

Учитывая специфику автоматизированных систем обработки экспериментальных данных можно выделить следующие достоинства использования облачных сервисов:

- возможность работать с данными на любом устройстве – абсолютно не важно, какой производительностью обладает гаджет, на котором будет запущена система, результат отображения зависит только от наличия интернета;
- возможность работать с данными на любой архитектуре – системы обработки данных представляются как веб-сервисы и работают в браузере любой операционной системы;
- работать с системой могут несколько пользователей – одну и ту же информацию могут просматривать и редактировать одновременно несколько пользователей с различных устройств;
- все данные хранятся в «облаке» – таким образом система полностью защищена от потери данных в связи с выходом устройства из строя;
- данные в системе всегда актуальны;
- пользователю всегда доступна самая последняя версия программного продукта – нет необходимости следить за выходом обновлений;
- возможность делиться информацией с научной группой из любой точки земного шара;
- надежность облачных сервисов обеспечивается тем, что программно-аппаратная платформа находится в специализированных Центрах обработки данных (ЦОД), которые включают резервирование сетевой и аппаратной инфраструктуры.

При создании систем обработки данных решается целый круг задач – сбор данных, построение хранилища, расчет и визуализация. Оперативно собрать необходимые данные для анализа и их обработки это только полдела, так как просто набор данных не представляет особого интереса и не может отобразить полной картины исследования. Облачные сервисы позволяют

получить информацию, отражающую наиболее полные сведения, и удобно ее представить всем пользователям. Это значительно повышает эффективность анализа.

Еще одним важным достоинством хранения данных в облаке является надежность. Почти все поставщики облачных ресурсов делают неоднократное резервное копирование данных. Например, в Azure любой элемент данных сохраняется как минимум три раза независимо от того, что это за данные. К тому же поставщики берут на себя все задачи восстановления данных внутри ЦОДа. Для обеспечения еще большей надежности в Azure три копии данных сохраняются в другом ЦОД, который расположен на расстоянии как минимум в 400 миль от первого на случай, если один дата-центр полностью выйдет из строя.

Единственным слабым местом использования облачных технологий является безопасность информации. Однако поставщики облачных услуг относятся к защите данных очень серьезно и постоянно развивают эту сферу. Но если хочется обеспечить максимальный уровень безопасности данных и быть полностью уверенным в том, что кроме владельца никто не сможет получить к ним доступ, то необходимо шифровать данные перед загрузкой в облачный сервис. При простом использовании систем обработки данных результаты также шифруются. Поэтому, это даже не решение проблемы, а необходимость. Однако, эта проблема присуща только для публичных облаков. Корпоративные облачные сервисы предусматривают безопасные каналы передачи данных, защищенные шифрованием или вовсе не имеющие стыков с сетью Интернет.

Одним из примеров использования облачных вычислений в автоматизированных системах обработки данных может стать разработка сервиса для обработки результатов экспериментального исследования помехоустойчивости многоуровневой флэш-памяти, проводимой различными коллективами исследователей. Обработка данных в «облаке» позволит реализовать совместную работу сразу нескольких исследовательских лабораторий в онлайн режиме и максимально сэкономить на реализации проекта, а также обеспечить высокую надежность хранения данных.

### Библиографический список

1. *Медведев А.* Облачные технологии: тенденции развития, примеры исполнения // Современные технологии автоматизации. 2013. № 2. С. 6–9.
2. Облачные вычисления (Cloud computing). 2012. URL: [http://www.tadviser.ru/index.php/Статья:Облачные\\_вычисления\\_\(Cloud\\_computing\)](http://www.tadviser.ru/index.php/Статья:Облачные_вычисления_(Cloud_computing)).
3. *Орландо Д.* Модели сервисов облачных вычислений: программное обеспечение как сервис. 2012.

УДК 681.51

**С. А. Иванова**

магистрант кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**Ю. П. Иванов**

канд. техн. наук, доцент – научный руководитель

## ИССЛЕДОВАНИЕ АЛГОРИТМОВ И РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ФИЛЬТРАЦИИ И КЛАССИФИКАЦИИ НАВИГАЦИОННЫХ СИГНАЛОВ

В настоящее время создание средств автоматизации расчетов является актуальной задачей. В учебной деятельности такие средства могут применяться не только для получения информации, но и для проверки знаний или правильности ручных расчетов. В Санкт-Петербургском государственном университете аэрокосмического приборостроения на кафедре аэрокосмических приборов и измерительно-вычислительных комплексов для выполнения лабораторных и курсовых работ по дисциплинам «Информационно-статистическая теория измерений» и «Комплексирование» студенты пользуются набором программ, созданным более

двадцати лет назад. Со временем корректная работа этих программ без установки дополнительного программного обеспечения (ПО) на большинстве компьютеров стала невозможной из-за несовместимости с современными версиями операционной системы Windows. Более того, эти программы не связаны между собой и отсутствует руководство по их использованию, что не позволяет студенту самостоятельно разобраться в особенностях их применения. В связи с этим потребовалось разработать новый программный продукт, специально предназначенный для изучения методов обработки измерительной информации с целью повышения уровня знаний студентов.

На основе широкого класса алгоритмов фильтрации и классификации навигационных сигналов реализовано ПО в виде приложения, работающего в последних версиях операционной системы Windows. Данное ПО было разработано в среде Microsoft Visual Basic 6.0 на языке программирования Visual Basic. К основным возможностям этого языка можно отнести реализацию гибкого и удобного интерфейса для приложения, обработку событий мыши и клавиатуры, работу с графикой, обработку файлов и каталогов на жестком диске, а также возможность внедрения в приложение различных объектов, поддерживаемых другими программами.

С помощью разработанной программы можно проводить сравнительный анализ различных видов алгоритмов фильтрации классификации сигналов, а также производить рациональный выбор наиболее подходящего из них для дальнейшего использования. На данный момент в ПО реализованы линейные оптимальные и оптимально-инвариантные алгоритмы оценки полезного сигнала комплексной системы, состоящей из двух измерителей, и классификации сигналов по критериям В. А. Котельникова и Неймана-Пирсона. В разработанном ПО решаются задачи синтеза и анализа алгоритма оценки полезного сигнала измерительной системы. С помощью программы можно анализировать не только качественные характеристики линейных безынерционных и инерционных комплексных систем, но и такие свойства, как робастность и помехозащищенность.

В этой работе измерения описываются линейной моделью с аддитивной погрешностью, некоррелированной с полезным сигналом [1]:

$$Y_i(t) = R_i X(t) + H_i(t),$$

где  $i = 1, 2$  – номер канала измерения;  $Y(t)$  – вектор результатов измерений с размерностью  $2 \times 1$ ;  $X(t)$  – значение полезного сигнала (ПС) в момент времени  $t$ ;  $R_i$  – вектор масштабных коэффициентов с размерностью  $2 \times 1$ ;  $H(t)$  – вектор аддитивных погрешностей измерений (ПИ) размерностью  $2 \times 1$ .

Исходными данными для синтеза оценки полезного сигнала в общем случае являются моменты полезного сигнала и помехи, а для классификации – пороговые значения входного сигнала и законы распределения полезного сигнала и помехи. В качестве критерия оптимальности или показателя качества системы выбран критерий минимума среднего квадрата ошибки оценки.

На данный момент в разработанном ПО реализованы линейные безынерционные алгоритмы оценки сигналов, которые можно проклассифицировать в соответствии со схемой, приведенной на рис. 1. Указанные разновидности безынерционных алгоритмов применены для получения оценки полезного сигнала в комплексной системе, состоящей из двух измерителей и контрольной аппаратуры (КА) с заданными вероятностными характеристиками. По результатам показаний КА принимается двувальтернативное решение об исправности или отказе измерителя. С помощью разработанной программы пользователь может рассмотреть результаты синтеза и анализа безынерционных линейных алгоритмов с учетом показаний реальной КА или идеальной КА при идеальных измерителях, а также с учетом надежности измерителей по критерию квазиэффективной точности.

На рис. 2 представлены сочетания видов корреляционных функций стационарных погрешностей измерителей комплексной системы, для которых в программе реализованы линейные инерционные алгоритмы фильтрации Винера. Важно отметить, что инерционную оценку сигнала можно получить не только для комплексной системы, но и для одного измерителя, рассматривая

корреляционную функцию погрешности одного из измерителей как корреляционную функцию помехи. В дальнейшем планируется реализовать в ПО алгоритмы фильтрации Калмана для тех же сочетаний корреляционных функций погрешностей измерителей и спектрально-финитные алгоритмы, универсальные для любых сочетаний видов корреляционных функций.



Рис. 1. Классификация линейных безынерционных алгоритмов оценки сигналов, реализованных в ПО



Рис. 2. Классификация линейных инерционных алгоритмов оценки сигналов, реализованных в ПО

Помимо синтеза и анализа оценки полезного сигнала по выбранному алгоритму в приложении предусмотрено выполнение классификации сигналов измерительной системы. Основной задачей классификации является нахождение оптимального изменения контрольного допуска на полезный сигнал по отношению к области допустимых значений по критериям В.А. Котельникова и Неймана-Пирсона [2]. В программе реализованы алгоритмы классификации сигналов для нормально-нормального, равномерно-нормального и нормально-равномерного законов распределения полезного сигнала и помехи. Проанализировать результаты классификации можно по графикам зависимостей рисков заказчика и изготовителя, достоверностей каналов измерений, вероятностей нахождения сигналов системы в области допустимых значений от изменения контрольного допуска на полезный сигнал, систематической погрешности измерений, среднего квадратичного отклонения полезного сигнала.

В процессе разработки данного программного продукта решались не только задачи предметной области, но и реализации пользовательского интерфейса. На данный момент его основными особенностями является контроль ввода исходных данных и наличие системы подсказок для исправления ошибок ввода. Помимо этого есть возможность сохранения введенных исходных данных при завершении работы с приложением и повторного их использования при запуске программы. Результаты работы программы представлены в удобной для восприятия форме: в виде графиков и таблиц. В дальнейшем планируется снабдить разрабатываемый пакет руководством и подсказками по его использованию, а также теоретическими материалами по алгоритмам обработки измерительной информации. Это позволит студентам самостоятельно разбираться в работе программы без консультации преподавателя и применять её для решения задач, связанных с обработкой сигналов.

В заключение следует отметить, что разработанное ПО объединяет в себе часть алгоритмов синтеза и анализа инерционной и линейной безынерционной оценок, а также квазиоптимальной классификации параметров сигналов измерительной системы. Внедрение программы в учебный процесс позволит студенту не только экономить время на расчетах, но и сравнивать различные виды алгоритмов обработки сигналов между собой по показателям качества, робастности и помехозащищенности измерительной системы, а затем определять, какой из них является наиболее подходящим для использования при заданных исходных данных. Таким образом пользователь получает более полное представление о преимуществах и недостатках каждого из рассмотренных им алгоритмов и в дальнейшем сможет применять полученные знания для решения более сложных задач.

#### **Библиографический список**

1. Иванов Ю. П., Никитин В. Г. Информационно-статистическая теория измерений. Методы оптимального синтеза информационно-измерительных систем, критерии оптимизации и свойства оценок: учеб. пособие. СПб.: ГУАП: 2011. 104 с.
2. Иванов Ю. П., Никитин В. Г., Чернов В. Ю. Контроль и диагностика измерительно-вычислительных комплексов: учеб. пособие. СПб.: ГУАП: 2002. 120 с.

**УДК 629.733.5:614.84**

**Д. Г. Колесников**

студент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**А. П. Григорьев**

ассистент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**В. Я. Мамеев**

доц., канд. техн. наук – научный руководитель

### **АКТУАЛЬНЫЕ ВОПРОСЫ ПРИМЕНЕНИЯ ДИРИЖАБЛЕЙ В ЗАДАЧАХ ПОЖАРОТУШЕНИЯ И ОСАЖИВАНИЯ ПЕПЛА ПРИ ИЗВЕРЖЕНИИ ВУЛКАНОВ**

21 марта 2010 года в Исландии произошло извержение вулкана Эйяфьядлайёкюдль, повлекшее экологические проблемы, связанные с гибелью животных (преимущественно птиц) и техногенным загрязнением воздуха пыле-пепельными образованиями и различными газами (в том числе ядовитыми и радиоактивными). В ходе данного катаклизма в верхние слои атмосферы (до высоты 11 км) были выброшены мелкие частицы вулканического стекла и кварца, попадание которых в авиационные двигатели потенциально может приводить к потере мощности, помпажу и разрушению. Сформированные в результате извержения облака вулканического пепла более чем на неделю нарушили авиасообщение в Северной Европе и повлекли отмену более чем 63 тыс. регулярных рейсов, способствовали нарушению регулярного транспортного сообщения между странами и частичному транспортному коллапсу. Так как ежегодно в среднем проис-

ходит одно крупное извержение вулкана, то данная проблема является достаточно насущной и актуальной.

Целью данной статьи является рассмотрение методов пожаротушения и осаживание пепла при извержении вулканов, а также анализ потенциальной возможности применения дирижаблей для этих целей.

Дирижабль (от фр. *dirigeable* – управляемый) – управляемое воздушное судно (управляемый аэростат) с двигателем (двигателями), у которого большая часть или весь вес уравновешивается аэростатической подъемной силой.

Дирижабль принадлежит к типу воздушных судов легче воздуха и получает подъемную силу с помощью наполнения его газом, более легким, чем воздух. Подъем, спуск, а также сохранение устойчивого положения в воздухе происходят согласно закону Архимеда, который гласит, что всякое тело, погруженное в жидкость или газ, испытывает давление, вытесняющее его вверх и равное весу жидкости или газа (воздуха) в объеме этого тела [1].

В строительстве дирижаблей установились три отличных друг от друга системы: жесткая, полужесткая и мягкая [1].

Жесткие – обычно крупные дирижабли – имеют жесткий каркас, разделенный на ряд газовых отделений (отсеков), внутри которых помещаются отдельные баллоны с газом. Каркас обтягивается полотном (за последнее время стал применяться листовая дюраль).

Полужесткие – имеют мягкую оболочку и при ней жесткую опору в виде специальной припаянной к оболочке стрелы и платформы. Такую же роль выполняют длинная гондола и несколько жестких распорок внутри дирижабля. Назначение этих креплений – обеспечить неизменность формы дирижабля, что достигается еще наличием внутри дирижабля баллонетов – особых мешков, помещаемых внутри оболочки и надуваемых воздухом.

Мягкие – мягкой оболочкой при полном отсутствии в ней жестких частей и с короткой, низко подвешенной к оболочке гондолой. Неизменность формы оболочки достигается наличием только одного или нескольких баллонетов.

Первая попытка построить дирижабль относится к 1812 г., когда в Российской империи инженером Лепехом для войны с Наполеоном был предложен оригинальный проект [1, 2], а первая удачная попытка датируется 1896 г., когда инженеру Шварцу удалось построить дирижабль, поднявшийся в воздух [3].

История развития отечественного и мирового дирижаблестроения подробно рассмотрена целым рядом авторов в работах [1–3].

Вследствие тех или иных проблем проектирования и применения от массового использования дирижаблей в середине двадцатого века отказались, и интерес к ним возродился лишь в восьмидесятых годах прошлого века.

Современные дирижабли лишены многих недостатков своих предшественников. Они заполняются пожаробезопасным гелием. Кроме этого, существенным усовершенствованием подвергся сам корпус, баллон, включая «обшивку» и несущую конструкцию. Для создания последней используют металлические фермы из авиационных сплавов. Саму оболочку делают из специальной ткани на основе лавсана. При покрытии применяется двуокись титана, в том числе делающая баллон почти абсолютно радиопрозрачным. Мотоблок дирижабля может состоять из одного или нескольких двигателей – как электрических, так и дизельных. За счет маршевых двигателей аэростат движется в заданном направлении, а для управления и маневрирования, в том числе и в режиме зависания, используются рулевые двигатели. Направление вектора тяги свободно изменяется на вертикальное. Бортовые системы позволяют успешно пилотировать корабль как днем, так и ночью.

Учитывая все возрастающий интерес к дирижаблям и активному успешному их применению в различных промышленных задачах целым рядом стран, авторы данной статьи предлагают рассмотреть потенциальную возможность применения дирижаблей в задачах пожаротушения.



Дирижабли могут быть использованы в автономном и/или оперативном режиме, в первом случае они осуществляют постоянный мониторинг, во втором – оперативно разворачиваются и снаряжаются для выполнения целевой задачи.

Основными преимуществами дирижаблей перед прочей авиационной техникой (вертолеты, самолеты) в задачах пожаротушения являются:

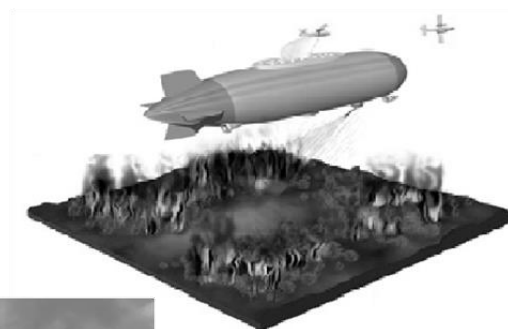
- безопасность, так как отсутствует турбореактивный двигатель, то есть исключены потенциальные аварии, связанные с помпажом;
- грузоподъемность;
- большая автономность;
- большая практическая дальность применения;
- инвариантность и вариативность операции забора воды.

Забор воды (рисунок) может быть осуществлен:

- с помощью насоса (см. рис. 1, а). За счет использования компактного (до 60 кг) насоса, к примеру, типа «Гном» (изолированный, защищенный и стойкий к агрессивным средам), осуществляется забор и подача воды на высоты порядка 20–30 м. Данная операция осуществляется за счет опускания насоса на штанге или тросе непосредственно в водоем (пруд, болото, озеро и т. д.) с помощью бухты отдачи с редуктором, позволяющим осуществлять разматывание троса на требуемую величину.

- загрузка воды с вертолета или любого другого летательного аппарата (см. рис. 1, б). Предполагается, что забор воды осуществляется вертолетом, с последующей процедурой переливания в цистерну дирижабля. При этом возникает ряд задач, связанных со стыковкой водоносного ковша вертолета и ответной части цистерны дирижабля. Потенциально можно решать данную задачу следующими способами: за счет проектирования системы направляющих конусов и соответствующих ответных частей, разработки направляющих конусов с системой электромагнитов, для точного позиционирования и стыковки ответных частей переливного механизма и проектирование системы автоматической перекачки. При данном способе необходимо учитывать температуру замерзания воды и, как следствие, ввести ограничение на допустимые высоты перелива.

- заправка с земли (см. рис. 1, в). За счет использования подручных средств (водоноски, гидранты и т. д.). Данный способ реализуется при базировании дирижабля на аэродроме, характеризуется простотой, отсутствием потенциальных ограничений, не требует разработки специализированных агрегатов заправки.



Для всех трех случаев также актуальна частная задача по расчету объема залитой воды, тем не менее в большинстве случаев она сводится к стандартной авиационной задаче по расче-

ту объема топлива (за счет поплавковых топливомеров получают значение уровня топлива, при известных характеристиках профиля бака нетрудно определить объем жидкости в нем). Учитывая нежесткие требования по точности определения запаса жидкости на борту, также потенциально возможно прямое непосредственное определение уровня жидкости в цистерне (за счет нанесения рисок на цистерну, использования контрольных щупов и т. д.).

После заполнения цистерн дирижабля водой возникает необходимость в решении следующих частных задач:

- реализация системы обеспечения теплового режима (использования нагревателей, для поддержания воды в жидком агрегатном состоянии);
- вопросы, связанные с центровкой (распределение жидкости в цистернах при непосредственном наполнении, совершении пространственных девиаций при полете к заданному месту и перераспределение жидкости в цистернах при непосредственной выборке в процессе пожаротушения). Данная задача также может быть сведена к стандартной авиационной, связанной с разработкой автомата центровки, обеспечивающего компенсацию опрокидывающего момента при неравномерной выработке топлива из разных баков;
- вопросы, связанные с распылением воды при непосредственном пожаротушении и осаждении вулканического пепла (выбор профиля струи, регулировка давления и пр.). Наиболее целесообразно, по мнению авторов доклада, использовать водяные пушки на турели (подвижное основание) с диафрагменным (сложного изменяемого профиля) выходом для оперативного формирования потребной струи (позволит решать одновременно задачи пожаротушения и осаждения пепла).

Характеристики прототипа дирижабля при пожаротушении следующие:

Название	Цеппелин LZ-127 «Граф Цеппелин»
Система/объем, м <sup>3</sup>	Жесткая/105000
Год постройки	1928
Максимальная/крейсерская скорость, км/ч	128/115
Наибольшая высота, м	7000
Радиус действия, км	10000 при 15 т нагрузке
Полезная нагрузка/средняя, т	25-55/ 30
Длина дирижабля	236,6
Диаметр (максимальный) дирижабля	30,5
Несущий газ	Водород и/или газ Блау
Число газовых отсеков	17
Силовая установка (марка и модель)/число двигателей и мощность, л. с.	Майбах VL II/ 5*530
Размеры гондолы (LxВxН), м	40x6x2,25
Наработка, ч	17200

Вопросы, связанные с дирижаблестроением и применением дирижаблей в России, на данном этапе рассматриваются недостаточно подробно. В статье были затронуты лишь общие вопросы дирижаблестроения и актуальные аспекты их непосредственного применения, тем не менее, обращаясь к разностороннему и богатому иностранному опыту, можно с уверенностью заявить о перспективности использования дирижаблей.

### Библиографический список

1. Ионов П. П. Дирижабли и их военное применение. М.: Государственное военное издательство, 1933. 120 с.
2. [Электронный ресурс], режим доступа: [http://samlib.ru/h/harlamow\\_i\\_b/dirigeable.shtml](http://samlib.ru/h/harlamow_i_b/dirigeable.shtml), свободный.
3. Кирилин А. Н. Дирижабли. МАИ: М., 2013. С. 405.

УДК 656.073.73

**М. С. Колокольцова**

студентка кафедры системного анализа и логистики

**Д. Л. Головцов**

зав. лаб. – научный руководитель

### БИКРИТЕРИАЛЬНАЯ НЕЛИНЕЙНАЯ ЗАДАЧА ПЕРЕВОЗКИ НЕОДНОРОДНЫХ ГРУЗОВ НА РАЗНОРОДНОМ ТРАНСПОРТЕ

К современному рынку грузоперевозок, помимо минимизации транспортных издержек, предъявляются многочисленные дополнительные требования, являющиеся зачастую более значимыми, чем фактор стоимости предоставляемой услуги. К данным требованиям можно отнести доставку продукции в нужное время (точно в срок), в нужное место, в необходимом количестве; оперативное и безошибочное оформление документов, сопровождающих доставку; сохранение определенных потребительских характеристик продукции и т.д.

Задача определения наиболее рационального способа транспортировки некоторого продукта довольно часто может быть описана с помощью транспортных моделей. Стандартная транспортная задача заключается в отыскании такого плана перевозок, при котором суммарная стоимость перевозок минимальна.

Однако во многих практических задачах минимизацию суммарной стоимости перевозок необходимо решать совместно с другими требованиями, которые тоже должны быть выполнены.

Очевидно, что эта проблема может быть решена с помощью любого из методов многокритериальной оптимизации [1]. Эти методы различны по своей природе и в общем случае дают решения, не совпадающие между собой, а вычислительная сложность такого класса задач является достаточно высокой [2].

Например, доставка скоропортящихся продуктов предполагает минимизацию максимального времени доставки, что может привести к увеличению стоимости перевозок, т.е. необходимо найти решение бикритериальной транспортной задачи:

$$\sum_{i=1}^n \sum_{j=1}^m f_{ij} x_{ij} \rightarrow \min; \quad (1)$$

$$\max_{(i,j)} \{t_{ij} : x_{ij} > 0\} \rightarrow \min. \quad (2)$$

В дополнение к этому, в реальной ситуации необходимо учесть и другие факторы и ресурсы.

Рассмотрим задачу доставки разнономенклатурных скоропортящихся продуктов разнородным транспортом с двумя целевыми функциями (1), (2).

Пусть для обеспечения перевозок может быть использовано  $s$  автохозяйств, в каждом из которых  $r$  типов автомашин. Машины разных типов, обладая различными эксплуатационными характеристиками и разной скоростью, могут доставлять любой из  $m$  грузов каждому из  $n$  потребителей.

Расстояние от места расположения  $g$ -го автохозяйства ( $g = 1, \dots, s$ ) до пункта производства  $i$ -го груза ( $i = 1, \dots, m$ ) известно. Известна скорость машин  $k$ -го типа ( $k = 1, \dots, r$ ) для всех маршрутов. Известно время погрузки и разгрузки машин каждого пункта назначения. С учетом этой информации можно определить  $t_{ijgk}$  – время занятости одной машины  $k$ -го типа  $g$ -го автохозяйства на работах по перевозке  $i$ -го груза  $j$ -му потребителю.

Математическую модель данной задачи можно представить в следующем виде:

$$\sum_{j=1}^m \sum_{i=1}^n x_{ijgk} \leq a_{gk} \quad \forall g, k; \quad (4)$$

$$\sum_{g=1}^s \sum_{k=1}^r d_{ij} x_{ijgk} = c_{ij} \quad \forall i, j; \quad (5)$$

$$x_{ijgk} \geq 0 \quad \forall i, j, g, k, \quad (6)$$

где  $a_{gk}$  – количество машин  $k$ -го типа в  $g$ -ом автохозяйстве,  $c_{ij}$  – число единиц  $i$ -го груза, подлежащего перевозке  $j$ -му потребителю,  $d_{ij}$  – число единиц  $i$ -го груза, которое перевозится в  $j$ -ый пункт назначения на одной машине (определяется по известной грузоподъемности машин),  $x_{ijgk}$  – количество машин  $k$ -го типа из  $g$ -го автохозяйства, предназначенных для перевозки  $i$ -го груза  $j$ -му потребителю.

Полученная четырехиндексная задача путем преобразований может быть сведена к классической двухиндексной транспортной задаче достаточно общим для многих задач способом [3].

В новых обозначениях задача сводится к вычислению переменных  $z_{\lambda\mu}$ , обращающих в минимум линейную форму:

$$\sum_{\lambda=1}^{mn} \sum_{\mu=1}^{sr} \tau_{\lambda\mu} z_{\lambda\mu} \rightarrow \min; \quad (7)$$

$$\sum_{\lambda=1}^{mn} z_{\lambda\mu} \leq b_{\mu}, \quad \mu = 1, 2, \dots, sr; \quad (8)$$

$$\sum_{\mu=1}^{sr} z_{\lambda\mu} = g_{\lambda}, \quad \lambda = 1, 2, \dots, mn; \quad (9)$$

$$z_{\lambda\mu} \geq 0 \quad \forall \lambda, \mu, \quad (10)$$

где  $\tau_{\lambda\mu} = t_{ijgk}, z_{\lambda\mu} = x_{ijgk}$

Однако, целевая функция (7) минимизирует суммарное время перевозок, в то время как требуется определить план перевозок, для которого время наиболее продолжительной перевозки достигает минимума, т.е.

$$\max_{(\lambda, \mu)} \{ \tau_{\lambda\mu} : z_{\lambda\mu} > 0 \} \rightarrow \min. \quad (11)$$

Решение данной задачи сводится к решению задачи о максимальном потоке [4].

Целевая функция (1) может быть представлена в следующем виде:

$$\sum_{\lambda=1}^{mn} \sum_{\mu=1}^{sr} f'_{\lambda\mu} z_{\lambda\mu} \rightarrow \min, \quad (12)$$

где  $f'_{\lambda\mu} = f_{ijgk}$  – стоимость перевозки  $i$ -го груза  $j$ -му потребителю одной машиной  $k$ -го типа  $g$ -го автохозяйства.

Решение задачи (12), (8) – (10) позволит определить план перевозок, минимизирующий суммарную стоимость доставки.

Очевидно, что с увеличением минимального времени  $\tau_{\lambda\mu}$  будет уменьшаться суммарная стоимость перевозки и наоборот.

Таким образом, для данной задачи не существует оптимального решения, обеспечивающего оптимум всем частным целевым функциям исходной задачи.

Одним из основных, фундаментальных понятий при исследовании таких задач является понятие оптимального по Парето, или эффективного решения. Оно представляет собой обобщение понятия точки максимума (минимума) числовой функции на случай нескольких функций: решение Паретооптимально, если значение любого из критериев можно улучшить лишь за счет ухудшения значений остальных критериев [5].

Одним из наиболее простых методов нахождения Паретооптимального решения с двумя критериями – является метод последовательных уступок [6], предполагающий, что лицо, принимающее решения, подводится к выбору решения путем постепенного ослабления первоначальных требований.

Таким образом, нахождение решения исходной четырехиндексной бикритериальной нелинейной задачи можно привести к нахождению решений нескольких достаточно несложных задач, что значительно упрощает процедуру управления транспортным предприятием в части оптимизации автотранспортного парка, снижения транспортных издержек и выполнения временных условий доставки груза.

### Библиографический список

1. Поляк Б. Т. Введение в оптимизацию. М.: Наука, 1983.
2. Diaz Joze A. Solving multiobjective transportation problems. Econ-math overview, 15, 1979, pp. 62–73.
3. Гольштейн Е.Г., Юдин Д.Б. Задачи линейного программирования транспортного типа. М.: Наука, 1969. 384 с.
4. Триус Е.Б. Задачи математического программирования транспортного типа. М.: Советское радио, 1967. 204 с.
5. Подиновский В.В., Ногин В.Д. Парето – оптимальные решения многокритериальных задач. М.: Наука, 1982.
6. Шикин Е. В., Шикина Г. Е. Исследование операций: учебник. М.: ТК Велби, «Проспект», 2006. 280 с.

УДК 004.021

**М. Красавин**

студент кафедры аэрокосмических компьютерных и программных систем

**Н.А. Шехунова**

проф., канд. техн. наук – научный руководитель

## СРАВНИТЕЛЬНЫЙ АНАЛИЗ ДЕТЕРМИНИСТИЧЕСКИХ АЛГОРИТМОВ ГЕНЕРИРОВАНИЯ БОЛЬШИХ ПРОСТЫХ ЧИСЕЛ ДЛЯ РОССИЙСКОГО СТАНДАРТА ЦИФРОВОЙ ПОДПИСИ

Введение

Мы знаем, что задача генерации простых чисел относится к классическим вычислительным проблемам, известным с древних времен. Простые числа являются ключом к разрешению многих математических проблем и используются в разных областях математики и ее приложениях. Криптография, имитационное моделирование, тестирование материнских плат персональных компьютеров – вот далеко не полный перечень практического применения простых чисел. Поэтому простые числа интересуют не только математиков, но и некоторые коммерческие организации, а также военных (разведку и контрразведку), ввиду их особого применения в области защиты информации. О важности вопроса изучения простых чисел говорят и следующие факты:

- 1) за поиск больших простых чисел назначены крупные денежные премии;
- 2) некоторые из простых чисел являются запрещенными (законы об авторском праве запрещают их хранить и распространять).

Все методы генерации простых чисел действуют по следующей схеме: строится случайное нечетное число  $n$  нужной длины и затем проверяется на простоту одним из известных тестов.

Существует два типа критериев простоты: детерминированные и вероятностные [1].

Вероятностные тесты можно эффективно использовать для тестирования отдельных чисел, однако их результаты с некоторой вероятностью могут быть неверными, тем не менее ценой ко-

личества повторений теста с модифицированными исходными данными вероятность ошибки можно сделать как угодно малой.

Детерминистические тесты позволяют с вероятностью 1 определить, является ли испытуемое число простым или нет. Они, как правило, более сложные в реализации, чем вероятностные. Однако, когда необходимо генерировать очень большие простые числа и иметь абсолютную гарантию того, что построенное число простое, предпочтительно использовать детерминистический тест, как это сделано в федеральном стандарте ЭЦП ГОСТ Р 34.10–94 и Р 34.10-2001[2].

В данной статье рассматриваются два алгоритма генерации простых чисел с использованием детерминированных тестов на простоту: алгоритм генерации простого числа, описанный в Российском стандарте электронной цифровой подписи ГОСТ Р 34.10–94 [3] и собственный алгоритм генерации простого числа основанный на той же теореме.

Описание алгоритмов

Для генерации больших простых чисел в ГОСТ Р 34.10-94 используется детерминистический тест, основанный на следующей теореме [4]:

Теорема. Пусть  $p = qN + 1$ , где  $q$  – нечетное простое число,  $N$  – четное, и  $p < (2q + 1)^2$ .

Число  $p$  является простым, если выполняются следующие два условия:

- 1)  $2^{qN} = 1 \pmod{p}$ ;
- 2)  $2^N \neq 1 \pmod{p}$ .

Схема построения алгоритма ГОСТ Р 34.10–94 описывается следующим образом. Пусть требуется сформировать простое число  $p$  длины  $t \geq 17$  бит. С этой целью строится убывающий набор натуральных чисел  $t_0, t_1, \dots, t_s$ , где  $t_0 = t$  и  $t_s < 17$ , для которых выполняется условие. Последовательно вырабатываются простые числа  $p_s, p_{s-1}, \dots, p_0$ , причем длина числа  $p_i$  равна значению  $t_i$  для всех  $i = 1, \dots, s$ . Исходное простое значение  $p_s$  формируется путем случайного выбора числа размером менее 17 бит и проверкой на простоту методом пробного деления.

Генерация простого числа  $p_{i-1}$  по простому числу  $p_i$  осуществляется с использованием формулы  $p_{i-1} = p_i N + 1$ ,

где  $N$  – случайное четное число, такое, что длина числа  $p_i N + 1$  равна значению  $t_i$ . Число  $p_{i-1}$  считается полученным, если одновременно выполнены следующие два условия:

- 1)  $2^{p_i N} = 1 \pmod{p_{i-1}}$ ;
- 2)  $2^N \neq 1 \pmod{p_{i-1}}$ .

Если хотя бы одно из условий не выполнено, то значение  $N$  увеличивается на два, вычисляется новое значение  $p_{i-1}$ , которое снова проверяется на простоту по указанным двум условиям. Такая процедура выполняется до тех пор, пока не будет получено простое число  $p_{i-1}$ .

Схема построения собственного алгоритма основанного на той же теореме описывается следующим образом. Пусть требуется сформировать простое число  $2^{255} \leq p \leq 2^{256}$ . Тогда:

- 1) исходные данные:  $q$  – случайное простое число,  $N$  – четное;
- 2)  $p = qN + 1$ .
- 3) Если
  - 3.1)  $p < (2q + 1)^2$  . ;
  - 3.2)  $2^{qN} = 1 \pmod{p}$ ;
  - 3.3)  $2^N \neq 1 \pmod{p}$  . ,

то  $p$  – простое.

Переходим к следующему пункту.

Иначе, к  $N = N + 2$ . и переходим к пункту 2.

4) Если  $p < 2^{255}$ , то  $q$  заменяем на . и переходим к пункту 2.

5) Если  $2^{255} \leq p \leq 2^{256}$ , то  $p$  найдено.

6) Если  $p > 2^{255}$ , то находим предыдущий множитель  $q$  и  $N = N + 2$  до тех пор, пока  $p = qN + 1$ . не окажется в промежутке  $2^{255} \leq p \leq 2^{256}$  и не будет удовлетворять условиям пункта 3.

Анализ алгоритмов

Отметим, что в стандарте отсутствуют ссылки на какую-нибудь научную литературу, что создает определенные трудности при анализе.

Всегда ли в результате алгоритмов получаем простые числа?

Поскольку оба алгоритма делают вывод о простоте сгенерированного числа на основании одной и той же теоремы, приведем ее доказательство.

Теорема. Пусть  $p = qN + 1$ , где  $q$  – нечетное простое число,  $N$  – четное, и  $p \ll (2q + 1)^2$ . Число  $p$  является простым, если выполняются следующие два условия:

$$1) 2^{qN} \equiv 1 \pmod{p};$$

$$2) 2^N \not\equiv 1 \pmod{p}.$$

Доказательство [4].

Пусть  $\gamma$  – есть порядок числа 2 по модулю  $p$  и  $p$  имеет следующее каноническое разложение:  $p = p_1^{\alpha_1} p_2^{\alpha_2} \dots p_h^{\alpha_h}$ . Ввиду условия 1)  $\gamma$  делит  $p - 1$ , т. е.  $\gamma \mid p - 1$ . В силу условия 2)  $\gamma$  не является делителем числа  $\frac{p-1}{q}$ . Отсюда следует, что  $q \mid \gamma$ .

Согласно теореме Эйлера  $2^{\varphi(p)} \equiv 1 \pmod{p}$ , следовательно,  $\gamma \mid \varphi(p) \Rightarrow q \mid \varphi(p)$ , где  $\varphi(p) = p_1^{\alpha_1-1} p_2^{\alpha_2-1} \dots p_h^{\alpha_h-1} (p_1 - 1)(p_2 - 1)(p_h - 1)$ . Пусть  $q$  совпадает с простым множителем  $p_i$ . Из такого допущения следует, что  $p = qn'$  для некоторого натурального числа  $n'$ . Однако по условию теоремы имеем  $p = qN + 1$ . Поскольку  $q > 1$  не может делить число 1, то приходим к противоречию, из которого следует, что  $q$  должно делить число  $p_{i-1}$ , по крайней мере, для некоторого одного из значений  $i \in \{1, 2, \dots, h\}$

Таким образом, существует некоторое натуральное  $n \geq 2$ , такое, что имеем  $p_i - 1 = qn$  и  $p_i = qn + 1$ . Следовательно, при некотором натуральном  $m$  получим:  $p = mp_i = m(qn + 1) = qN + 1 \Rightarrow m = q(N - mn) + 1$ .

При некотором натуральном  $s \geq 0$  имеем  $m = qs + 1$ . и  $p = (qn + 1)(qs + 1)$ .

Пусть  $p$  есть составное число, тогда  $s \geq 2$ . (поскольку  $N$  и  $n$  – четные числа а  $s = N - mn$ ), из чего следует  $p \geq (2q + 1)^2$ . Это противоречит условию теоремы, следовательно,  $s = 0$ . и число  $p$  является простым. Что и требовалось доказать.

Большое значение имеет распознают ли алгоритмы все простые числа, при оценке их эффективности, поскольку основные временные затраты алгоритмов связаны с перебором большого количества чисел, прежде чем будет построено число, простоту которого удастся дока-

зять. В данных алгоритмах простые числа могут быть ими не распознаны [5]. Ниже приведен ряд пых чисел, которые не будут приняты двумя алгоритмами как простые, и просто продолжают пр:

$$p = 131 \cdot 530 + 1 = 69431 \text{ ку собственный алгоритм генерации простого числа отклоняется от}$$

соотношения  $d \lfloor t/2 \rfloor$ , то существуют числа, которые не проходят из-за расширения диапазона.

Например:

$$p = 163 \cdot 82 + 1 = 13367, q = 163. \text{ простое.}$$

Заметим, что условия алгоритма стандарта более слабые, чем в собственном. Это означает, что алгоритм ГОСТ в среднем отвергает меньшее количество проверяемых чисел, но и нод ивно меньшем диапазоне.

Результаты эксперимента

С целью сравнительного анализа изучаемых алгоритмов был проведен следующий компьютерный эксперимент. Оба алгоритма были реализованы программно. Все числа, полученные в ходе работы алгоритмов были проверены методом пробного деления (перебором делителей от 2 до  $\sqrt{n/2}$ ). Каждым алгоритмом было сгенерировано большое количество простых чисел, размер которых выбирался случайно в диапазоне до 512 битов. При этом ни одно из построенных чисел не было определено, методом пробного деления, как составное.

В ходе работы программы, тестирующей алгоритм ГОСТ, было обнаружено 3 простых числа, не прошедших проверку условий теоремы. Это следующие числа:

- 1)  $p = 131 \cdot 530 + 1 = 69431, q = 131$ ;
- 2)  $p = 521 \cdot 2546 + 1 = 1326467, q = 521$ ;
- 3)  $p = 32771 \cdot 1504 + 1 = 8540515853, q = 32771$ .

Аналогичная программа, проверяющая собственный алгоритм, выявила 8 ошибочно отброшенных чисел. Отличающихся из них – 5, а именно:

- 1)  $p = 163 \cdot 82 + 1 = 13367, q = 163$ ;
- 2)  $p = 389 \cdot 312 + 1 = 121369, q = 389$ ;
- 3)  $p = 439 \cdot 280 + 1 = 122921, q = 439$ ;
- 4)  $p = 1289 \cdot 504 + 1 = 649657, q = 1289$ ;
- 5)  $p = 8060489 \cdot 397424 + 1 = 3203431780337, q = 8060489$ .

Что примечательно, все числа были найдены алгоритмом ГОСТ, найдены и собственным.

Заключение

Данное исследование выявило, что главным недостатком обоих алгоритмов является то, что в ходе генерации чисел, алгоритмы иногда ошибочно пропускают простые числа. Кроме того, ошибки возникали в основном в начале работы алгоритмов. Это говорит о том, что их влияние на эффективность алгоритма не критично.

Установлено, что собственный алгоритм находит больше чисел из-за расширения диапазона, что напрямую влияет на криптостойкость системы.

### Библиографический список

1. Крэндэлл Р., Померанс К. Простые числа. Криптографические и вычислительные аспекты.
2. Процессы формирования и проверки электронной цифровой подписи. М.: Госстандарт России, 2001. 12 с.
3. Процедуры выработки и проверки электронной цифровой подписи на базе асимметричного криптографического алгоритма. М.: Госстандарт России, 1994. 14 с.
4. Молдовян Н.А. Теоретический минимум и алгоритмы цифровой подписи. Спб.:БХВ-Петербург, 2010. 304 с.
5. Анохин М. И., Варновский Н. П., Сидельников В. М., Яценко В. В. Криптография в банковском деле. М.: МИФИ, 1997.



УДК 681.51

**М. А. Крохалева**

студент кафедры аэрокосмических приборов и ИВК

**Ю. П. Иванов**

доцент, канд. техн. наук. – научный руководитель

## ИССЛЕДОВАНИЕ ОПТИМАЛЬНОГО АДАПТИВНОГО МЕТОДА ФИЛЬТРАЦИИ СИГНАЛОВ

Цель работы заключается в проведении исследования и сравнительного анализа по точности оптимальной финитной обработки сигналов и фильтрации Калмана по результатам моделирования.

В качестве априорной информации используются следующие данные:

– модель измерения – линейная, описываемая временной последовательностью:

$$Y_i = X_i + H_i,$$

где  $Y_i$ ,  $X_i$ ,  $H_i$  – скалярные случайные гауссовские стационарные временные последовательности, определяющие соответственно измерение, являющееся случайным процессом, оцениваемое значение полезного сигнала и помеху измерения в виде дискретного белого шума на  $i$ -м шаге,  $i=1 \dots N$ ;  $M(X_i) = 0$ ;  $M(H_i) = 0$ ;

– законы распределения полезного сигнала и помехи – нормальные;

– погрешности полезного сигнала и помехи не коррелированы, т. е. взаимный корреляционный момент  $K(X_i, H_i) = 0$ ;

– корреляционные функции заданы в следующем виде:

– для полезного сигнала  $K_X(i-j) = \sigma_X^2 \exp(-\alpha|i-j|d)$ ,

где  $i, j = 1 \dots N$ ,  $\sigma_X$  – среднеквадратическое отклонение полезного сигнала;  $d$  – дискрет; модель помех  $H_i$  – дискретный гауссовский белый шум с корреляционной функцией

$$K_H(i-j) = \sigma_H^2 \delta_{i,j},$$

где  $\sigma_H$  – среднеквадратическое отклонение помехи  $\delta_{i,j}$  – дельта-функция,  $i, j = 1 \dots N$ ;

В процессе оценки из измерений  $Y_i$  и значений полезного сигнала  $X_i$  составляются векторы  $Y_1$  и  $X_1$  размерностью  $r \times 1$ , элементы которых, от  $i$  до  $i-r+1$ , являются измерениями на соответствующих шагах.

Для получения оптимальной оценки при использовании финитной фильтрации по критерию минимума среднего квадрата ошибки оценки используется следствие теоремы ортогонального проецирования [1], на основании которого оптимальная оценка фильтрации определяется по формуле

$$\hat{X}_1^* = A^* Y_1,$$

где  $i = 1 \dots N$ ,  $A$  – матричный оператор оптимальной финитной обработки размерностью  $r \times r$ , определяемые формулами

$$A^* = K_{X_1, Y_1} K_{Y_1}^{-1},$$

$$K_{X_1, Y_1} = K_{X_1},$$

где  $K_{X_1}$  и  $K_{Y_1}$  – матрицы корреляционных моментов векторов  $X_1$  и  $Y_1$  размерностью  $r \times r$ .

В случае отсутствия информации о полезном сигнале  $K_{X_1}$  и известной только матрице помех  $K_{H_1}$  исследуется адаптивная фильтрация, в которой матрица корреляционных моментов определяется в виде следующего соотношения:

$$KA_{Y1i} = KA_{Y1i-1} + \frac{1}{i-1} \left[ \left[ (Y1_i - M(Y1_i))(Y1_i - M(Y1_i))^T \right] - KA_{Y1i-1} \right],$$

где  $KA_{Y1}$  определяется рекуррентно при известном начальном значении  $KA_{Y1}$ ,

а в силу некоррелированности погрешности и полезного сигнала матрица  $KA_{X1}$  определяется формулой

$$KA_{X1} = KA_{Y1} - K_{H1},$$

где  $KA_X$  и  $K_{H1}$  – матрицы размерностью  $r \times r$ .

При адаптивном методе обработки оптимальная оценка фильтрации определяется по формуле

$$\hat{X}1_i^* = Aa^* Y1_i,$$

где матричный оператор адаптации  $Aa^* = KA_{X1} KA_{Y1}^{-1}$ .

Данный метод рассматривается на примере вычисления координат в навигационной системе, которая применяется на летательных аппаратах. В ее состав входит, в частности, блок МПВ-5, функцией которого является прием информации со спутников. В сигнале содержится флюктуационная ошибка, которая и устраняется с помощью исследуемого фильтра.

При моделировании алгоритмов были выбраны следующие параметры:

- объем выборки  $N = 10000$ ;
- диапазоны изменения входных данных:  $\sigma_H = 0,1 \dots 10 \text{ м/с}^2$ ;  $\alpha = 0,01 \dots 0,11/\text{с}$ ;
- $r = 2 \dots 8$ ;  $d = 2 \dots 4$ ;
- дискрет  $d$  определяется в соответствии с теоремой Котельникова [1], согласно которой дискрет не должен превышать

$$d < \frac{\pi}{\alpha \text{tg} \left( \frac{\pi}{2} - 0,05 \frac{\pi}{2} \right)},$$

т.е.  $d < 24$  при указанных выше параметрах и  $\alpha = 0,01$ .

В результате моделирования при использовании оптимальных финитной (Dwef), адаптивной (DwefA2) обработки сигналов и при фильтрации Калмана (Dwefk2) были получены зависимости статистических дисперсий ошибок оценок от времени соответственно (рис. 1).

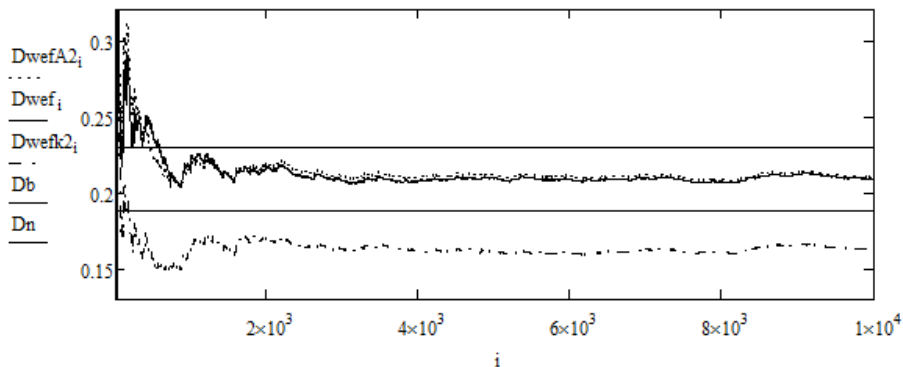


Рис.1. Зависимость значений дисперсий от дискретного времени  $i$ , где  $Db$  и  $Dn$  – верхняя и нижняя границы при допуске в 10%

Как видно из рис.1, финитная обработка незначительно уступает по точности обработке фильтром Калмана. Также можно заключить, что адаптивная обработка после окончания периода адаптации, близка к финитной. Время переходного процесса, при допуске в 10% от стационарного

финитного установившегося значения, занимает 450 секунд. Обработка фильтром Калмана превосходит финитную обработку в 1,27 раз (на конец выборки при  $\alpha = 0,01$ ;  $\sigma_H = 1$ ;  $d = 2$ ;  $r = 5$ ).

Также проводился анализ зависимости дисперсий ошибок оценок при финитной обработке (DE2) и фильтрации Калмана (DEk2) от изменения исходных данных в указанных выше диапазонах (рис. 2, 3).

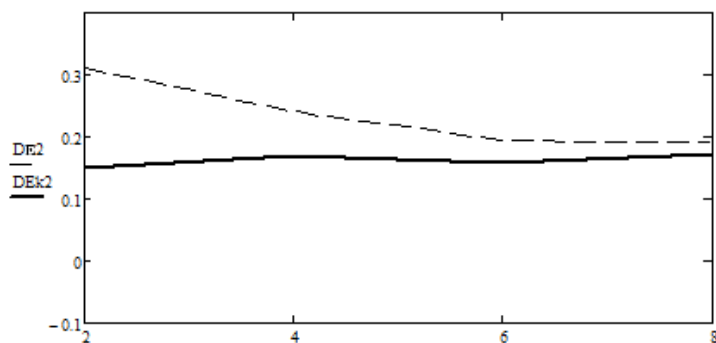


Рис.2. Зависимость дисперсий ошибок оценок DE2 и DEk2 от памяти  $r$

Из графиков на рис.2 видно, что при увеличении памяти  $r$  финитная оптимальная обработка приближается по точности к фильтрации Калмана, а при  $r = 8$  эти виды обработок практически совпадают, поэтому при использовании алгоритма финитной обработки необходимо повышать это значение учитываемых элементов.

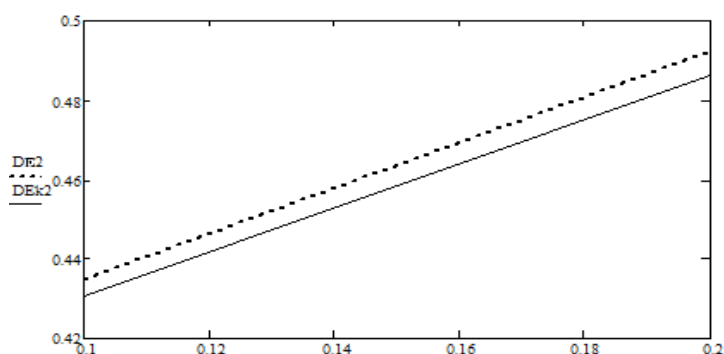


Рис.3. Зависимость дисперсий ошибок оценок DE2 и DEk2 от  $\alpha$

При анализе графиков на рис.3 можно заключить, что при увеличении  $\alpha$  дисперсии ошибок оценок всех видов обработки увеличиваются, при этом наблюдается незначительное отклонение дисперсий при финитной фильтрации от дисперсий при обработке фильтром Калмана, что указывает на схожую реакцию алгоритма финитной обработки на приближение к безынерционной обработке.

По результатам сравнительного анализа выявлены следующие достоинства финитной обработки по отношению к фильтрации Калмана:

- более универсальная, поскольку ее вид не зависит от характера КФ помехи, применяется для стационарных и нестационарных процессов;
- более устойчивая к изменению параметров вследствие отсутствия обратной связи;
- является более простая за счет другого математического аппарата;
- не требует свойства марковости, т. е. представления сигналов в пространстве состояний, в связи с чем данную обработку можно применять для любого интервала времени.

В качестве недостатка финитного метода обработки следует отметить, что требуется увеличение объема памяти для приближения по точности этого вида обработки к фильтрации Калма-

на, что вполне разрешимо посредством высокой производительности современных вычислительных машин, а также при помощи использования спектрально-финитной обработки сигналов.

### Библиографический список

1. Иванов Ю. П., Никитин В.Г. Информационно-статистическая теория измерений: учеб. пособие. СПб.: ГУАП, 2011. 104 с.
2. Иванов Ю. П., Бирюков Б. Л. Информационно-статистическая теория измерений. Модели сигналов и анализ точности систем: учеб. пособие / Ю. П. Иванов, Б. Л. Бирюков. СПб.: ГУАП, 2008. 160 с.
3. Тихонов В. И. Статистический анализ и синтез радиотехнических устройств и систем: учеб. пособие для вузов. М.: радио и связь, 1991. 608 с.

УДК 004.94

**Л. И. Курбанов**

студент кафедры аэрокосмических компьютерных и программных систем.

**Н. А. Матвеева**

научный руководитель, аспирант кафедры аэрокосмических компьютерных и программных систем.

## ПРИМЕНЕНИЕ ПОДХОДА ПОИСКА Р-МЕДИАНЫ ПРИ РАЗМЕЩЕНИИ СПЕЦИАЛИЗИРОВАННЫХ УЗЛОВ В СЕТИ-НА-КРИСТАЛЛЕ

Для увеличения производительности и скорости передачи данных в проектируемой сети-на-кристалле (СенК) необходимо максимизировать пропускную способность каналов передачи данных и минимизировать энергопотребление и задержки. Для решения данной задачи существует подход (рис.1), основанный на оптимальном размещении специализированных сложно-функциональных (СФ) блоков в СенК. В качестве СФ-блока понимается узел, который выполняет определенную задачу в СенК. СФ-блоком может быть процессор, контроллер памяти, устройство ввода/вывода или любое другое функциональное устройство. СенК состоит из множества узлов и множества линий связи, соединяющих эти узлы. Узлы в СенК бывают двух типов: коммутаторы и терминальные узлы. <sup>1</sup>

Для решения задачи размещения СФ-блоков достаточно иметь структурное представление СенК, которое, в свою очередь, можно описать с помощью графа. Определим сеть в виде графа  $G(V, E)$ , где  $V$  – множество узлов в сети,  $E$  – множество линий связи. В терминологии теории графов данная задача трактуется как поиск  $p$ -медианы. Задачу о нахождении  $p$ -медианы данного графа  $G$  будем трактовать как нахождение местоположения в СенК заданного числа СФ-блоков ( $p$ ), к которым прикрепляются ближайшие узлы СенК. При этом объединение множества медианных вершин  $V_p$  и множества прикрепляемых вершин  $V_A$  должно полностью покрывать множество всех вершин  $V$  графа  $G$ , т.е. должно выполняться логическое равенство:  $V_p \cup V_A = V$ .<sup>2</sup>

При размещении СФ-блоков в СенК используются следующие ограничения:

Расстояние  $[D, hops]$ . Определяет максимально возможное расстояние от медианы до прикрепляемой к ней вершины.

<sup>1</sup> Nadezhda Matveeva, Elena Suvorova and Lev Kurbanov Applying the  $p$ -medians in the design of modern systems-on-chip / Nadezhda Matveeva, // Proceedings of Artificial Intelligence and Natural Language & Information Extraction, Social Media and Web Search FRUCT Conference, Saint-Petersburg, Russia, 9-14 November 2015. 102-109 pp.

<sup>2</sup> Кристофидес Н. Теория графов. Алгоритмический подход. М.: Мир, 1978.

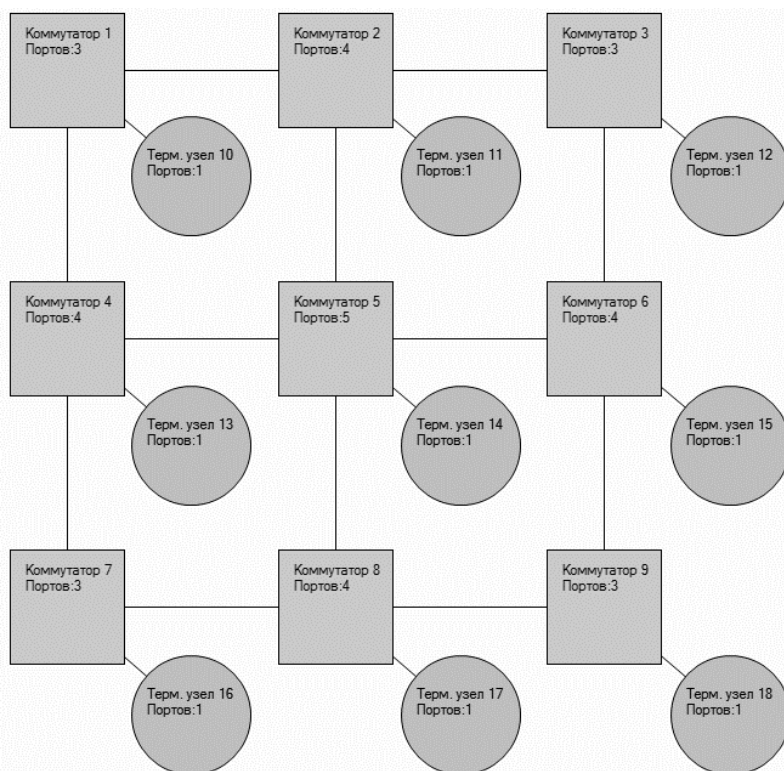


Рис. 1. Пример сети-на-кристалле топологии типа «решетка 3x3»

Загрузка  $[\Delta, \%]$ . Определяет допустимое отклонение от равномерного прикрепления вершин к медианам. Количество прикрепляемых вершин  $[N_a]$  к медианам определяется неравенством:  $\frac{|V/P|}{|P|} \left(1 - \frac{\Delta}{100}\right) \leq N_a \leq \frac{|V/P|}{|P|} \left(1 + \frac{\Delta}{100}\right)$ , где  $V$  – множество всех вершин,  $P$  – множество медианных вершин.

Определим критерии оптимальности решения для данных ограничений:

- минимальное расстояние от СФ-блока до остальных узлов в СенК;
- максимально равномерное прикрепление узлов к СФ-блокам.

Эти критерии можно использовать в следующих режимах:

1. Без ограничений. Ищутся самые оптимальные решения по расстоянию, и по возможности осуществляется равномерное прикрепление. Могут быть проигрыши в равномерности прикрепления.

2. Расстояние. Ищутся самые оптимальные решения по расстоянию, и по возможности осуществляется равномерное прикрепление. Если пользователь указал ограничение по расстоянию меньше минимально возможного, решения не найдутся. Если пользователь указал ограничение по расстоянию больше минимально возможного, найдутся только оптимальные решения. Могут быть проигрыши в равномерности прикрепления.

3. Расстояние и Загрузка. Ищутся самые оптимальные решения по расстоянию, но которые будут удовлетворять ограничению по загрузке. Если ни одно решение с оптимальным расстоянием не удовлетворяет ограничению по загрузке, то расстояние до вершин увеличивается в пределах, указанных в ограничении по расстоянию. Поиск производится до тех пор, пока не будет удовлетворено ограничение по загрузке. Могут быть проигрыши по расстоянию.

Рассмотрим пример. Необходимо разместить 2 процессора ( $p=2$ ) в сети-на-кристалле размером 3x3 с топологией типа «решетка» (рис.2) при этом достичь минимального расстояния от процессоров до других узлов и добиться максимально равномерного прикрепления вершин. В данном примере используется режим «Без ограничений».

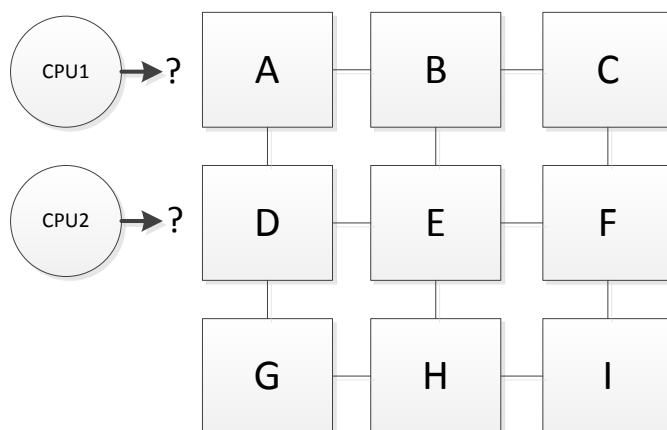


Рис.2. Пример сети 3x3, в которой нужно разместить 2 процессора

Алгоритм:

1. Построим матрицу кратчайших расстояний. Эта матрица строится алгоритмом Форда-Фолкерсона или волновым алгоритмом. Волновой алгоритм может также запомнить кратчайший путь.

	A	B	C	D	E	F	G	H	I
A	0	1	2	1	2	3	2	3	4
B	1	0	1	2	1	2	3	2	3
C	2	1	0	3	2	1	4	3	2
D	1	2	3	0	1	2	1	2	3
E	2	1	2	1	0	1	2	1	2
F	3	2	1	2	1	0	3	2	1
G	2	3	4	1	2	3	0	1	2
H	3	2	3	2	1	2	1	0	1
I	4	3	2	3	2	1	2	1	0

2. Отсортируем матрицу кратчайших расстояний по возрастанию расстояния. Индекс у элементов показывает расстояние от данной вершины до вершины в левом столбце.

A	A <sub>0</sub>	B <sub>1</sub>	D <sub>1</sub>	C <sub>2</sub>	E <sub>2</sub>	G <sub>2</sub>	F <sub>3</sub>	H <sub>3</sub>	I <sub>4</sub>
B	B <sub>0</sub>	A <sub>1</sub>	C <sub>1</sub>	E <sub>1</sub>	D <sub>2</sub>	F <sub>2</sub>	H <sub>2</sub>	G <sub>3</sub>	I <sub>3</sub>
C	C <sub>0</sub>	B <sub>1</sub>	F <sub>1</sub>	A <sub>2</sub>	E <sub>2</sub>	I <sub>2</sub>	D <sub>3</sub>	H <sub>3</sub>	G <sub>4</sub>
D	D <sub>0</sub>	A <sub>1</sub>	E <sub>1</sub>	G <sub>1</sub>	B <sub>2</sub>	F <sub>2</sub>	H <sub>2</sub>	C <sub>3</sub>	I <sub>3</sub>
E	E <sub>0</sub>	B <sub>1</sub>	D <sub>1</sub>	F <sub>1</sub>	H <sub>1</sub>	A <sub>2</sub>	C <sub>2</sub>	G <sub>2</sub>	I <sub>2</sub>
F	F <sub>0</sub>	C <sub>1</sub>	E <sub>1</sub>	I <sub>1</sub>	B <sub>2</sub>	D <sub>2</sub>	H <sub>2</sub>	A <sub>3</sub>	G <sub>3</sub>
G	G <sub>0</sub>	D <sub>1</sub>	H <sub>1</sub>	A <sub>2</sub>	E <sub>2</sub>	I <sub>2</sub>	B <sub>3</sub>	F <sub>3</sub>	C <sub>4</sub>
H	H <sub>0</sub>	E <sub>1</sub>	G <sub>1</sub>	I <sub>1</sub>	B <sub>2</sub>	D <sub>2</sub>	F <sub>2</sub>	A <sub>3</sub>	C <sub>3</sub>
I	I <sub>0</sub>	F <sub>1</sub>	H <sub>1</sub>	C <sub>2</sub>	E <sub>2</sub>	G <sub>2</sub>	B <sub>3</sub>	D <sub>3</sub>	A <sub>4</sub>

3. Выберем две строки (так как  $p=2$ ), исключая вершины-медианы в строке.

<b>A</b>	A <sub>0</sub>	B <sub>1</sub>	D <sub>1</sub>	C <sub>2</sub>	E <sub>2</sub>	G <sub>2</sub>	F <sub>3</sub>	H <sub>3</sub>	I <sub>4</sub>
<b>B</b>	B <sub>0</sub>	A <sub>1</sub>	C <sub>1</sub>	E <sub>1</sub>	D <sub>2</sub>	F <sub>2</sub>	H <sub>2</sub>	G <sub>3</sub>	I <sub>3</sub>
<b>C</b>	C <sub>0</sub>	B <sub>1</sub>	F <sub>1</sub>	A <sub>2</sub>	E <sub>2</sub>	I <sub>2</sub>	D <sub>3</sub>	H <sub>3</sub>	G <sub>4</sub>
<b>D</b>	D <sub>0</sub>	A <sub>1</sub>	E <sub>1</sub>	G <sub>1</sub>	B <sub>2</sub>	F <sub>2</sub>	H <sub>2</sub>	C <sub>3</sub>	I <sub>3</sub>
<b>E</b>	E <sub>0</sub>	B <sub>1</sub>	D <sub>1</sub>	F <sub>1</sub>	H <sub>1</sub>	A <sub>2</sub>	C <sub>2</sub>	G <sub>2</sub>	I <sub>2</sub>
<b>F</b>	F <sub>0</sub>	C <sub>1</sub>	E <sub>1</sub>	I <sub>1</sub>	B <sub>2</sub>	D <sub>2</sub>	H <sub>2</sub>	A <sub>3</sub>	G <sub>3</sub>
<b>G</b>	G <sub>0</sub>	D <sub>1</sub>	H <sub>1</sub>	A <sub>2</sub>	E <sub>2</sub>	I <sub>2</sub>	B <sub>3</sub>	F <sub>3</sub>	C <sub>4</sub>
<b>H</b>	H <sub>0</sub>	E <sub>1</sub>	G <sub>1</sub>	I <sub>1</sub>	B <sub>2</sub>	D <sub>2</sub>	F <sub>2</sub>	A <sub>3</sub>	C <sub>3</sub>
<b>I</b>	I <sub>0</sub>	F <sub>1</sub>	H <sub>1</sub>	C <sub>2</sub>	E <sub>2</sub>	G <sub>2</sub>	B <sub>3</sub>	D <sub>3</sub>	A <sub>4</sub>

4. Выведем в новую таблицу прикрепляемые вершины, медианы, к которым они прикрепляются и расстояние до них в порядке возрастания.

<b>Прикрепляемые вершины</b>	A	C	E	G	I	D	F	A	C	G	I
<b>Медианы</b>	B	B	B,H	H	H	B,H	B,H	H	H	B	B
<b>Расстояние</b>	1	1	1	1	1	2	2	3	3	3	3

5. Удалим прикрепляемые вершины, до которых расстояние не минимальное.

<b>Прикрепляемые вершины</b>	A	C	E	G	I	D	F	A	C	G	I
<b>Медианы</b>	B	B	B,H	H	H	B,H	B,H	H	H	B	B
<b>Расстояние</b>	1	1	1	1	1	2	2	3	3	3	3

6. Из полученной таблицы можно однозначно определить какие вершины прикрепятся к медианам, т.е вершины, которые могут прикрепиться только к одной медиане.

<b>Прикрепляемые вершины</b>	A	C	E	G	I	D	F
<b>Медианы</b>	B	B	B,H	H	H	B,H	B,H
<b>Расстояние</b>	1	1	1	1	1	2	2

7. Распределяем оставшиеся вершины между медианами так, чтобы достичь максимальной равномерности нагрузки.

<b>Прикрепляемые вершины</b>	E	D	F
<b>Медианы</b>	B	B	H
<b>Расстояние</b>	1	2	2

На рис. 3 показано данное решение. Медианами являются узлы В и Н. К медиане В прикрепляются узлы А,С,Д,Е. К медиане Н прикрепляются узлы F, G, I.

Следует отметить, что при увеличении количества узлов в сети и количество искомым медиан время работы алгоритма возрастает экспоненциально. Экспериментально были получены зависимости времени работы алгоритма от количества узлов в сети и значений  $p$ .

Тестирование проводилось на компьютере со следующими характеристиками:

- ОС: Windows 8.1;
- процессор: Intel Core i7;
- частота: 2.93 GHz;
- ОЗУ: 4Гб;
- разрядность системы: x 64

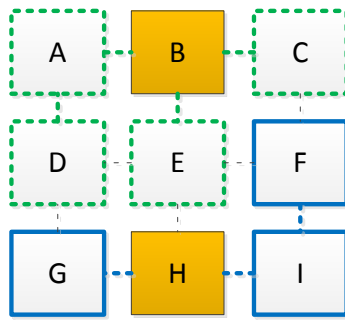


Рис.3. Пример решения сети 3x3

Во время тестирования были отключены все сторонние службы и приложения. Перед началом тестирования загрузка ЦП составляла 0–1%. Во время тестирования загрузка ЦП достигала 14–15%.

Для тестирования использовались топологии типа “решетка” следующих размеров  $n \times n$ , где  $n=3, 10$ . Для топологий 3x3, 9x9 и 10x10 производился поиск 2-х, 3-х и 4-х медиан. Для топологий 4x4, 5x5, 6x6, 7x7 и 8x8 производился поиск 2, 3, 4 и 5 медиан. По результатам тестирования были получены следующие графики зависимостей (рис. 4).

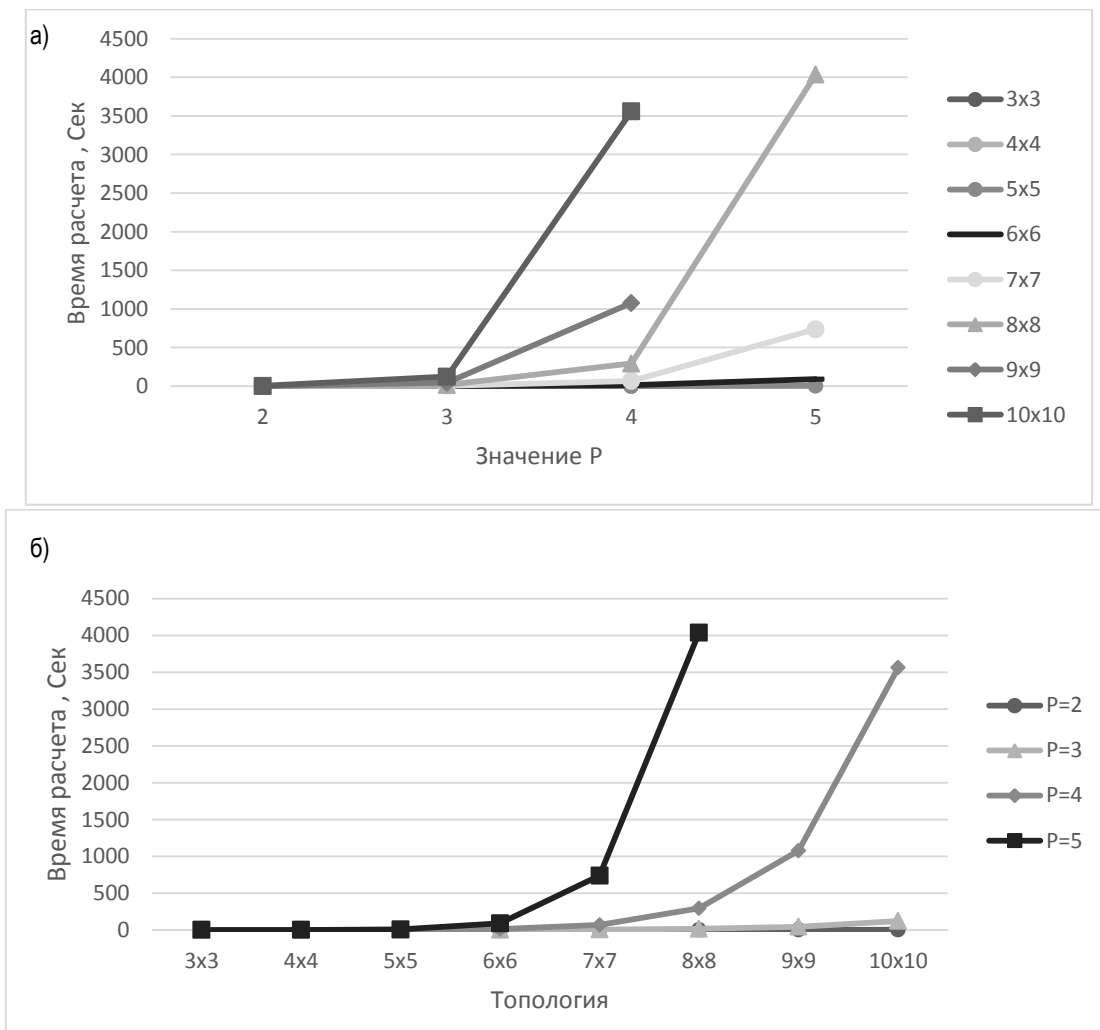


Рис.4. График зависимости времени работы программно реализованного алгоритма от количества узлов в сети для различных значений P



Из графиков виден экспоненциальный характер зависимости времени как от количества узлов в сети, так и от количества ищущихся медиан.

Количество возможных сочетаний из  $N$  вершин по  $p$  вершин можно представить в виде следующей формулы:  $C(p,N) = \frac{N!}{(N-p)!p!}$ , где  $N$  – количество вершин в сети;  $N = n \times n$ .

Максимальные временные значения в полученных данных тестирования. Для обработки сети размером  $8 \times 8$  для 5 медиан было потрачено примерно 66 мин. Для обработки сети размером  $10 \times 10$  для 4 медиан было потрачено примерно 58 мин.

Для сетей, содержащих более 100 узлов, следует применить эвристический алгоритм, что позволит значительно сократить временные затраты, но даст околооптимальное решение.

В дальнейшей работе будут исследоваться методы оптимизации алгоритма поиска  $p$ -медиан, например в сетях с регулярной топологией (например, «Решетка») можно применить стратегию «Разделяй и Властвуй», т. е. сеть разбивается на кластеры, количество которых определяется количеством ищущихся медиан, т.е. значением  $p$ . Кластеры должны представлять собой отдельные связанные компоненты графа, количество узлов в которых будет одинаковым. В каждом кластере будет производиться локально поиск одной медианы.

### Библиографический список

1. *Nadezhda Matveeva, Elena Suvorova and Lev Kurbanov* Applying the  $p$ -medians in the design of modern systems-on-chip / *Nadezhda Matveeva*, // Proceedings of Artificial Intelligence and Natural Language & Information Extraction, Social Media and Web Search FRUCT Conference, Saint-Petersburg, Russia, 9-14 November 2015. 102–109 pp.
2. *Кристофидес Н.* Теория графов. Алгоритмический подход. М.: Мир, 1978.

УДК 629.7.05

**И. В. Лебедев**

магистрант кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**Б. Л. Бирюков**

ст. преп. – научный руководитель

## СОВМЕСТНАЯ ОПТИМИЗАЦИЯ ПАРАМЕТРОВ ФИЛЬТРА НИЖНИХ ЧАСТОТ И ОПТИМАЛЬНОГО ФИЛЬТРА ПАРАМЕТРОВ ЛИНЕЙНОГО ДВИЖЕНИЯ КВАДРОКОПТЕРА

Массовый интерес к квадрокоптерам возник в последнее десятилетие, когда стали широко распространены миниатюрные микроэлектромеханические датчики, относительно емкие и мощные аккумуляторы и электродвигатели с улучшенными характеристиками. Из-за большого числа быстро движущихся элементов чем в беспилотных ЛА, выполненных по стандартной самолетной схеме на борту квадрокоптера появляются более интенсивные вибрации, поэтому требуются меры по снижению их влияния на точность измерений. В то же время присутствуют и собственные погрешности датчиков с флуктуационной составляющей погрешности. Для ослабления влияния погрешностей используются алгоритмы фильтрации. Вследствие того, что погрешности имеют разную природу, для их ослабления могут быть применены отдельные фильтры с целью упрощения алгоритма фильтрации. Их оптимизация также может быть выполнена отдельно. Вместе с тем, принимая во внимание то, что показателем качества системы является конечная точность определения рассматриваемых параметров движения, оптимизацию фильтров следует осуществлять согласованно.

Учитывая различные виды погрешностей, допустимо с целью упрощения синтеза рассматривать отдельное построение соответствующих фильтров. Так как спектр вибраций на бор-

ту квадрокоптера является дискретным и расположен преимущественно выше по частоте, чем другие составляющие погрешности, для снижения влияния вибрационной погрешности используется фильтр нижних частот (ФНЧ). При синтезе фильтра предполагается взаимная некоррелированность вибрационной составляющей с остальными составляющими погрешности.

Построение ФНЧ основывается на результатах исследования фильтров Чебышева и Бесселя и выборе наиболее подходящего из них. Для разрабатываемой системы наиболее важна линейность амплитудно-частотной характеристики фильтра и меньший выброс при переходных процессах, поэтому был выбран фильтр Бесселя и получена его передаточная функция для определенных параметров полета квадрокоптера.

Фильтр Калмана используется для остальных составляющих погрешности. Синтез фильтра Калмана осуществляется в варианте построения непрерывной стохастической системы и последующим определением параметров фильтра. В качестве исходного варианта был использован алгоритм фильтра Калмана в нестационарном режиме. Оптимальная мерная векторная оценка  $\hat{z}(t)$  сигнала  $z(t)$  описывается уравнением [1]:

$$\dot{\hat{z}}(t) = A(t)\hat{z}(t) + B(t)u(t) + K(t)[y(t) - H(t)\hat{z}(t)]$$

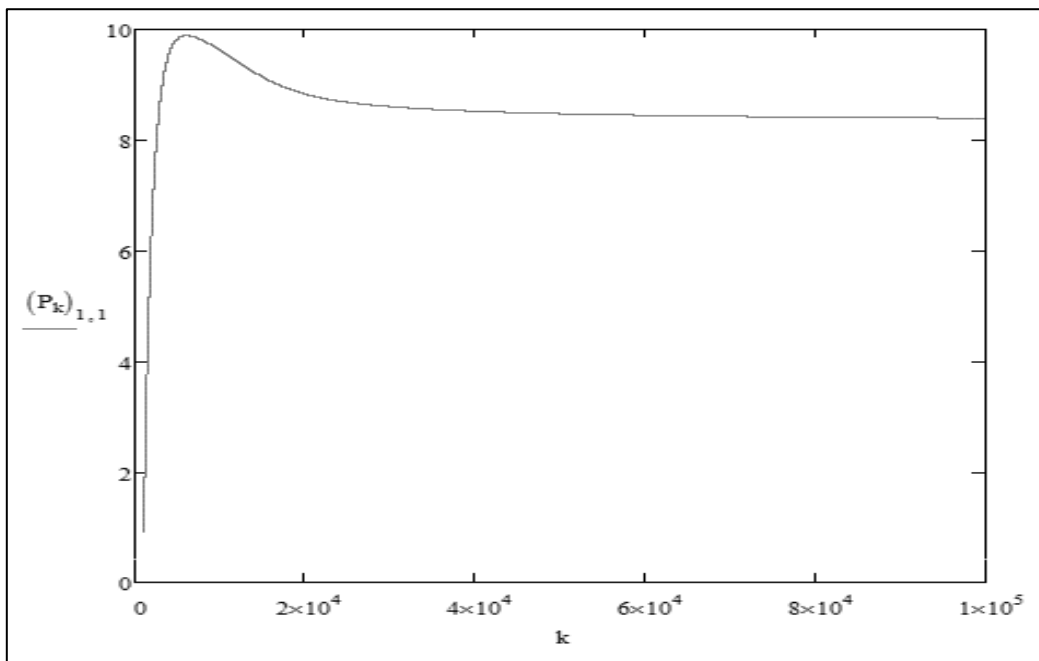
где  $K(t)$  – матричный коэффициент усиления (матрица передачи Калмана), определяемы следующим выражением:

$$K(t) = [P(t)H^T(t) + V(t)]R^{-1},$$

где  $P(t)$  – ковариационная матрица ошибок фильтрации, для нахождения которой необходимо решить уравнение Риккати:

$$\dot{P}(t) = A(t)P(t) + P(t)A^T(t) + Q(t) - P(t)H^TR^{-1}(t)H(t)P(t).$$

Для выполнения оптимизации системы, состоящей из двух фильтров, в стационарном режиме необходимо выражения для фильтра Калмана перевести в операторный вид. Это представляется возможным реализовать, используя установившиеся значения матрицы  $P(t)$  фильтра Калмана. В частности, процесс установления элемента  $P(t)_{1,1}$  матрицы  $P(t)$  показан на рисунке.



Зависимость  $P(t)_{1,1}$  от времени

Для оценивания качества полученной системы необходимо получить дисперсию ошибки оценки полезного сигнала. В данном случае рассматривалось оценивание высоты полета квадрокоптера  $H(t)$ .

Полная ошибка, проходящая через систему, складывается из двух частей:

– динамическая ошибка, представляющая собой отклонение оценки сигнала от истинного значения, возникающее при обработке измеренного сигнала. Динамическая ошибка в операторном представлении вычисляется по формуле

$$\varepsilon_d(p) = \hat{H}(p) - H(p),$$

где  $\hat{H}(p)$  – результат преобразования  $H(p)$  в системе,  $H(p)$  – полезный сигнал;

– флуктуационная ошибка, представляющая собой отклонение оценки сигнала от истинного значения, возникающее из-за наличия в выходном сигнале датчиков погрешностей, вызываемых вибрациями на борту ЛА, ветром, электронным шумом компонентов и т.д.

Флуктуационная ошибка вычисляется по формуле

$$\varepsilon_f(p) = \varepsilon_{f1}(p) + \varepsilon_{f2}(p),$$

где  $\varepsilon_{f1}(p)$  и  $\varepsilon_{f2}(p)$  – флуктуационные составляющие ошибки оценки высоты по измерительному каналу акселерометра и баровысотомера соответственно.

После получения выражения для динамической и флуктуационной составляющей погрешности рассчитывается дисперсия этой погрешности. В общем виде дисперсия отдельной составляющей ошибки оценки рассчитывается по следующей формуле [2]:

$$D_\varepsilon = \frac{1}{2\pi} \int_{-\infty}^{\infty} |\Phi(j\omega)|^2 S_\varepsilon(\omega) d\omega,$$

где  $\Phi(j\omega)$  – передаточная функция, включающая в себя все преобразования процесса (полезного сигнала или погрешности) при прохождении через систему;  $S_\varepsilon(\omega)$  – спектральная плотность процесса. Так как подынтегральное выражение представляют собой дробно-рациональную функцию заданного вида от  $j\omega$ , то такой интеграл является стандартным следующего вида:

$$D_\varepsilon = I_n = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{B_n(j\omega)}{A_n(j\omega)A_n(-j\omega)} d\omega,$$

где  $A_n$  и  $B_n$  – многочлены соответствующего вида.

Приведенное выше выражение для вычисления интеграла стандартного вида используется для вычисления дисперсии каждой составляющей погрешности, которые после суммируются.

Проведенное моделирование показало, что с увеличением частоты среза ФНЧ уменьшается динамическая составляющая дисперсии ошибки оценки, а усредненное значение дисперсии флуктуационной составляющей – увеличивается. Это позволило определить оптимальную величину частоты среза, при которой достигается минимальное значение дисперсии ошибки оценки. Использование дополнительного фильтра является целесообразным, так как приводит к существенному снижению составляющей дисперсии ошибки оценки, вызванной погрешностью акселерометров из-за вибрации на борту квадрокоптера. При этом размерность фильтра Калмана не повышается, а реализация ФНЧ не требует значительных вычислительных затрат.

### Библиографический список

1. Тихонов В. И., Харисов В. Н. Статистический анализ и синтез радиотехнических устройств и систем: учеб. пособ. для вузов. М.: Радио и связь, 2004. 608 с.
2. Бесекерский В. А., Попон Е. П. Теория систем автоматического управления. СПб.: Профессия, 2003. 752 с.

УДК 629.735.33

**Д. Ю. Лукашов**

студент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**В.Г. Никитин**

доц., канд. техн. наук – научный руководитель

### **РАЗРАБОТКА МЕТОДИКИ ПОДГОТОВКИ ТЕСТОВ ДЛЯ КОНТРОЛЯ И ДИАГНОСТИКИ АНАЛОГО-ЦИФРОВЫХ ПАРАМЕТРОВ БОРТОВОГО ОПТИЧЕСКОГО ИЗМЕРИТЕЛЬНОГО КОМПЛЕКСА**

В авиации часто возникает необходимость решения задачи по определению пространственных координат воздушных и наземных объектов по их тепловому излучению (ИК-поток). Для этого на многих летательных аппаратах (ЛА) используются различные варианты бортового оптического измерительного комплекса (БОИК), который определяет наличие объекта и угловые координаты (азимут и угол места) по инфракрасному излучению [1]. Эффективность использования БОИК определяется точностью выдаваемых координат и полнотой выполняемых задач, которые в свою очередь определяются трудозатратами и стоимостью проведения регламентных проверок, величиной достоверности и глубины контроля. Для этих целей используется наземная система контроля и диагностики (НСКД), которая позволяет проводить все виды испытаний, пердические и предполетные проверки, отладку и настройку БОИК после модернизации или ремонта. В связи с окончанием назначенного срока службы и с прекращением в предыдущие годы выпуска промышленностью фоточувствительного элемента, используемого в БОИК, а также большинства элетрорадиоэлементов было принято решение модернизировать измерительный комплекс. Также невозможно произвести заново предыдущий вариант контрольно-проверочной аппаратуры и было принято решение о разработке новой системы контроля БОИК с увеличением контролируемых параметров, точностных характеристик и глубины диагностики. Новая НСКД должна сократить влияние человеческого фактора на объективность контроля и упростить процедуру контроля для оператора.

В начале работы стояли следующие задачи:

- алгоритм контроля и диагностики должен обеспечивать проверку с помощью НСКД как старых вариантов БОИК, так и модернизированных;
- тестовые сигналы должны имитировать реальные задачи БОИК;
- создать возможность формирования нескольких режимов проверок; (при проведении ПСИ, периодических и типовых испытаниях, предполетной и периодической проверки);
- упростить алгоритм работы оператора и интерфейса программы по сравнению с контрольно-проверочной аппаратурой старого образца;
- обеспечить возможность проверки одного блока из комплекса БОИК;
- создать возможность эксплуатации в зависимости от состояния изделия;
- обеспечить глубину диагностики неисправностей до конструктивно сменного модуля.

Процесс контроля авиационных приборов, несмотря на его дороговизну и трудоемкость, зачастую имеет решающее значение эффективности использования прибора. В конечном счете контроль сводится к выбору соответствующего решения, принимаемого относительно дальнейшего использования объекта контроля [2]. Эти решения могут быть следующие:

- 1) использовать контролируемый объект по назначению;
- 2) признать объект контроля неисправным и требующим ремонта или списания;
- 3) осуществить подстройку и регулировку объекта.

Таким образом, система контроля как система принятия решения является многоальтернативной. Ошибки решения в реальных случаях, безусловно, могут иметь место и необходимо их свести к минимуму, чтобы принимаемые решения были идеальными. Был выбран тестовый кон-

троль, при котором на объект контроля подаются специальные тестовые воздействия, а по выходным сигналам определяется его реакция на входные воздействия и определяется в допуске ли параметр. Допуски на параметры установлены в технических условиях (ТУ) на объект контроля. При формировании режима контроля стоит придерживаться формирования наиболее вероятного режима работы прибора непосредственно при выполнении той или иной задачи. Реализация режима работы объекта достигается приведением его в начальное рабочее состояние и подачей на входы тестовых сигналов. Реакция объекта контроля выражается в виде выходных сигналов, которые измеряются аппаратурой контроля. Иногда целесообразно одновременно с оценкой показателей качества в процессе контроля решать и задачу диагностики системы, хотя при добавлении к контролю функции диагностики, естественно, возрастает количество тестовых сигналов и проверяемых параметров. Поэтому рациональнее будет сначала произвести полный контроль комплекса в выбранном режиме, проанализировать выявленные отказы и на основании их произвести диагностику. Полноту диагностики необходимо ограничить на уровне определения отказавшего блока и, если это возможно, модуля в его составе.

По своей структуре БОИК состоит из четырех блоков (рис.1). Блок питания (БП) формирует из бортовой сети ЛА необходимые напряжения для работы БОИК. Имитатор контрольного сигнала (ИКС) имитирует тепловой объект в процессе предполетной проверки. Оптико-механический блок (ОМБ) состоит из креновой платформы (КП) и оптического блока (ОБ) и осуществляет непосредственно захват ИК-потока. Блок вычислителя (БВ) осуществляет управление работой ОМБ, обработку данных, перевод их в код для отправки на ЛА и обработку принятого кода с ЛА. БОИК имеет встроенную систему контроля, которая формирует сигналы исправностей блоков («Испр. БП», «Испр. ОМБ», «Испр. БВ»), на основе сигнала с ФПУ формирует сигнал «БОИК готов». Но данная система контроля формирует сигналы исправности на основании только лишь включения блоков и не содержит информации о состоянии входящих в блок элементов и о состоянии параметров изделия. Для получения этой информации и проводятся проверки при помощи НСКД.

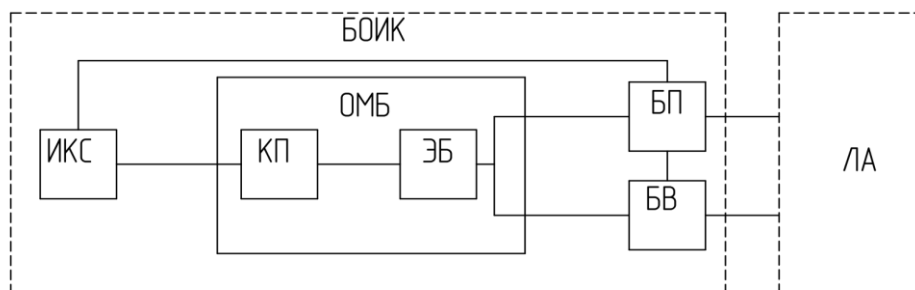


Рис. 1. Структурная схема БОИК

Для оперативного контроля БОИК будет использоваться штатное подключение к носителю, поэтому в качестве параметров контроля будут использоваться все доступные выходные сигналы (табл.), а в качестве тестов – входные (табл. 1).

Таблица 1

Перечень параметров контроля

№ п/п	Параметр	Тип параметра
1	Видеосигнал	Аналоговый сигнал
2	Координаты Ц	Цифровой сигнал
3	Питание РРУ	Аналоговый сигнал
4	НС и КС	Цифровой сигнал
5	Импульсы стр. развертки	Цифровой сигнал
6	Напр. верт. развертки	Аналоговый сигнал

7	Мн.Ц.	Цифровой сигнал
8	Захват	Цифровой сигнал
9	Сигналы исправности	Цифровой сигнал
10	Изделие готово	Цифровой сигнал
11	Контрольные коды	Цифровой сигнал

Таблица 2

Перечень тестовых сигналов

№ п/п	Тестовое воздействие	Тип тестового воздействия
1	Координаты ЦУ	Цифровой сигнал
2	Угол крена	Аналоговый сигнал
3	Тактовые импульсы	Цифровой сигнал
4	Ручная регулировка усиления (РРУ)	Аналоговый сигнал
5	Контроль	Цифровой сигнал
6	2 Режим	Цифровой сигнал
7	ЦУ	Цифровой сигнал
8	Ввод	Цифровой сигнал
9	Сброс	Цифровой сигнал
10	Контрольные коды	Цифровой сигнал

Цифровые коды тестов (табл.3), и параметров (табл. 4) состоят из адресной и информационной частей. В ходе проверки необходим постоянный мониторинг изменения параметров в зависимости от изменения одного или нескольких тестов. Необходимые наборы тестов для различных проверок уже записаны в память НСКД и оператор лишь выбирает нужную проверку, но в ПО реализуется возможность и ручного изменения тестов для проверки. Для анализа цифровых кодов используется сигнатурный анализ, основанный на преобразовании длинных последовательностей двоичных сигналов в двоичное число, которое сравнивается с записанным в памяти эталоном. Допуски на аналоговые параметры хранятся в памяти НСКД и при необходимости могут быть изменены оператором. Аналоговые тесты имеют собственные допуски, а потому также контролируются НСКД перед началом каждой проверки.

Таблица 3

Структура тестовых цифровых кодов

Наименование	Состав слова																											
	Разряды информационной части															Разряды адресов												
	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1				
Код ЦУ по УМ	Зависит от задаваемого угла (табл. 5)															0	0	0	0	0	0	1	1	1	0	0	0	1
Код ЦУ по АЗ	Зависит от задаваемого угла (табл. 5)															0	0	0	*	0	1	0	1	1	1	1	1	1
Команды управления	0	0	0	0	*					0	0	0	0	0	0	0	1	1	0	0	0	1	0	1				
Контрольный код 1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1	1	0	0	1				
Контрольный код 2	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	1				

\*При включении команд «ЦУ», «2 Режим», «Ввод», «Сброс» и «Контроль» выдаются единицы соответственно в 17, 18, 19, 20-м разрядах 13-го слова и в 9-м разряде 12-го слова.

Таблица 4

Структура цифровых кодов параметров

Наименование	Состав строка																							
	Разряды информационной части															Разряды адресов								
	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1
	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0									

Код Ц по АЗ	Зависит от угла БОИК (табл. 5)	*	1	1	0	1	0	1	1	1	1										
Код Ц по УМ	Зависит от угла БОИК (табл. 5)	*	0	1	0	1	1	0	0	0	1										
Контрольный код 3		1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	1	1	0	1
Контрольный код 4		0	1	0	1	0	1	0	1	0	1	0	1	1	1	0	1	1	0	1	1

\* В 9, 10, 11-м разрядах Код Ц по АЗ содержатся соответственно сигналы «Испр. БП», «Испр. ОМБ», «Испр. БВ», «БОИК готов»; в 11 и 12-м разрядах Код Ц по УМ – сигналы «Захват» и «Мн.Ц.». Единицы свидетельствуют о наличии данных сигналов.

Таблица 5  
Зависимость номера и цены разрядов кодов углов по АЗ и УМ.

Номер разряда	24	23	22	21	20	19	18	17	16	15	14	13
Цена разряда	±	90°	45°	22°30'	11°15'	5°37'	2°48,5'	1°24,4'	42,2'	21,1'	10,5'	5,25'

Для формирования тестовых сигналов и анализа параметров была разработана функциональная схема НСКД (рис.2). После выбора режима проверки система формирует ряд тестов по данным из памяти и проводит анализ параметров изделия, по окончании проверки формирует вердикт относительно дальнейшего использования БОИК. При этом в ходе одной проверки характеристики параметров записываются в память и сравниваются в ходе выполнения проверки. Это позволяет осуществить функцию прогнозирования состояния.

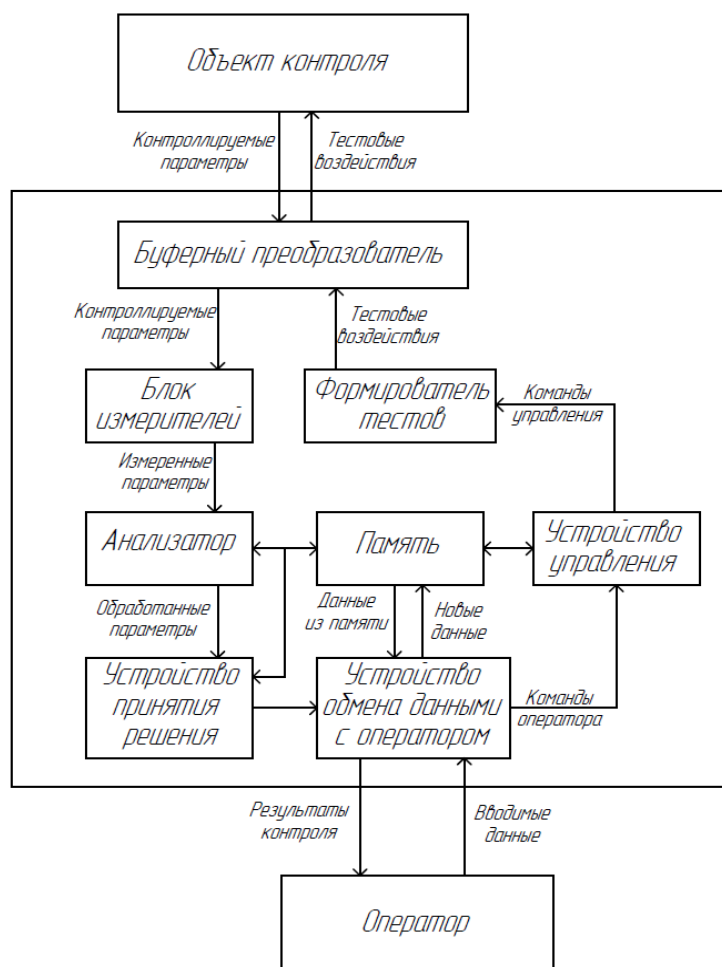


Рис. 2. Функциональная схема наземной системы контроля и диагностики

Использование НСКД позволяет производить не только все виды испытаний после продления назначенного срока службы и проверки в ходе эксплуатации, а также при помощи функции прогнозирования внедрить эксплуатацию по состоянию, которая заметно сократит затраты на периодические ремонты и повысит эксплуатационную надежность БОИК. НСКД реализует все требуемые для проверки изделия функции и представляет собой систему, которая может симулировать реальную задачу для БОИК.

### **Библиографический список**

1. Руководство по технической эксплуатации бортового оптического измерительного комплекса БОИК АЖ2.007.006 РЭ.
2. *Иванов Ю.П., Никитин В.Г., Чернов В.Ю.* Контроль и диагностика измерительно-вычислительных комплексов: Учебное пособие. СПбГУАП. СПб, 2004. 98 с.

### **УДК 681.2**

**М. С. Мазиков**

студент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**А. А. Тарасенков**

ассистент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов, научный руководитель

## **ИСПОЛЬЗОВАНИЕ УЛЬТРАЗВУКОВЫХ ДАЛЬНОМЕРОВ НА БОРТУ БПЛА**

Беспилотные летательные аппараты в последнее время развиваются очень динамично. Существует множество их видов, как по размерам, так и по характеру полета (самолетного, вертолетного типов). Для функционирования они все нуждаются в различных приборах для связи с внешним миром. Это необходимо для осуществления управления удаленно, что позволяет использовать их в опасных и труднодоступных местах, не подвергая опасности оператора. Современный уровень технологий позволяет спроектировать воздушное судно, способное работать полностью автономно, получив необходимые данные непосредственно перед вылетом. Круг задач, которые можно решать таким образом широк: от доставки необходимых грузов и наблюдения до использования в качестве полноценной боевой единицы в армии.

Половина всех летных происшествий с воздушными судами происходит при заходе на посадку и при ее осуществлении. Поэтому для ее облегчения дополнительно используют различные средства. Примером можно привести высотомеры малых высот, присутствие которых на всех воздушных судах обязательно. Часто на борту использую радиовысотомеры малых высот. В таких же системах нуждаются и беспилотные летательные аппараты.

В качестве высотомера малых высот можно использовать ультразвуковой дальномер. Он работает на временном принципе: нахождение расстояния до объекта путем измерения времени между излученной и принятой ультразвуковыми волнами. Ультразвуковая волна является зондирующим сигналом. Ультразвуковым колебаниям соответствуют частоты в диапазоне от 16-20 кГц до  $10^8$  Гц [3]. Зону распространения можно представить в виде конуса. В связи с этим и возникают ограничения в использовании. На скорость распространения ультразвуковой волны напрямую влияет вязкость среды. Чем она выше, тем выше скорость распространения волны и предельная дальность. Звуковые волны подразделяются по типу волн: продольные, поперечные, изгибные, крутильные – в зависимости от условий возбуждения и распространения. В жидкостях и газах распространяются только продольные волны, в твердых телах могут возникать также поперечные и другие перечисленные типы волн. В продольной волне направление колебаний частиц совпадает с направлением распространения волны, поперечная волна распространяется



перпендикулярно направлению колебаний частиц [2]. В зависимости от формы излучателя и характера его движения могут быть цилиндрические (создаются цилиндром, который сжимается и расширяется в направлении своего радиуса) и сферические волны (точечный излучатель – пульсирующий шарик, размеры которого малы, по сравнению с длиной излучаемой волны).



При установке УЗ дальномера необходимо учитывать некоторые особенности:

1. Турбулентность от винтов ослабляет и изменяет направление испускаемого датчиком сигнала. Измерения, проведенные инженерами компании MaxBotix на квадрокоптере, показали, что в зависимости от положения датчика величина ослабления сигнала изменялась больше, чем в десять раз. Для того, что бы этого не происходило, необходимо устанавливать излучатель и приёмник в месте геометрического центра аппарата под несущей рамой.

2. Акустические шумы. Они не ослабляют, а наоборот, усиливают сигнал, занижая показания. Защитить датчик можно с помощью пористой резины, располагая его под электронными узлами судна.

3. Кондуктивные электрические помехи. Усилители токов, управляющие скоростью вращения моторов, вызывают провалы мощности в сети питания. Так же помехи вносят радиочастотные блоки (ZigBee) при передаче данных. Зашумленные линии питания приводят к сбоям в работе датчиков. Для их устранения достаточно поставить сетевой фильтр, состоящий из конденсатора на 100 мкФ и резистора на 10–100 Ом.

4. Индуктивные электрические помехи. Электромагнитные шумы вызывают переключение больших токов при запуске двигателей и радиочастотных цепей высокой мощности. Их мощность достигает 260 мВ, их влияние возможно на некоторые линии с аналоговым выходом. При возникновении такой проблемы достаточно использовать экранированные кабели, заземленные только со стороны микроконтроллера (снижения шума с 260 до 10мВ).

5. Вибрация несущей рамы. Может возникнуть вибрация рамы, на которую установлен датчик. Для предотвращения ее влияния необходимо использовать пористую резину или подобные материалы.

Ультразвуковые дальномеры обладают рядом преимуществ, по сравнению с другими дальномерами. Они имеют элементарную конструкцию, низкую цену и меньшее влияние от подстилающей поверхности по сравнению с лазерными дальномерами. Выигрывают по массогабаритным характеристикам и точности у радиовысотомеров.

Недостатками является меньшая точность по сравнению с лазерными дальномерами и следующими ограничениями применения:

1. при повышении частоты путем снижения длины излучаемой волны увеличивается чувствительность к более мелким объектам;
2. частичные отражения (паразитный эхо сигнал) могут исказить результаты измерений (наличие криволинейных или наклонных к направлению излучения сигнала поверхностей);
3. измерение расстояния до объекта, состоящего из звукопоглощающего, изолирующего материала или имеющего тканевую поверхность будет менее точным;
4. ослабления отраженного сигнала от объекта, имеющего небольшие размеры;
5. возникновение паразитного эха сигнала при высокой влажности и осадках (снег, дождь);
6. сильный ветер может повлиять на распространение волн, что приводит к дополнительным ошибкам [3].

Помехи возникают от условий окружающей среды: наличие различных взвесей в воздухе (туман, дым, пар, пыль), осадков (снег дождь, град), изменение температуры, воздушные потоки. Все эти факторы могут влиять на точность измерения, но для решения задачи определения высоты необходимо учитывать температуру и наличие препятствия для распространения ультразвуковой волны, так как остальные помехи не так существенны.

В качестве примера можно привести прибор, исключая жесткую посадку, позволяющий лететь на заданной высоте (до 2,4 м) Distance Sensor (табл. 1). Имеется 2 режима реагирования:

- 1) пропорциональная реакция на высоту по скорости;
- 2) пропорциональная реакция на скорость в зависимости от высоты.

Первый режим используется для отталкивания от земли на самолетах и вертолетах и удерживание последних на заданной высоте. Второй режим делает возможным полет самолета на заданной высоте (1,7 м от земли).

Таблица 1

Характеристики прибора Distance Sensor

Питание	Потребляемый ток	Направленность	Контролируемый уровень высот	Размеры датчика	Вес
4,5 – 5,5 В	25 мА	60°	0,2–2,4 м	65×18×18 мм	16 г

Таблица 2

Примеры ультразвуковых излучателей/приёмников

Наименование	Номинальная рабочая частота, кГц	Чувствительность, дБ	Направленность (град.), давление	Область детектирования, м	Диапазон рабочих температур, °С	Разрешение, мм
MA40B8R	40	-(60±3)	50 симметричн.	0.2-6	-30..+85	9
MA40S4R	40	-(60±3)	80 симметричн.	0.2-4	-40..+85	9
MA80A1	75±5	-47	7	0.5-5	-10..+60	4

Максимальная дальность зависит от характеристик ультразвукового приемника и излучателя. Зная мощность излучения, чувствительность датчика и коэффициент затухания в воздуш-

ной среде (большинство времени работы), её можно найти. Мощность излучателя 120 дБ (ультразвуковые датчики излучатели пьезокерамические серии МА40), приёмника 63 дБ и коэффициенты затухания в воздушной среде  $\alpha = 1,6 \cdot 10^{-2}$  Нп/мм [3] ( $1,842 \cdot 10^{-3}$  дБ/мм). Разность между мощностью излучателя и приёмника есть величина, на которую может ослабнуть сигнал. Разделив эту величину на коэффициент затухания получаем расстояние в миллиметрах, пройдя которое, ультразвуковая волна будет обладать достаточной мощностью для реагирования приёмника. Получаем 30940 мм  $\approx$  31 м. При учёте, что ультразвуковой волне необходимо пройти двойную дистанцию, получаем 15,5 м – теоретическое максимальное расстояние при использовании данных компонентов.

### Библиографический список

1. Лазерные приборы и методы измерения дальности : учеб. пособие / В.Б. Бокшанский, Д.А. Бондаренко, М.В. Вязовых, И.В. Животовский, А.А. Сахаров, В.П. Семенов ; под ред. В.Е. Карасика. М.: Изд-во МГТУ им. Н.Э. Баумана, 2012. – 92, [4] с.
2. Радиотехнические навигационные средства самолетовождения и посадки. В 2 ч. Ч. 1: учеб. пособие. / [М.А. Ковалев и др.]. – Самара: Изд-во Самар. гос. аэрокосм. ун-та, 2011. – 76 с.
3. Балдев Р., Раджендран В., Паланичами П. Применение ультразвука. М.: Техносфера, 2006.
4. Кайно. Акустические волны: Устройства визуализации и аналоговая обработка сигналов: Пер. с англ. М.: Мир, 1990.

УДК 519.216

**А. Г. Максимов**

магистрант кафедры аэрокосмических приборов и измерительно-вычислительных комплексов.

**В. Я. Мамаев**

доц., канд. техн. наук – научный руководитель

## МАТЕМАТИЧЕСКАЯ МОДЕЛЬ РАЗВОРОТА ТЯЖЕЛОГО САМОЛЕТА ДЛЯ ИМИТАЦИИ В ТРЕНАЖЕРЕ

Эффективность функционирования авиационной транспортной системы (АТС) определяют: безопасность, регулярность и экономичность. В свою очередь, безопасность и экономичность во многом зависят от качества проводки ЛА по заданному маршруту.

Ранее предполагалось, что разворот ЛА происходит по окружности [2]. На самом деле движение происходит по сложной кривой, из-за того, что ЛА, как инерционный объект мгновенно входить в разворот не может. Для приобретения соответствующих навыков выполнения разворотов в математические модели, реализуемые на авиационных тренажерах (АТ), необходимо внести поправки.

Для осуществления безопасных маневрирования и более точного выполнения разворотов с учетом инерционности самолета предлагается внедрить в практику тренажерной подготовки методику Молоканова Г.Ф. [1]. В своей работе он показал, что если не учитывать время ввода в разворот и вывода из него, то это приведет к большим погрешностям, что особенно сказывается в зонах активного маневрирования (например, при заходе на посадку).

Развороты с постоянным креном

При неучете участков ввода в разворот и вывода из него и отсутствии ветра траектория разворота будет окружностью радиуса  $R$ , уравнения которой

$$dX = V dt \cos \psi;$$

$$dZ = V dt \sin \psi.$$

При наличии ветра, считая его в районе разворота постоянным, уравнения траектории движения ЛА в прямоугольной системе  $XOZ$  земных координат, ось абсцисс которой совмещена с прямолинейным этапом подхода к развороту, запишутся в виде

$$dX = Vdt\cos\psi + Udt\cos\delta,$$

$$dZ = Vdt\sin\psi + Udt\sin\delta,$$

где  $\psi$  – курс, отсчитываемый от оси  $OX$ ;  $U$  – скорость ветра;  $\delta$  – его направление. На рис. 1. отображено движение ЛА с постоянным углом крена по приведенным выше зависимостям.

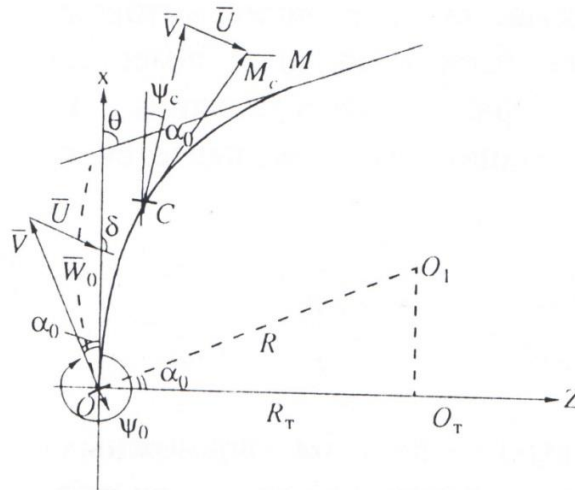


Рис. 1. Вывод основных соотношений разворота с постоянным углом крена

Затягивание процесса ввода в разворот приводит к тому, что величина угла крена оказывается несколько больше  $\gamma_0$ . Различие в этих углах крена составляет единицы, а то и доли градуса. Найти оптимальную его величину можно, варьируя углы крена в пределах от  $\gamma_0$  до  $\gamma_c$  или вблизи указанного интервала.

Общий недостаток разворота с постоянным углом крена заключается в том, что при наличии ветра фактическая траектория ЛА отклоняется от трассы. Поэтому при необходимости строгого следования по траектории заданного радиуса угол крена в процессе разворота должен изменяться по определенному закону.

Сложность выдерживания в процессе разворота постоянного угла крена или изменяющегося по определенному закону вынуждает искать иные способы точного полета по трассе заданного радиуса. При использовании не программного, а позиционного способа управления ЛА по крену, предоставляется возможность его коррекции в зависимости от измеренного отклонения ЛА от трассы и его производной.

Аналогичную жесткую привязку ЛА к криволинейной траектории может обеспечить высокоточная спутниковая навигационная система. Это позволит существенно повысить точность и безопасность выполнения экипажами ЛА маневров как в районе аэропортов, так и при разворотах на очередном этапе маршрута.

Развороты с постоянным радиусом

Найдем закон управления ЛА по крену для повышения точности полета по дуге окружности заданного радиуса. Для этого необходимо найти штилевую траекторию движения, которая за счет ветра отобразится в земной системе координат окружностью требуемого радиуса:

$$X = R\sin p;$$

$$Z = R(1 - \cos p).$$

Уравнение трассы радиуса  $R_T$  в земной системе координат  $XOZ$  запишется в виде

$$(X - R_T)^2 + Z^2 = R_T^2,$$

где

$$\sin\psi = \sin(p - \alpha) = \sin p \cos \alpha - \cos p \sin \alpha = \sin p \sqrt{1 - \sin^2 \alpha} - \cos p \frac{U}{V} \sin(\delta - p);$$

$$t_p = R_T \frac{P_1 - P_0}{W_T}.$$

При записи уравнений для участков ввода в крен ЛА и вывода из него предполагалось, что угловая скорость  $\omega$  мгновенно возрастает до определенного значения. Такое допущение противоречит динамике процесса ввода в крен особенно тяжелого ЛА, поэтому необходимо оценить те погрешности, которые возникнут при расчете по кинематическим уравнениям траектории движения ЛА на участке его ввода в разворот.

Траектория участка ввода с учетом ветра будет клотоидой, координаты которой изменяются по закону

$$X = Vt - \frac{(g\omega)^2}{40V} t^5 + Ut \cos \delta,$$

$$Z = \frac{g\omega}{6} t^3 - \frac{(g\omega)^3}{336V^2} t^7 + Ut \cos \delta.$$

Без учета динамики процесса результаты кинематических расчетов за счет большей угловой скорости ввода ЛА в крен оказываются завышенными по курсу в среднем на  $0,5-1^\circ$ , по углу крена на первых секундах – на  $3-4^\circ$  и по координате  $Z$  к его окончанию – на 5-10 м. Эти погрешности можно существенно уменьшить, подобрав зависимость для угловой скорости  $\omega_x$  ввода ЛА в крен. После их интегрирования можно получить более точные значения координат  $X$  и  $Z$  на участке ввода ЛА в разворот. Расчеты показывают, что погрешности определения координаты  $Z$  не превышают 2-3 м, но выражения оказываются довольно громоздкими.

Решение задачи точного выполнения разворотов по заданным криволинейным траекториям, требующим полета с изменяющимся во времени углом крена, должно быть возложено на навигационно-пилотажный комплекс, учитывающий динамику движения ЛА и позволяющий автоматически парировать его отклонения от заданной траектории маневра.

### Библиографический список

1. Молоканов Г.Ф. Кинематические алгоритмы управления летательным аппаратом при выходе на цель и развороте на новый этап // Изв. РАН. Теория и системы управления. 1995. №1.
2. Хиврич И.Г., Белкин А.М., Миронов Н.Ф. Воздушная навигация: учеб. пособ. для вузов / И.Г. Хиврич, Н.Ф. Миронов, А.М. Белкин. М.: Транспорт, 1984. 328с.

УДК 656.073.9

**Д. А. Мелеховец**

студент кафедры системного анализа и логистики

**Н. Н. Майоров**

доц., канд. техн. наук – научный руководитель

## ВОПРОСЫ ВЫБОРА РАЦИОНАЛЬНОГО МАРШРУТА ТРАНСПОРТИРОВКИ НЕГАБАРИТНЫХ ГРУЗОВ ЖЕЛЕЗНОДОРОЖНЫМ ТРАНСПОРТОМ

Транспортировка негабаритных грузов является сложной задачей, но на сегодняшний день является довольно распространенной и актуальной, так как требует особых условий организации подготовки маршрута, крепления, погрузки и выгрузки крупногабаритного груза и непосредственно самой перевозки.

Для перевозки крупногабаритных грузов на железнодорожном транспорте используются специальные платформы и транспортеры с большой грузоподъемностью.

Крупногабаритный груз – это груз, превышающий предельные характеристики, и который не возможно погрузить на подвижной состав. негабаритным является груз, который при размещении на открытом подвижном составе, находящемся на прямом горизонтальном участке пути (при совпадении в одной вертикальной плоскости продольных осей вагона и пути), превышает очертание габарита погрузки, или его геометрические выносы в кривых за пределы габарита погрузки превышают геометрические выносы в соответствующих кривых расчетного вагона [1]. Существуют различные зоны и степени негабаритности, которые зависят от размера груза и его расположения на подвижном составе. Каждая из зон соответствует высоте от уровня головки рельса, в пределах которой груз выходит за погрузочный габарит. Каждая зона имеет от 3 до 6 степеней и груз будет сверхнегабаритным, если он выйдет по горизонтали за пределы очертаний степеней.

Груз может иметь нижнюю, верхнюю, боковую и вертикальную сверхнегабаритность, по которой затем определяются условия приема груза к перевозке.

При перевозке будут учитываться аспекты всего маршрута, габариты приближения строений, расстояния между осями смежных путей прямых и искривленных участков перегона, характеристики промежуточных станций.

Для решения задачи организации транспортировки негабаритного груза в любом случае необходимо использовать программное обеспечение. На рынке России существует программное обеспечение «СТМ», включающее в себя комплекс программ «Rail-Офис» для железнодорожных перевозок и логистики. Программа «СТМ» позволяет произвести расчет перевозки с учетом необходимых условий и характеристик, что вручную сделать почти невозможно.

«Rail-Тариф» является удобным и точным инструментом для расчета железнодорожной провозной платы за перевозку грузов по территории России, стран СНГ, Латвии, Литвы, Эстонии и Грузии [2].

Процесс расчета максимально автоматизирован. Удобный и современный интерфейс позволяет в одном окне просматривать исходные данные, результаты и подробности расчета с возможностью перехода на регламентирующий документ. Предусмотрены сохранение, печать и экспорт данных [2].

Рассчитаем перевозку негабаритного груза на примере яхты.

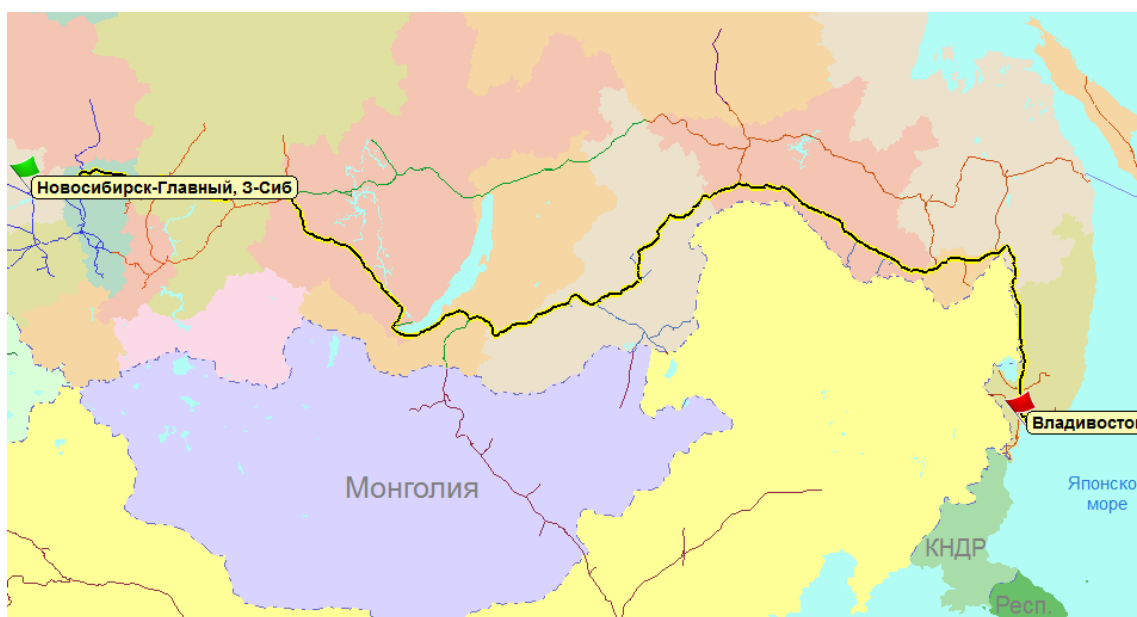


Рис. 1. Оконная форма работы в программе «СТМ» – «Rail-Атлас»

Эта программа позволяет проложить маршрут из пункта отправления в пункт назначения, рассмотреть характеристики станций.

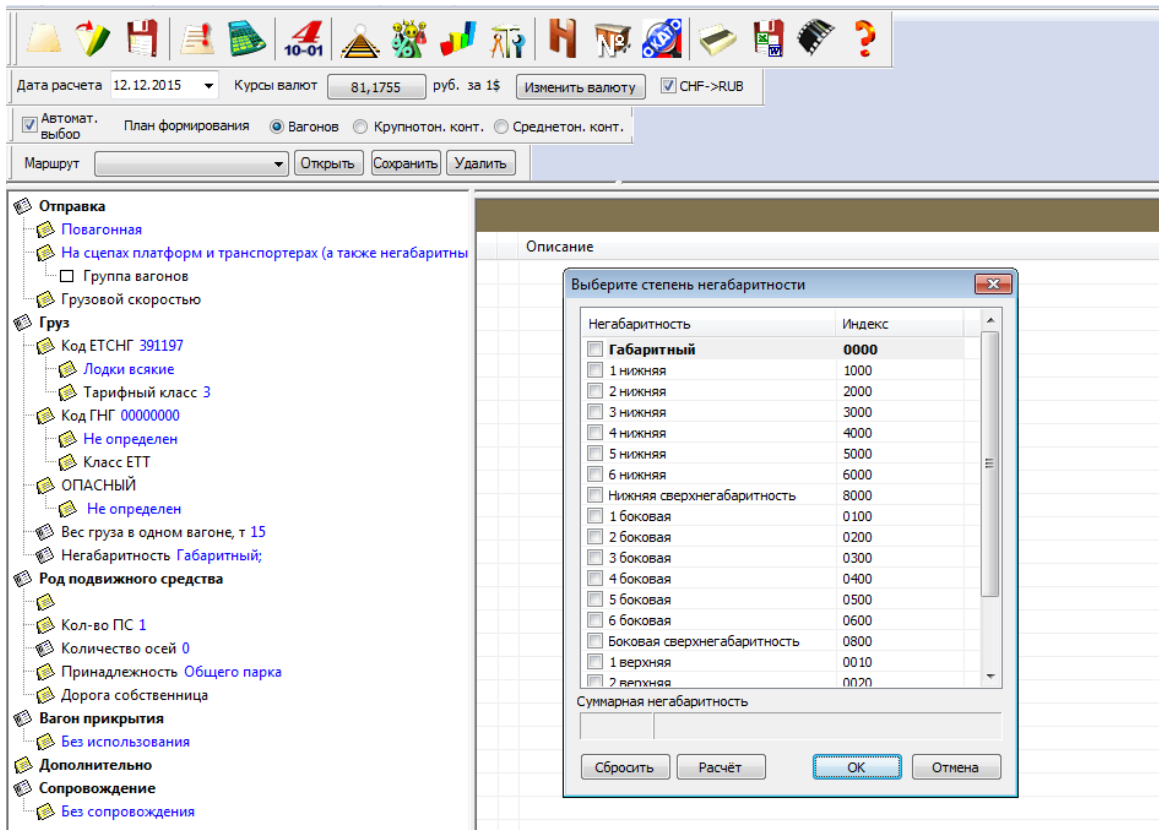


Рис. 2. Окнальная форма работы в программе «СТМ» – «Rail-Тариф»

Эта программа имеет возможность выбора степени негабаритности груза.

60 сут. (на 5958 км), Суточный пробег 100 км		
2 сут. на операции, связанные с отправлением и прибытием		
1 сут. на отправлении негабаритных грузов с 1 по 5 степень негабаритности		
Итого. Срок доставки 63 сут.		
Расчетный вес 7,00 т.		
<b>Схема 35-Перевозка негабаритных грузов на платформах и в полувагонах</b>	<b>286920,00</b>	<b>RUB</b>
1,74 Коэффициент для грузов третьего класса	499240,80	RUB
1,04 Коэффициент на перевозку грузов третьего класса	519210,43	RUB
3,475 Коэффициент индексации	1804256,25	RUB
<b>Итого</b>	<b>1804256,25</b>	<b>RUB</b>
337,00 Скидка с общего тарифа на универсальные вагоны	1803919,25	RUB
<b>Итого</b>	<b>1803919,00</b>	<b>RUB</b>
<b>Сумма за все вагоны</b>	<b>3607838,00</b>	<b>RUB</b>
<b>Итого: Провозная плата</b>	<b>3607838,00</b>	<b>RUB</b>
18,00 НДС	649410,84	RUB
<b>Итого</b>	<b>4257248,84</b>	<b>RUB</b>
<b>Договорной сбор за сменное сопровождение и охрану</b>	<b>16632,00</b>	<b>RUB</b>
<b>Сумма за все вагоны</b>	<b>33264,00</b>	<b>RUB</b>
<b>Итого: Охрана</b>	<b>33264,00</b>	<b>RUB</b>
18,00 НДС	5987,52	RUB
<b>Итого</b>	<b>39251,52</b>	<b>RUB</b>
<b>Итого охрана без НДС</b>	<b>33264,00</b>	<b>RUB</b>
<b>Итого охрана (НДС)</b>	<b>5987,52</b>	<b>RUB</b>
<b>Итого охрана</b>	<b>39251,52</b>	<b>RUB</b>
<b>Стоимость перевозки без НДС</b>	<b>3641102,00</b>	<b>RUB</b>
<b>НДС</b>	<b>655398,36</b>	<b>RUB</b>
<b>Стоимость перевозки</b>	<b>4296500,36</b>	<b>RUB</b>
<b>Итого за тонну без НДС</b>	<b>260078,71</b>	<b>RUB</b>
<b>Итого за тонну</b>	<b>306892,88</b>	<b>RUB</b>
<b>Итого за тонно-километр без НДС</b>	<b>43,65</b>	<b>RUB</b>
<b>Итого за тонно-километр</b>	<b>51,51</b>	<b>RUB</b>

Рис. 3. Окнальная форма работы в программе «СТМ» - «Rail-Тариф»

Эта программа позволяет рассмотреть все подробности расчетов по выбранному маршруту.



Вывод:

Сегодня для расчета транспортировки крупногабаритного груза используются специализированные программные средства. Комплекс программ «Rail-Офис» позволяет выполнять большой объем решения вопросов перевозки тяжеловесных и негабаритных грузов.

Программа имеет возможность автоматизированного расчета тарифа на перевозку, прокладки маршрута, а также содержит справочную информацию и правовую базу документов.

### Библиографический список

1. Инструкция по перевозке негабаритных и тяжеловесных грузов на железных дорогах государств – участников СНГ, Латвийской Республики, Литовской Республики, Эстонской Республики [Электронный ресурс] – Режим доступа: <http://docs.cntd.ru/document/902393263>.
2. Программное обеспечение для ж/д логистики и ВЭД «СТМ» [Электронный ресурс] – Режим доступа: <https://www.ctm.ru>.

### УДК 629.7.018.3

**Е. А. Мелидин**

студент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**В. С. Егоров**

магистрант кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**А. П. Григорьев**

ассистент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**В. Я. Мамаев**

доц., канд. техн. наук научный руководитель

## ИНТЕЛЛЕКТУАЛЬНЫЙ ЭЛЕКТРОННЫЙ УЧЕБНИК ДЛЯ ПОДГОТОВКИ АВИАЦИОННЫХ СПЕЦИАЛИСТОВ

В настоящее время мультимедийные технологии считаются одними из наиболее перспективных, популярных и распространенных педагогических информационных технологий, позволяющих сделать обучение более эффективным. Были рассмотрены вопросы [1–5] проектирования электронных учебных пособий, являющихся неотъемлемой составной частью мультимедийных технологий на этапе теоретического обучения, а также сформулированы основные подходы к реализации [1] интеллектуального электронного учебника (ИЭУ). Таким образом, в рамках данной статьи целесообразно рассмотреть роль и место ИЭУ в образовательном процессе, а также проанализировать технико-технологические аспекты при непосредственном его проектировании.

Отметим, что использование ИЭУ в обучающем процессе позволяет реализовать сквозную непрерывную подготовку высококвалифицированных авиационно-инженерных кадров и обеспечивает простоту обучения [1].

Развитие и формирование концепции ИЭУ можно условно разделить на следующие этапы:

- оцифровывание бумажных материалов. Создание полноценных цифровых пособий, доступных лишь на локальных персональных компьютерах;
- перенос оцифрованных материалов в область интернет-ресурсов, формирование разветвленной структуры (семантической навигации);
- внедрение модулей проверки и закрепления теоретических знаний в виде открытых/закрытых форм тестирования знаний (рис.1) после изучения предметной области учебного материала;



– разработка интегрируемой в ИЭУ «медиа-библиотеки» для размещения методико-дидактических материалов обучающего процесса (рис. 2). Под термином «медиа-библиотека» здесь и далее будем понимать с медиаданные, (документы, изображения, презентации, аудиофайлы и видео) и менеджера, который облегчает систематизацию хранимого материала и доступ к нему;

– завершающим шагом становится создание активного клиент-серверного взаимодействия с обучаемым (рис. 3), а также внедрение элементов интеллектуальности (адаптивности) [1].

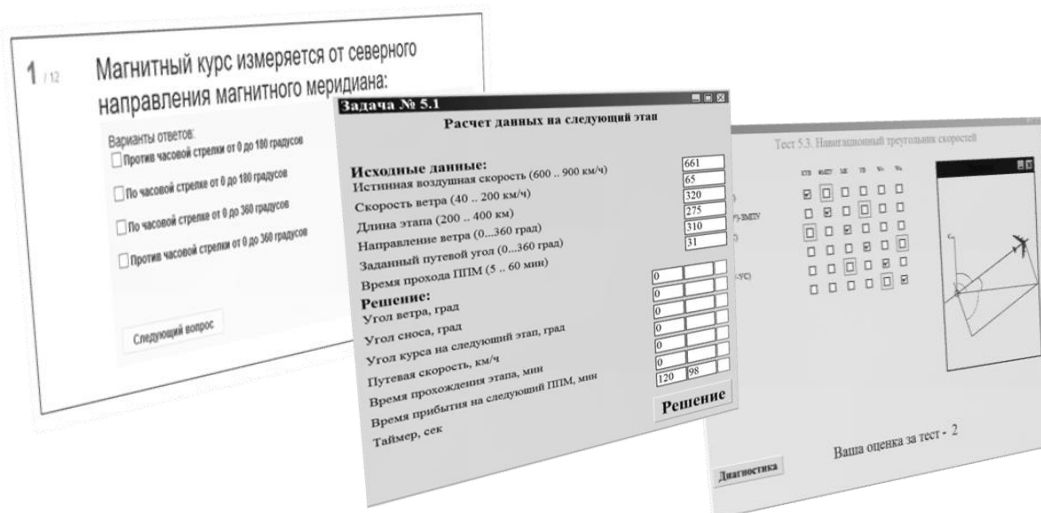


Рис. 1. Формы тестирования

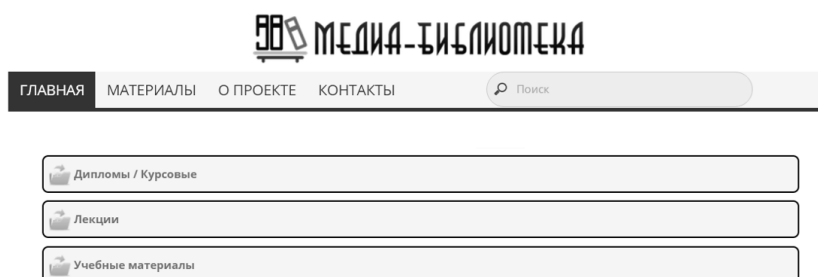


Рис. 2. Медиа-библиотека

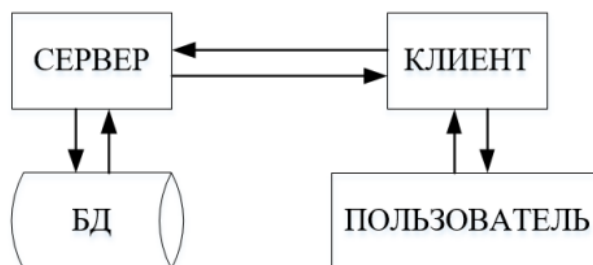


Рис. 3. Клиент-серверное взаимодействие

Достаточно часто на изучение определенного материала выделяется недостаточное количество времени для освещения всей темы целиком, не всегда есть возможность разобраться с материалом в полном объеме, тем самым ИЭУ существенно помогает облегчить задачу не только понимания текста, но и предоставляет возможность обучаться столько времени, сколько необходимо для полного анализа полученной информации. Основная задача ИЭУ [1] – частичная разгрузка преподавателя за счет реорганизации классно-урочной формы проведения аудиторных занятий (проведение занятий в формате лекций с последующим «отсылком» к ИЭУ) и повышение

эффективности (по критериям – скорость и глубина усвоения) процессов передачи информации, что осуществляется за счет свойства «интеллектуальности» (адаптация учебного материала под характерные особенности восприятия каждого отдельного обучаемого). При частичной трансформации классно-урочной формы с пассивным участием в обучающем процессе объекта обучения (обучаемый) во внеаудиторную самостоятельную работу, при которой объект становится активным и мотивированным (за счет вариативности работы с ИЭУ, адаптации к индивидуальным особенностям обучаемого, построения индивидуальных маршрутов обучения) субъектом обучающего процесса, появляется возможность организовать такие эффективные приемы работы, как опережающая самостоятельная работа, проблемно-поисковые и кейс-технологии и пр., а также дать преподавателю возможность быть не просто «передатчиком теоретических знаний и контролером усвоения», а вести диалог с уже подготовленными в рамках самостоятельной работы с ИЭУ обучаемыми на более высоком уровне, обращая внимания на проблемные аспекты изучаемой дисциплины, указывая на «тонкие нюансы», освещение некоторых спорных дискуссионных моментов, которые возникают в рамках изучения любой из дисциплин.

Проанализировав роль и место ИЭУ в учебном процессе, рассмотрим технико-технологические аспекты при его непосредственном проектировании. При разработке ИЭУ решаются такие задачи как:

- электронное представление (целые коллекции изображений, текстов и данных, сопровождающихся звуком, видео, анимацией и другими визуальными эффектами) предметной области на базе методико-дидактических материалов [1];
- разработка элементов психофизиологической диагностики[2](тест на модальность) для адаптации процесса передачи учебного материала каждому конкретному обучающемуся с ИЭУ;
- разработка алгоритма работы ИЭУ (определение, как выдавать информацию);
- представление клиент-серверной реализации ИЭУ;
- разработка системы контроля знаний на базе адаптивных тестовых заданий[3].

В соответствии с концепцией, предложенной авторами данной статьи, ИЭУ состоит из (рис. 4):хранилища учебного материала; системы тестового контроля; программы мониторинга посещения ресурса; интеллектуальной адаптивной системы, учитывающей оптимальный размер и вид подачи учебного материала для того или иного обучаемого.

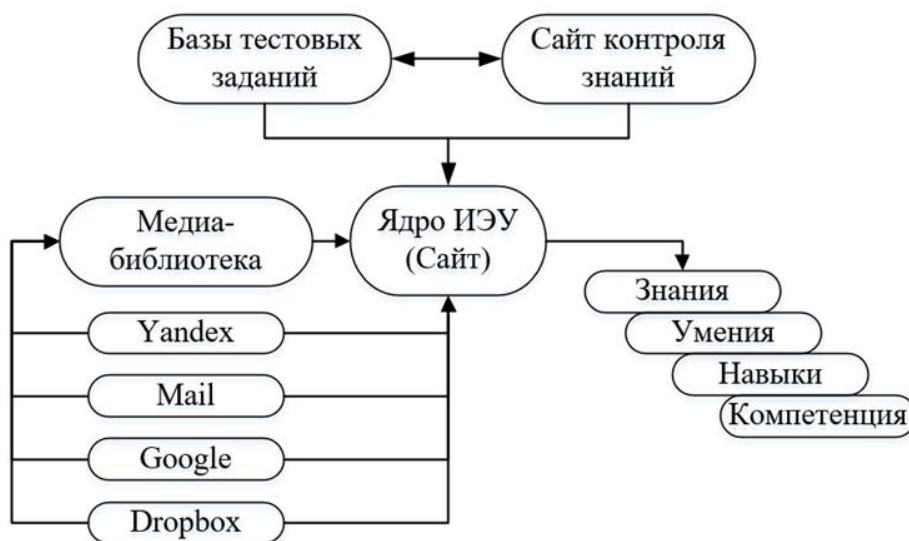


Рис. 4. Состав ИЭУ

С точки зрения технической реализации, основной частью ИЭУ является ядро (сайт), к которому подключаются сторонние базы данных (см. рис. 4). В качестве информационной базы

применяются как базы данных и базы знаний, размещенные на физических носителях, так и банки данных организованные по облачным технологиям. Ядро отвечает за контроль посещаемости, за тестовый контроль знаний, алгоритмы психофизиологической диагностики или подключение отдельных сторонних систем.

Термин «облачные технологии» или в более узком смысле «облачные вычисления» (англ. cloud computing) подразумевает предоставление пользователям удаленного доступа к услугам, вычислительным ресурсам и приложениям (включая операционные системы и инфраструктуру) через сеть Интернет. Преимуществами «облачных вычислений» являются: экономичность за счет отсутствия необходимости покупки и/или модернизации программного обеспечения; отказоустойчивость и безопасность – автоматическое выделение и освобождение необходимых ресурсов в зависимости от потребностей приложения; техническое обслуживание, обновление программного обеспечения производит провайдер услуг; удаленный доступ к данным в облаке позволяет работать из любой точки планеты, где есть доступ в сеть Интернет. Недостатки облачных вычислений связаны с: отсутствием доступа к внутренней инфраструктуре ресурса; потенциальной сохранностью пользовательских данных; необходимостью иметь надежный (стабильный и/или защищенный) и быстрый доступ в сеть Интернет; с потенциально возможной потерей данных в результате неполадок удаленного сервера.

В отличие от электронного учебника основной задачей ИЭУ является не только предоставление доступа к учебным материалам, но и реализация мотивационной составляющей обучающего процесса, что предполагает организацию каждому отдельному обучаемому комфортных (адаптивных, индивидуальных) условий при удаленной работе с учебным материалом, а также дистанционном контроле знаний.

В ИЭУ интегрирован модуль психофизиологической диагностики [2,4], осуществляющий тестирование модальности (доминирующего канала восприятия информации) с целью оптимального перераспределения информационных потоков по каждому из каналов восприятия обучаемого (аудиальный, визуальный, кинестетический) для снижения информационной нагрузки обучаемого, интенсификации степени его познавательной деятельности. Обучаемому предлагается пройти тест на определение модальности (основного канала восприятия информации). Если обучаемый не проходит тест, то ИЭУ не адаптирует учебный материал к особенностям его восприятия, обучаемый сам решает, с какими материалами работать – текстом, аудио или видео [5]. При прохождении входного психологического тестирования обучаемому предоставляется информация о доминирующем канале восприятия, выводятся рекомендации по изучению теоретического материала предметной области в соответствии с модальностью. После получения рекомендаций обучаемый принимает решение – следовать рекомендациям теста или же самостоятельно по собственному усмотрению перейти к требуемому разделу ИЭУ.

Потенциально ИЭУ может быть представлен в виде «каркаса», инвариантного к предметным областям, который позволяет индивидуализировать процесс восприятия, повысить качество передачи теоретического материала, за счет задействования доминирующего канала восприятия каждого отдельного обучаемого, что в конечном счете дает более качественный вид кривой научения (насыщение, хранение, забывание). Создание единого каркаса позволяет ввести ИЭУ во все возможные сферы обучения, повышает качество знаний и снижает нагрузку на преподавателя, дав ему возможность сосредоточиться на передаче высокопрофессиональных знаний студентам.

Задачи, связанные с проектированием ИЭУ являются достаточно нетривиальными, сложноформализуемыми на данный момент слабо изученными, вследствие междисциплинарности аспектов проектирования, разнородности тех или иных задач, решаемых ИЭУ (в том числе и его отдельными блоками) и отсутствия общей идеологии построения, использующей единый аппарат разработки.

#### **Библиографический список**

1. Григорьев А. П., Долгий А. И. Интеллектуальный электронный учебник для теоретической подготовки авиационных специалистов. Педагогическая наука и Современное образование: сб. статей Международ-

ной научно-практической конференции, посвященной дню российской науки. Санкт-Петербург: РГПУ им. А. И. Герцена, 2015. С. 239–242.

2. Григорьев А. П., Смирнов Г. В. «Адаптация по типу восприятия окружающего мира и индивидуализация обучения студентов аэрокосмического профиля /Adaptation by type perception of the world, and individualization of teaching students aerospace profile. Новые образовательные стратегии в современном информационном пространстве: сб. статей международной ежегодной научной интернет-конференции. СПб: РГПУ им. А. И. Герцена, 2015. С. 226–229.

3. Глова В. И., Дуплик С. В. Модели педагогического тестирования обучаемых // Вестник Казан, гос. техн. ун-та им. А.Н. Туполева. 2003 г. №2. С. 74–79.

4. Григорьев А. П., Маров В. Д., Смирнов Г. В. Психофизиологические аспекты организации адаптивного обучения в высшей технической школе. Педагогическая наука и современное образование: сборник статей Международной научно-практической конференции, посвященной дню российской науки. СПб: РГПУ им. А. И. Герцена, 2015. С. 247–249

5. [Электронный ресурс], режим доступа: [http://fit-herzen-conf.ru/statii/285\\_grigoriev.php](http://fit-herzen-conf.ru/statii/285_grigoriev.php), свободный.

УДК 004.72

**В. И. Миненко**

студент кафедры аэрокосмических компьютерных и программных систем

**Е. А. Суворова**

канд. техн. наук, доцент – научный руководитель

## СРАВНИТЕЛЬНЫЙ АНАЛИЗ БОРТОВЫХ ТЕЛЕКОММУНИКАЦИОННЫХ СЕТЕЙ

### Введение

В работе рассматриваются существующие бортовые инфокоммуникационные стандарты SpaceWire, FlexRay, AFDX, рассмотрены их особенности и производится их сравнение.

Важной особенностью современных бортовых информационно-измерительных комплексов является постоянная модернизация существующего оборудования. При этом количество функций, выполняемых бортовыми инфокоммуникационными стандартами, возрастает, что приводит к увеличению количества обрабатываемых данных, поступающих от различных датчиков, что в свою очередь приводит к разработке и внедрению новых стандартов и технологий. В данной работе будут рассмотрены некоторые из них.

### Протокол SpaceWire

Протокол SpaceWire разрабатывался в соответствии с следующими требованиями: высокая скорость передачи информации, малые задержки доставки сообщений, устойчивость к отказам и сбоям, низкое энергопотребление, электромагнитная совместимость, компактная реализация в СБИС, поддержка систем реального времени и системных функций бортовых комплексов[1].

Стандарт SpaceWire представляет собой интерфейс, который может быть реализован под конкретную задачу. Он регламентирует логические протоколы, физические разъемы и кабели, электрические свойства соединений, которые определяют канал связи SpaceWire, архитектуру коммуникационной сети и обеспечивают средства передачи пакетов информации от исходного узла до требуемого узла назначения через масштабируемую коммуникационную сеть (табл. 1).

Информация в каналах передается в полнодуплексном режиме(в обе стороны канала одновременно) со скоростью 2–400 Мбит/с. Информация передается в виде последовательного битового потока, который подвергается DS-кодированию.

Сформированные и закодированные битовые потоки преобразуются в электрические сигналы и передаются по каналу связи с использованием низкого напряжения LVDS (Low Voltage Differential Signalling – дифференцированные сигналы низкого уровня). Так как для передачи дан-

ных с помощью LDVS требуется два провода, то для обеспечения полнодуплексной передачи данных каждый кабель SpaceWire содержит четыре экранированные витые пары.

Таблица 1

Стек протоколов стандарта SpaceWire

Название уровня	Функции
Сетевой	Маршрутизация пакетов в сетевых узлах
Уровень	Формат пакетов и метод инкапсуляции данных в пакеты
Уровень обмена	Установление и управление соединением точка-точка, включая инициализацию канала связи, управления потоком данных, синхронизацию, обнаружение неисправностей и перезапуск соединения
Символьный	Формирование и кодирование символов данных и управления.
Сигнальный	Преобразование данных в сигналы линии связи, спецификация их электрических и временных параметров, границ помехоустойчивости и скорости передачи данных в канале
Физический	Определяет разъемы, кабели, проводники на печатных платах и технические требования по обеспечению электромагнитной совместимости

### Протокол AFDX

AFDX (AvionicsFullDuplexSwitchedEthernet) – авиационная сеть передачи данных базирующаяся на основе 10/100MbitEthernet. AFDX использует специальный протокол для обеспечения безопасной и надежной передачи данных, с гарантированным временем их передачи [3,4].

Основной целью стандарта AFDX (табл.2) является создание детерминистской сети передачи данных. Эта цель достигается при помощи предоставления выделенных полос пропускания трафика для каждого маршрута информации в сети и обеспечения доступности спецификации качества обслуживания (QoS) на каждом узле системы.

Сеть AFDX состоит из самолетных компьютеров и коммутаторов. В этой сети самолетные компьютеры являются потребителями сервисов передачи в сети. Уровень конечных систем (КС) на содержащем его компьютере предоставляет этот сервис. Конечные системы позволяют приложениям, исполняющимся на компьютере, посылать и получать данные достоверным и безопасным способом.

Конечные системы коммутируются на основе виртуальных каналов с приоритизацией трафика (TrafficShaping), с помощью назначения пропускной способности каждого виртуального канала. Непревышение пропускной способности виртуальных каналов достигается благодаря согласованным действиям отправителей и коммутаторов. Цель функции формирования трафика заключается в ограничении мгновенной скорости кадров в виртуальных каналах путем введения интервалов между кадрами.

Конечные Системы связываются друг с другом при помощи коммутирующих пакеты компьютеров, называемых «коммутаторами AFDX». Основной задачей коммутаторов является обмен кадрами между конечными системами, подключенными к их портам. В дополнение к этому коммутаторы реализуют правила контроля трафика на заранее сконфигурированных виртуальных каналах, статически определенных пользователем. Виртуальный канал (ВК) это связь, используемая конечной системой для передачи данных через сеть. Виртуальный канал представляет собой однонаправленный «один ко многим» коммуникационный маршрут.

Гарантированное обслуживание обеспечивает четкую, математически обоснованную задержку, ограниченную сквозной передачей кадра. В результате этого, для того, чтобы гарантировать ограниченную задержку, необходимо гарантировать определенную пропускную способность на уровне канала передачи.

Таблица 2

Стек протоколов стандарта AFDX

Название	Функции
Приложения	Обеспечение взаимодействия пользователей с сетью

Представления	Кодирование/декодирование входящих и исходящих данных. Данные приложений прикладного уровня, форматируются для дальнейшей передачи по сети. Данные, полученные из сети форматируются для дальнейшей передачи приложениям
Сеансовый	Установка и разрыв соединений, управление обменом информацией и синхронизация.
Транспортный	Контроль процесса передачи данных (например, проверяет все ли пакеты доставлены) и слежение за ошибками передачи
Сетевой	Маршрутизация данных
Канальный	Обеспечение синхронизации с физическим уровнем. Данные, полученные с физического уровня упаковываются в пакеты, проверяется их целостность
Физический	Передача битового потока через сеть. Определяется метод передачи данных, физические свойства проводников и др.

### Протокол FlexRay

FlexRay – высокоскоростной сетевой протокол, обеспечивающий передачу данных в режиме реального времени, получивший широкое признание в отрасли автомобилестроения [2].

В протоколе FlexRay, управление доступом к среде передачи, основано на повторяющемся коммуникационном цикле. Вся коммуникация организована на статическом и динамическом принципе передачи. FlexRay осуществляет связь с использованием двух физически разделенных линий со скоростью передачи данных 10 Мбит / с каждый. Эти две линии в основном используются для обеспечения избыточности, а следовательно отказоустойчивости передачи сообщений. Протокол FlexRay использует механизм тактовой синхронизации, каждый узел по отдельности синхронизируется с кластером, по кадрам передаваемым от других узлов. Глобальная синхронизация обеспечивает, коррекцию скорости и смещения передачи данных. Также FlexRay включает в себя независимые шины guardian, обеспечивающие дополнительную поддержку при локализации ошибок. Bus guardian необходимы для выявления неисправных узлов и системных прерываний, сохранения гарантированного времени передачи и коррекции ошибок.

Таблица 3

Стек протоколов стандарта FlexRay

Название	Функции
Приложений	Обеспечение взаимодействия пользователей с сетью
Канальный	Обеспечение синхронизации с физическим уровнем, управление логическим каналом (LLC), управление доступом к среде (MAC)
Физический	Передаёт битовый поток через сеть. Определяет метод передачи данных, физические свойства проводников и др

### Сравнение протоколов

Таблица 4

Сравнение протоколов

Характеристика	SpaceWire	FlexRay	AFDX
Описание	SpaceWire – сеть для космических аппаратов, основанная на части стандарта соединения IEEE 1355	FlexRay – высокоскоростной сетевой протокол для автомобилей	Авиационная сеть передачи данных базирующаяся на основе 10/100MbitEthernet
Применение	Высокоскоростные линии и сети для использования на борту космических аппаратов	Автомобильные приложения управления	Коммерческие самолеты
Конкретные системы	Космические аппараты: розетка, гея	Автомобили: BMW X5, BMW 7 Series	Самолеты: Airbus A380, A400M, A350 and Boeing 787

Характеристика	SpaceWire	FlexRay	AFDX
Коммуникация	По событиям	По времени (статические сегменты) и событиям (динамические сегменты)	По событиям, гарантированная пропускная способность
Скорость передачи данных Мбит/с	400 Мбит/с	10 Мбит/с (20 Мбит/с без резервирования)	100 Мбит/с
Размер пакета	5 байт служебной информации, полезная нагрузка стандартом не ограничена	8-байт служебной информации, 0-254 байт полезных данных	55 байт служебной информации, 17-1471 данных
Помехоустойчивое кодирование	Нет	Да	Да
Тип передачи данных	Дуплексная	Полудуплексная	Дуплексная
Временная синхронизация	Не требуется, но доступна через распределение времени	Да	Нет
Реализация физического уровня	4 экранированных витых пары с LVDS (Low Voltage Differential Signalling)	Витая пара	Ethernet 100 Base-T витая пара
Сетевая топология	Точка точка и через коммутатор	Шина, пассивная звезда(коммутатор), точка-точка, активная звезда(маршрутизатор)	Коммутируемая звезда, и "звезда", один или несколько лучевых узлов которой являются центральными узлами других "звезд"
Избыточность	Нет	Да, для статических частей резервирование, для динамических опционально	Да, обязательной дублирование передачи данных
Локализация ошибок	Нет	Да, на локальной или центральной шине	Да, неисправность обнаруживается через резервированную сеть
Исправление ошибки	Нет	Да, реализуется на шине	Обнаружение и исправление при коммутации
Сообщение об ошибке	Да, проверка четности, неправильная точка назначения, ошибка конца пакета	Да	Да, коммутатор обнаруживает ошибки и генерирует сообщение
Устойчивость к потерям сообщений	Нет	Да, через избыточность	Дублирование сети
Обнаружение отказов узлов	Только при ошибке соединения. Таймаут на уровне приложений	Да.	Нет
Мгновенное получение сообщения об отказе узла	Только в случае отказа линии связи	Нет	Только узлы, ожидающие данные, на уровне приложений
Подход к обработке ошибок	Переподключение и сообщение приложению об ошибке. Отчет приложению о обнаружении физической ошибки	Bus Guardians предотвращает передачу неисправных узлов	Ошибочные пакеты отбрасываются. Если пакет нельзя восстановить маршрутизатор сообщает об отказе

Выводы

Данное исследование предоставляет собой развернутый обзор бортовых телекоммуникационных сетей широкоиспользуемых в настоящее время. Обзор хорошо показывает основные различия сетей.

#### **Библиографический список**

1. ESA (European Space Agency). Standard ECSS-E-ST-50-52C, "Space engineering. SpaceWire – Remote memory access protocol. European cooperation for space standardization". Noordwijk: ESA Publications Division ESTEC, 2010.
2. FlexRay Consortium (2005), FlexRay Communications System Protocol Specification Version 2.1.
3. Aeronautical Radio, Inc. (2005), Aircraft Data Network Part 7 Avionics Full-Duplex Switched Ethernet (AFDX) Network.
4. Condor Engineering, Inc. (2005), ADFX Protocol Tutorial.

**УДК 004.942**

**Е. Н. Мухина**

студентка кафедры аэрокосмических приборов и измерительно - вычислительных комплексов

**В. Г. Никитин**

канд. техн. наук, доц. – научный руководитель

### **ИССЛЕДОВАНИЕ ВЫЧИСЛИТЕЛЯ БОРТОВОГО ИЗМЕРИТЕЛЬНО-ОПТИЧЕСКОГО КОМПЛЕКСА**

В настоящее время в авиационных комплексах широко применяются электронные устройства. Они работают в условиях внутренних и внешних механических сил. Внутренние силы возникают вследствие работы механизмов внутри самих радиоэлектронных средствах (РЭС), чаще всего за счет эксцентриситета вращающихся узлов, вызывая «силовое» механическое возбуждение конструкции РЭС. Воздействие внешних сил передается от объекта-носителя на РЭС через точки крепления, вызывая «кинематическое» возбуждение.

К внешним воздействия относятся:

1. случайные вибрации;
2. однократные и многократные удары;
3. линейное ускорение;
4. акустический шум.

Механические воздействия (МВ) вызывают 30–50% отказов РЭС. Наиболее опасными механическими воздействиями, ухудшающими надежность и стабильность аппаратуры, являются вибрация и удары. Источниками данных воздействий могут быть разного рода двигатели, в том числе и реактивные, быстро вращающиеся разбалансированные массы, дорожная тряска, акустические шумы, ударная волна и многие другие [1]. МВ в РЭС приводят к изменению активного сопротивления в полупроводниках, деформации электрорадиоэлементов (ЭРЭ), помехам в каналах передачи, к механическим разрушениям конструкции. Поэтому на стадии разработки РЭС очень важно исследовать ее реакции и выдерживание МВ, заданных в технических условиях (ТУ) и предотвратить возможность поломки.

В результате расчетов изделие не должно иметь резонанса конструктивных элементов при испытании и должно быть работоспособным, сохранять внешний вид и соответствовать требованиям изложенным в ТУ [2].

Моделирование платы на МВ проводилось в среде АСНИКА-ТМ [3]. Данный пакет предназначен для моделирования печатных узлов (ПУ) на тепловые, механические и комплексные воз-



действия. В подсистеме рассматривались такие механические воздействия случайная вибрация, одиночный удар, удар многократного воздействия и линейное ускорение.

Исследуемая печатная плата, (рис. 1.), была смоделирована в пакете AltiumDesigner [4],[5], в котором позднее сохранена в формате brd для последующей передачи в подсистему АСОНИКА-ТМ.

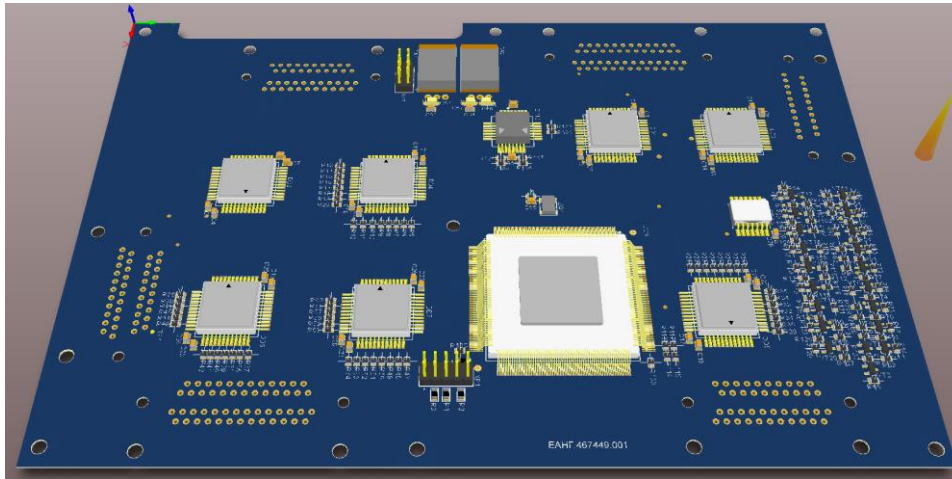


Рис.1. 3D-модель печатной платы

Для моделирования механических воздействий в среде АСОНИК-ТМ необходимо задать: параметры материала печатной платы (рис. 2), параметры электрорадиоизделий ПУ и описать внешнее механическое воздействие.


Название параметра	Значение параметра
Параметры слоя	
 Толщина слоя, [мм]	2.00
Материал слоя	СФ-1-35
Количество вырезов	0
Параметры материала	
Плотность, [кг/м <sup>3</sup> ]	1850.00
Кoeffициент теплопроводности, [Вт/(К*м)]	0.30
Удельная теплоемкость, [Дж/(кг*К)]	420.00
Кoeffициент черноты, [отн. ед.]	0.80
Модуль упругости по оси X, [ГПа]	16.90
Модуль упругости по оси Y, [ГПа]	16.90
Модуль упругости под углом 45°, [ГПа]	16.90
Кoeffициент Пуассона по оси X, [отн. ед.]	0.22
Кoeffициент Пуассона по оси Y, [отн. ед.]	0.22
Кoeffициент Пуассона под углом 45°, [отн. ед.]	0.22
Кoeffициент механических потерь (для вибрации), [отн. ед.]	2.64E-03
Кoeffициент механических потерь (для удара), [отн. ед.]	2.64E-03
Кoeffициент зав. КМП от напряжения (для вибрации), [1/Па]	7.46E-10
Кoeffициент зав. КМП от напряжения (для удара), [1/Па]	7.46E-10
Кoeffициент зав. модуля упругости по оси X от температуры, [ГПа/°C]	0.14
Кoeffициент зав. модуля упругости по оси Y от температуры, [ГПа/°C]	0.14
Кoeffициент зав. модуля упругости под углом 45° от температуры, [ГПа/°C]	0.14
Кoeffициент зав. КМП от температуры (для вибрации), [1/°C]	4.10E-04
Кoeffициент зав. КМП от температуры (для удара), [1/°C]	4.10E-04
Кoeff. зав. от температуры коэф. зав. КМП от напряжения (для вибр.), [1/(Па*°C)]	-9.00E-13
Кoeff. зав. от температуры коэф. зав. КМП от напряжения (для удара), [1/(Па*°C)]	-9.00E-13
Максимальная допустимая температура нагрева, [°C]	85.00
Минимальная допустимая температура охлаждения, [°C]	0.00
Максимальное допустимое напряжение на изгиб, [МПа]	0.00

Рис.2 Параметры материала печатной платы

1. Случайная широкополосная вибрация. Плавное изменение частоты от 10 до 2000 Гц. График спектральной плотности виброускорения показан на рис. 3

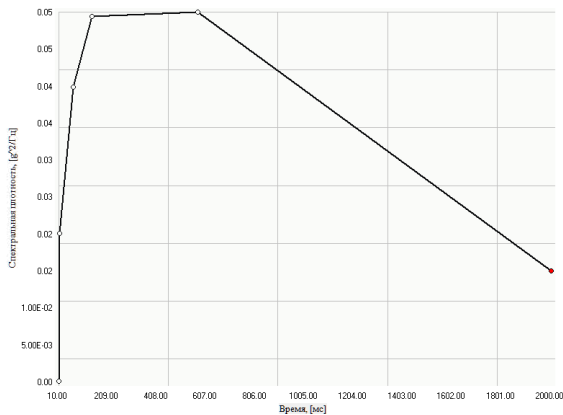


Рис.3. График спектральной плотности виброускорения

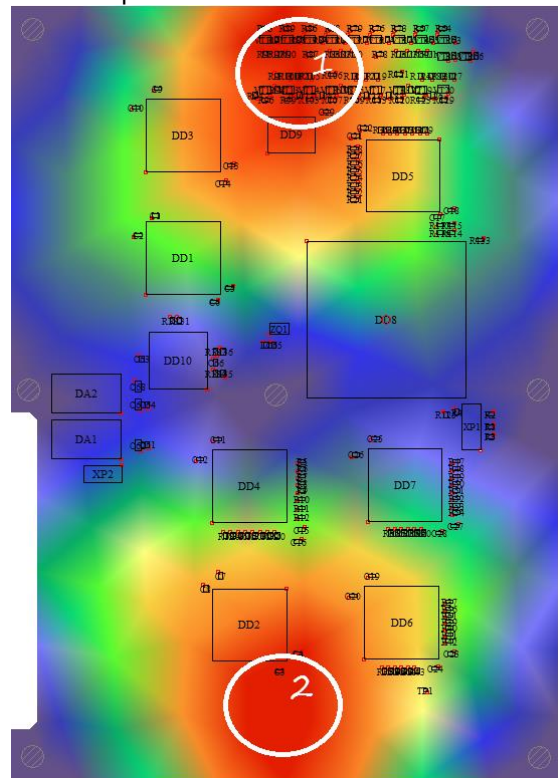


Рис.4. Результаты моделирования

Как видно из рис. 4 максимальная нагрузка при вибрации приходится на точках 1 и 2. Для них была построена на рис. 5 зависимость спектральной плотности от частоты, где видно, что на частотах 450, 680, 1400 Гц многократно превышает значения в ТУ.

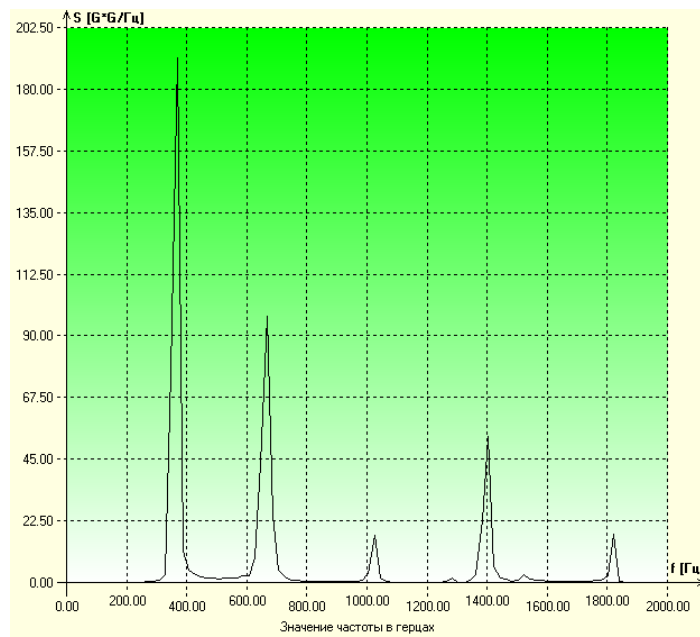


Рис.5. График зависимости спектральной плотности от частоты

2. Одиночный удар. 20 одиночных ударов по оси x с ускорением 15 g и длительностью ударного импульса 10–15 мс. График воздействия показан на рис. 6.

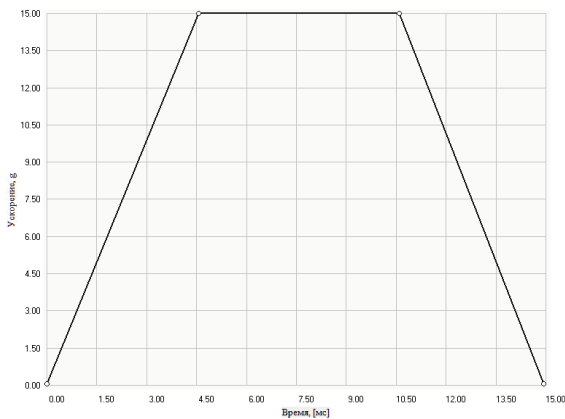


Рис.6. График воздействия одиночного удара

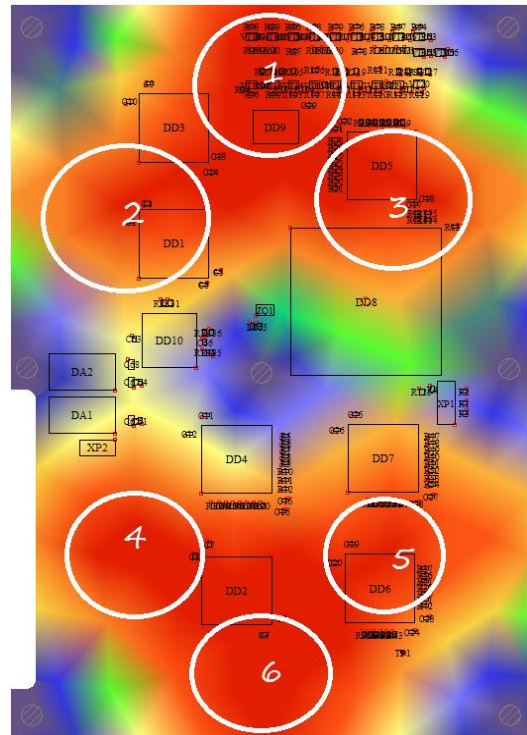


Рис.7. Поля ускорений на печатной плате

Из рис. 7 видно, что максимальная нагрузка при одиночном ударе приходится на участках 1–6. Проведем для них анализ и построим зависимость ускорения от времени (рис. 8), где видно, что на участках 1 и 6 ускорение превышает норму на 13%, что в данном случае удовлетворяет ТУ.

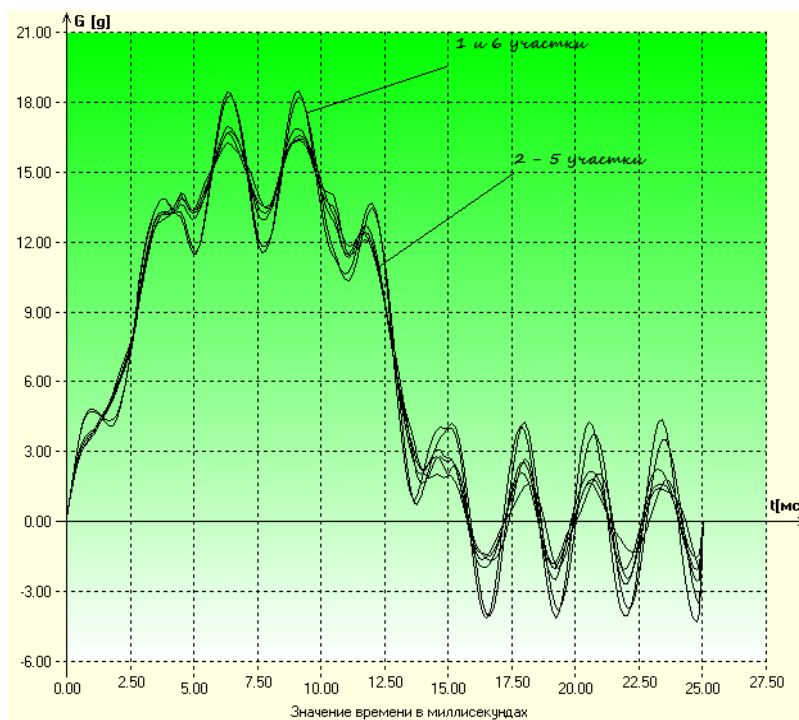


Рис.8 График зависимости ускорения от времени

3. Многократный удар. 3 удара до 6 g и длительность ударных импульсов 10– 80 мс при частоте 40–80 уд. мин. График воздействия отдельно взятого импульса показан на рис. 9, а результаты моделирования – на рис. 10.

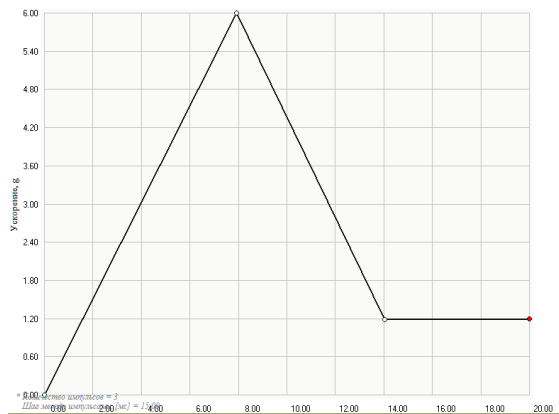


Рис.9. График воздействия отдельно взятого импульса

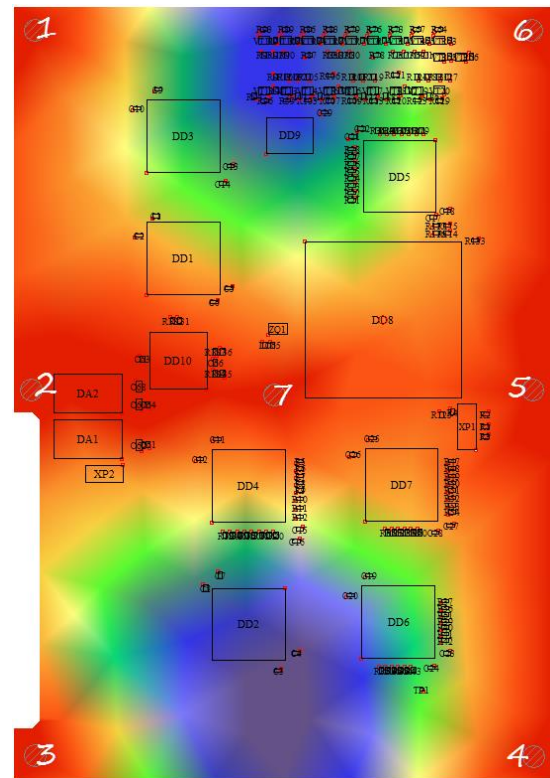


Рис.10. Поля ускорений на печатной плате)

Как видно из рис. 10, максимальная нагрузка при многократных ударах приходится на точки крепления платы 1–7. Для данных точек была построена на рисунке 11 зависимость ускорения от времени, где видно, в точках крепления ускорение превышает норму в 50%, что довольно существенно, так как такое воздействие ведет к деформации платы.

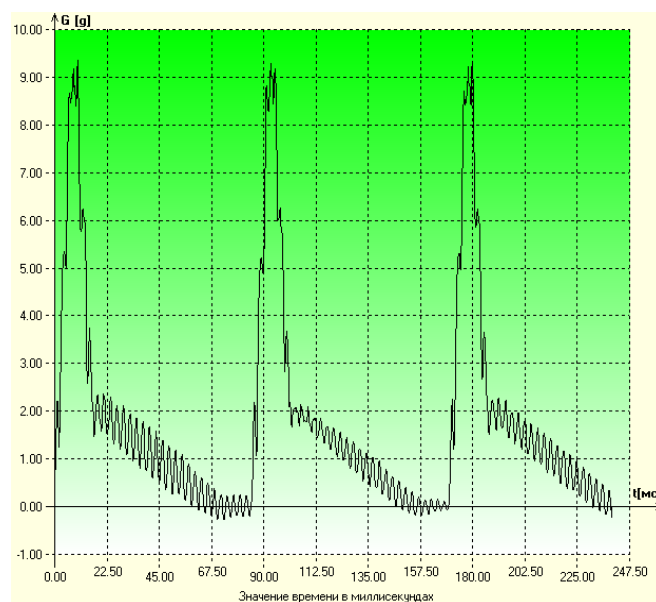


Рис.11. Зависимость ускорения от времени при многократном ударном воздействии

Проведенное исследование платы вычислителя бортового измерительно-оптического комплекса позволяет сделать следующие выводы: механические воздействия, такие как одиночные удары, не критичны для исследуемой платы в отличие от случайной вибрации и многократных ударов, которые в данном случае могут привести к разрушению и потере работоспособности изделия. Для обеспечения устойчивости блока вычислителя к вибрациям необходимо блок установить на амортизаторы, предназначенные для демпфирования диапазона вредных вибраций. Представленный метод моделирования позволяет провести анализ надежности для прогнозирования возможных внешних воздействий, которые могут вывести устройство из работоспособного состояния. Разработанная система создана на базе АО «НПК Пеленгатор».

### Библиографический список

1. Малов А.В. Разработка автоматизированной подсистемы обеспечения стойкости радиоэлектронных средств к механическим воздействиям на основе системы виброизоляции/ Дисс.канд.тех.наук М., 2011.
2. Руководство по эксплуатации бортового измерительно-оптического комплекса.
3. Автоматизированная Система Обеспечения Надежности и Качества Аппаратур [Электронный ресурс] // Официальный интернет портал [сайт]. URL <http://asonika.ru/> (дата обращения 20.05.2016)
4. Суходольский В.Ю. Altium Designer проектирование функциональных узлов РЭС на печатных платах: Учебное пособие. СПб.:БВХ-Петербург, 2010.- 480 с.
5. Altium Designer 6. Руководство пользователя. Altium. 2007. - 248с.

УДК 004.021(075.8)

**С. М. Пахарев**

студент кафедры аэрокосмических компьютерных и программных систем

**А. Ю. Сыщиков**

ст. препод. – научный руководитель

## МЕТОДЫ ФОРМАЛЬНОЙ ВЕРИФИКАЦИИ ПРИЛОЖЕНИЙ

Введение

Верификация приложений является одним из методов оценки разрабатываемого программного обеспечения (ПО). С её помощью производят проверку соответствия разрабатываемой системы заранее определенным стандартам, правилам и сертификатам. Методы верификаций приложений принято разделять на следующие группы:

- методы динамического анализа;
- методы статического анализа;
- экспертный анализ;
- формальные методы;
- синтетические методы.

Группа методов верификации, которые основываются на экспертном анализе ПО, как правило, не могут быть автоматизированы – для верификации таким способом привлекается группа специалистов, которые производят ручную проверку оцениваемых свойств. Этот способ является наименее удобным при верификации промышленных приложений, поскольку требует большого количества специалистов для оценки и не предоставляет гарантированного результата.

Методы динамического анализа (динамическая верификация) используются для оценки уже готового ПО или его компонентов в режиме моделирования. Существуют два основных подхода для динамической верификации: мониторинг и тестирование. Оба подхода требуют неоднократного запуска оцениваемого ПО, что лишает программиста возможности производить оценку своего продукта в ходе разработки.

Методы статического анализа используются для проверки полученного кода программ без его запуска. Этот подход имеет сильное преимущество перед всеми остальными: программисту не требуется создание дополнительных описаний своей системы или преобразование своей программы к какому-то определенному стандарту, поскольку проверяется качество и корректность именно созданного кода программы.

Формальные методы верификации используются для проверки моделей разрабатываемых приложений. Большая часть формальных методов требует представления ПО либо в виде математических выражений, либо в виде графа. Формальные методы верификации можно применять ещё до начала разработки приложения, поскольку они не требуют исходного кода программы, а основываются лишь на взаимосвязях и взаимоотношениях между рассматриваемыми компонентами. В качестве компонентов можно рассматривать как модули программ, так и функции, и даже операторы, что позволяет использовать этот подход на различных уровнях гранулярности.

Синтетические методы сочетают в себе одновременно несколько возможных методов верификации приложений. Такие методы являются наиболее сложными для создания, однако дают наиболее полный результат, поскольку используют сразу несколько подходов для проверки. Так, синтетический метод может сочетать в себе одновременно и динамические методы, которые будут применяться, к примеру, к уже готовой системе для оценки времени работы ПО, и статический анализ при оценке программы для нахождения ошибок при работе с памятью.

Целью выполненной работы было исследование основных формальных методов верификации как наиболее часто используемых для оценки не только готовых приложений, но и их моделей до начала этапа разработки, когда цена ошибки не так велика по сравнению с готовым продуктом.

Основные методы формальной верификации

Метод логического вывода

Метод логического вывода основывается на применении дедуктивных и индуктивных доказательств какого-то свойства. К примеру, для возникновения утечек памяти необходимо выполнение двух условий: в каком-то месте программы должна быть выделена некоторая область памяти, после чего она нигде не освобождается. Таким образом метод логического вывода для анализа утечек памяти сводится к двум простым шагам: нахождения всех мест, в которых выделяется память ( $n$ ) и в которых память освобождается ( $m$ ). В случае, если  $n > m$ , то логично предположить о вероятных утечках памяти. Однако такой подход применим только для линейных программ без ветвлений, что на практике встречается не часто.

Основным математическим подходом для использования метода логического вывода является применение логики Флойда-Хоара [1], которая состоит из системы с набором логических правил, может применяться для оценки не только последовательных, но и параллельных программ. Основной её характеристикой является тройка Хоара. Тройка представляет собой два условия (предусловие и постусловие) и одно утверждение. Принятое обозначение:

$$\{P\}C\{Q\},$$

где  $P$  - предусловие,  $Q$  - постусловие,  $C$  - проверяемое утверждение (команда).

Тройка Хоара интерпретируется следующим образом: если предусловие выполняется, то в случае корректности исполнения проверяемой команды постусловие также будет верным.

Рассмотрим простейший пример реализации тройки Хоара:

$$\{x = 5\}x := x * 2\{x = 10\}$$

В данном случае предусловием является выражение  $x = 5$ , которое является начальным условием для выполнения последующей команды. После выполнения команды  $x := x * 2$  планируется получить результат  $x = 10$ . В случае, если результат выполнения команды не будет совпадать с постусловием, то команда выполнена неверно и именно в этом месте программы возможна ошибка.



Основной идеей логики Флойда-Хоара является задание для каждой конструкции языка предусловия и постусловия, записанных в виде логической формулы. Именно это и является большим недостатком логики Флойда-Хоара: в общем случае эта задача является NP-полной.

Метод проверки моделей

Метод проверки моделей (model checking) является наиболее популярным методом формальной верификации параллельных систем с конечным числом состояний. Для верификации с помощью этого способа необходимо создать модель проверяемой системы, а также набор формальных спецификаций. Для разработки модели системы чаще всего используется модель Крипке, характеризующаяся множеством состояний, отношений переходов и функцией разметки. Спецификация для системы создается на одном из языков формальной логики. В качестве языков формальной логики, как правило, выступает темпоральная логика, с помощью которой возможно описание работы системы во времени.

Основной проблемой, возникающей при использовании метода проверки моделей, является эффект комбинаторного взрыва, возникающий в пространстве состояний [2]. Это не позволяет использовать проверку моделей для систем с большим количеством взаимосвязей между компонентами, а также в программах, использующих входные данные больших размеров. В связи с этой проблемой, а также с тем, что программа должна иметь конечное число состояний, метод становится неприменим при оценке достаточно широкого круга задач.

Теория графов и сети Петри

Ещё одним известным методом формальной верификации является представление системы в виде графа и анализ его формальных свойств. Чаще всего для преобразования программы в графовую структуру применяется модель Сети Петри [3], состоящая из двудольного ориентированного мультиграфа (рисунок). Взаимоотношение между компонентами программы представляются в виде дуг, соединяющих вершины графа, которые бывают двух типов: позиции и переходы. Сеть Петри применяется для анализа динамических дискретных систем и на сегодняшний день существует множество алгоритмов, позволяющих проверять большое количество свойств, среди которых однозначность результатов вычислений и тупиковые ситуации.

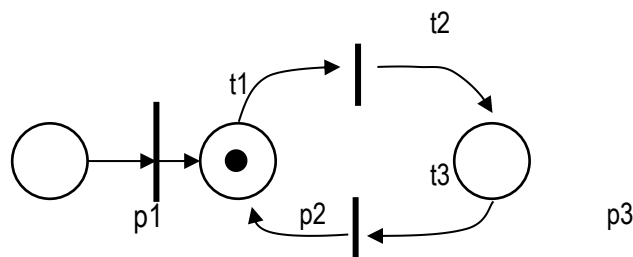


Рис. Пример сети Петри

Переход  $t1$  является тупиковым и никогда не будет выполнен в ходе работы этой сети, в отличие от переходов  $t2$  и  $t3$ , которые являются «живыми». Позиция  $p1$  является недостижимой. Однако цепь  $p2 \rightarrow t2 \rightarrow p3 \rightarrow t3 \rightarrow p2$  является циклом в графе, который будет выполняться бесконечно и программа будет постоянно выполнять одну и ту же работу. Наличие цикла в графе является одним из четырех условий Коффмана для возникновения дедлоков в программе [4].

Заключение

Существует множество методов верификации, но каждый из методов обладает своими ограничениями и недостатками. Некоторые алгоритмы требуют сложных математических вычислений, некоторые работают только с конечным числом состояний программы, а для некоторых методов требуются полностью готовые компоненты. Чем на более позднем этапе была произведена верификация программы и выявлена ошибка, тем сложнее её исправить, и стоимость этого исправления возрастает нелинейно. Формальные методы верификации удобны для использова-

ния на различных стадиях разработки, что позволяет применять их даже на этапе проектирования системы. Кроме того, формальные методы верификации не требуют четких доказательств и сложных теорем, что облегчает создание спецификаций к тестируемым системам. Для проверки достаточно лишь формального описания, к примеру, в виде графовой модели. Методы формальной верификации позволяют находить логические ошибки и ошибки взаимодействия компонентов системы на различных уровнях гранулярности, что также является безусловным преимуществом использования данного подхода при анализе разрабатываемых систем.

### Библиографический список

1. Hoare C. A. R. An axiomatic basis for computer programming //Communications of the ACM. 1969. Т. 12. №. 10. С. 576-580.
2. Карпов Ю. Г. MODEL CHECKING. Верификация параллельных и распределенных программных систем. БХВ-Петербург, 2010.
3. Котов В. Е. Сети Петри. 1984.
4. Shibu K. V. Intro To Embedded Systems 1E.

### УДК 629.735.33

**А. Н. Понедельникова, А. Ю. Крузе, А. М. Колесов**

студенты группы 1334

**А. С. Михальчук**

научный руководитель

### АНАЛИЗ ТОЧНОСТИ РЕЗУЛЬТАТОВ РАСЧЕТА РАЗЛИЧНЫМИ МЕТОДАМИ ЗАДАЧ О ПРОГИБЕ ПЛАСТИНКИ

В статье приведён фактический расчет прогиба пластинок в двух видах заземления: полным и двусторонним .

Ключевые слова: пластина; прогиб; точность; аналитический метод; численный метод.

С точки зрения ценности основной показатель расчетов – точность. Большинство расчетов можно получать различными методами. Эти методы имеют различные теоретические основания. Они также различаются возможностями своих приложений.

Первая группа – аналитические методы. Они дают результат с любой требуемой точностью, но решение с их помощью может быть получено только для ограниченного круга задач.

Вторая группа – численные методы. Они обладают большей универсальностью, но всегда требуют исследования точности получаемых результатов.

Исследование точности результатов – необходимая и повседневная практика инженерной работы. Рассмотрим процесс исследования точности на примере решения простейшей задачи о прогибе пластинки.

Таким образом, произвели расчет прогиба пластинок, свободно опертых на боковых кромках и жестко заделанных, тремя разными методами, и сравним итоговые результаты расчета, сделав вывод о погрешностях результатов.

Дана пластина, свободно опертая на кромках  $x=0$  и  $x=a$  и жестко заделанная на кромках  $y = \pm(b/2)$ , действие на пластину, равномерно распределяется по всей ее площади. Расчётная схема представлена на рис.1, где  $a=2$ ;  $b=1$

Расчет прогиба пластинки первым методом: методом Фурье

Метод Фурье (метод разделения переменных) – метод решения дифференциальных уравнений в частных производных, основанный на представлении искомых функций в виде рядов [2].



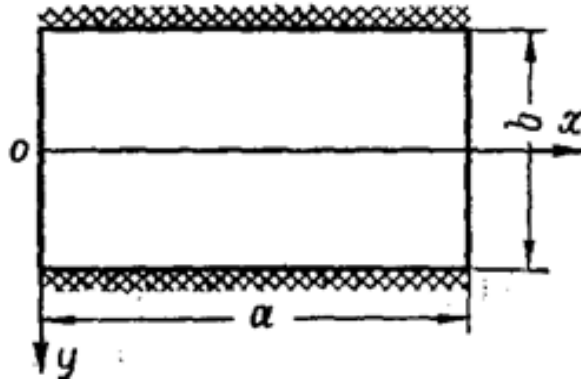


Рис. 1. Схема задачи

Размеры пластин в направлении осей  $Ox$  и  $Oy$  обозначим буквами  $a$  и  $b$  соответственно, а толщину пластины – буквой  $h$ .

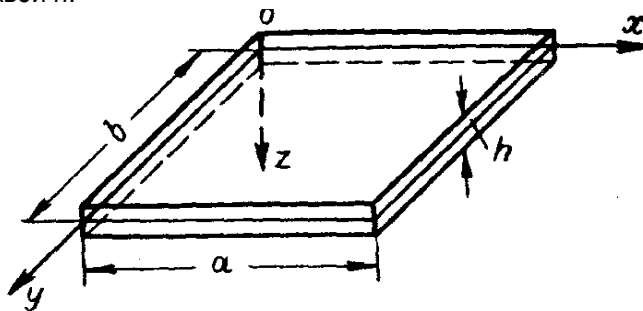


Рис. 2. Схема задачи

Дифференциальное уравнение изгиба абсолютно жестких пластин.

$$\nabla^2 \nabla^2 w = \frac{\partial^4 w}{\partial x^4} + 2 \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} = \frac{q(x,y)}{D} \quad (1)$$

Уравнение (1) представляет собой дифференциальное уравнение в частных производных с постоянными коэффициентами.

Здесь  $D = \frac{Eh^3}{12(1-\mu)^2}$  – цилиндрическая жесткость пластин;  $q$  – нормальное давление на

пластинку;  $E$  – модуль Юнга;  $\mu$  – коэффициент Пуассона;  $h$  – толщина пластинки.

Пусть кромки  $x = 0$  и  $x = a$  свободно оперты. Будем искать решение в виде формулы (2).

$$w = \sum_m f_m(y) \sin \frac{m\pi x}{a} \quad (2)$$

Дифференциальное уравнение, определяющее функции  $f_m(y)$ :

$$f_m^{IV}(y) - 2 \left( \frac{m\pi}{a} \right)^2 f_m''(y) + \left( \frac{m\pi}{a} \right)^4 f_m(y) = 0 \quad (3)$$

Обыкновенное линейное дифференциальное уравнение четвертого порядка с постоянными коэффициентами. Общий интеграл дифференциального уравнения функции  $f_m(y)$ .

$$f_m(y) = A_m ch \frac{m\pi}{a} y + B_m sh \frac{m\pi}{a} y + C_m \frac{m\pi}{a} y ch \frac{m\pi}{a} y + D_m \frac{m\pi}{a} y sh \frac{m\pi}{a} y + F_m(y), \quad (4)$$

где  $f_m(y)$  - частное решение дифференциального уравнения (1).

Входящие в выражение постоянные интегрирования должны быть определены из условий закрепления опорных кромок пластины  $y=0$  и  $y=b$ ,  $f_m(0) = 0$ ,  $f_m(b) = 0$ .

Выражение для функции  $f_m(y)$ :

$$f_m(y) = A_m \operatorname{ch} \frac{m\pi}{a} y + D_m \frac{m\pi}{a} y \operatorname{sh} \frac{m\pi}{a} y + \frac{4qa^4}{D(m\pi)^5} \quad (5)$$

Входящие в это выражение постоянные интегрирования  $A_m$  и  $D_m$ , должны быть определены из условий для функций  $f_m(y)$  при  $y = \pm \frac{b}{2}$ .

Граничные условия для функций  $f_m(y)$ :

$$\left. \begin{aligned} f_m\left(\pm \frac{b}{2}\right) &= 0 \\ f_m'\left(\pm \frac{b}{2}\right) &= 0 \end{aligned} \right\} \quad (6)$$

Выражение для прогиба пластины, свободно опертой на кромках  $x=0$  и  $x=a$  и жестко заделанной на кромках  $y = \pm \frac{b}{2}$ : (7)

$$w = \frac{4qa^4}{D\pi^5} \sum_m^{1,3,5} \frac{1}{m^5} \left[ 1 + 2 \frac{\operatorname{sh} \frac{u_m}{2} \operatorname{sh} \frac{m\pi y}{a}}{\operatorname{sh} u_m + u_m} \frac{m\pi y}{a} - \left( 1 + \frac{u_m \operatorname{sh}^2 \frac{u_m}{2}}{\operatorname{sh} u_m + u_m} \right) \frac{\operatorname{ch} \frac{m\pi y}{a}}{\operatorname{ch} \frac{u_m}{2}} \right] \sin \frac{m\pi x}{a} \quad (7)$$

Решив это выражение с помощью MatCAD, был получен прогиб пластинки. При реальных расчетах невозможно суммировать бесконечные ряды. Поэтому ограничиваемся суммами 9 и 14 членов. Определяем разницу между этими суммами прогиба в средней точке и, эта разница будет характеризовать точность проведенного расчета. Эти результаты приведены в таблице 1.

Таблица 1

Результаты расчета по методу Фурье

Прогиб в средней точке при суммировании 9 членов	Прогиб в средней точке при суммировании 14 членов	Ошибка	Относительная ошибка
0.00033780005507661	0.00033780014445462	$8.937801 \cdot 10^{-11}$	$2.646 \cdot 10^{-7}$

Графические результаты расчета представлены на рис. 3.

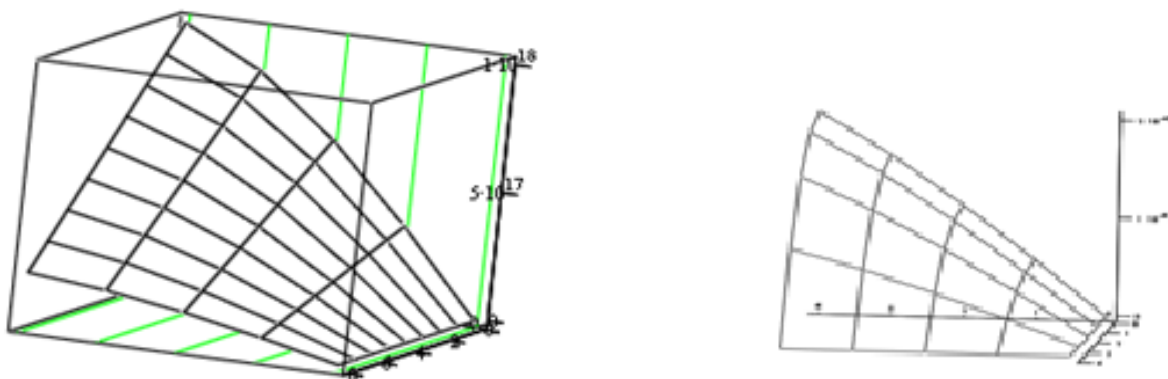


Рис. 3. Внешний вид графика функции  $W_2$  и  $W_4$  четверти пластинки

Второй метод, используемый в исследовании – метод конечных разностей  
 Источник метода конечных разностей – формула Тейлора

$$w(x_{i+1}, Y) = w(x_i + \Delta x, Y) = w(x_i, Y) + \Delta x \frac{\partial}{\partial x} w(x_i, Y) + \left(\frac{\Delta^2 x}{2}\right) \frac{\partial^2}{\partial x^2} w(x_i + \theta, Y) \quad (8)$$

Этот метод численного решения краевых задач для дифференциальных уравнений называют также методом сеток. Суть метода состоит в следующем: на рассчитываемую область наносится сетка с узлами. Все производные, входящие в дифференциальные уравнения и граничные условия, приближенно заменяются соответствующими разностными отношениями (по формулам численного дифференцирования) и таким образом выражаются через неизвестные узловые значения искомой функции. В результате приходим к системе линейных алгебраических уравнений относительно значений функций в узлах сетки. Решение этой системы с последующей интерполяцией в промежутках между узлами позволяет в конечном счете получить приближенное решение рассматриваемой задачи. Исходное уравнение пластинки в конечно – разностном представлении:

$$\nabla^2 \nabla^2 w = \frac{\partial^4 w}{\partial x^4} + 2 \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} = \frac{q}{D}, \quad (9)$$

где  $w$  – прогиб произвольной точки срединной поверхности пластины,  $q$  – распределенная нагрузка, перпендикулярная к средней точке,  $D$  – цилиндрическая жесткость.

Пусть исследуемая сетка будет квадратной с шагами по  $x$  и  $y$ , равными  $\Delta$ . В каждом узле этой сетки выполняется исходное уравнение. Для равновесия пластинки в узле  $K$  используется конечно – разностная схема (рис.4).

Конечно – разностная схема для задачи по пластинке в точке  $K$ . Преобразование уравнения отображено в формуле 10.

$$(\nabla^2 \nabla^2 w)_k = \frac{q_k}{D}, \quad k = 1, 2, \dots, N \quad (10)$$

где  $N$  – число узлов сетки;  $q_k$  – средняя нагрузка на площадке  $\Delta x \Delta y$ , окружающей узел  $k$ .

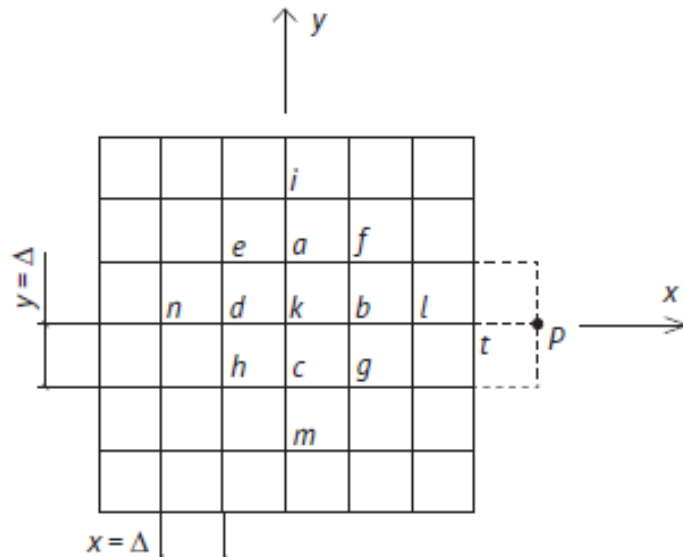


Рис. 4. Разностная схема уравнения пластинки

Если в узле приложена сосредоточенная сила  $F_k$ , то  $q = \frac{F_k}{\Delta k^2}$  в развернутом виде при равномерно распределенной  $q$  конечно – разностный аналог имеет вид (11)

$$20w_k - 8(w_a + w_b + w_c + w_d) + 2(w_e + w_f + w_g + w_h) + w_i + w_l + w_m + w_n = K1 \frac{q_k \Delta^4}{D} \quad (11),$$

где  $K_1$  – весовой коэффициент, зависящий от площади загрузки  $k$ -го узла.

Разностная схема уравнения пластинки представлена на (рис.5). Цифрами показаны коэффициенты перед значением искомой функции в соответствующей точке.

Таким образом, зная размеры пластинки, можно нарисовать сетку.

Используя выражение (10), было составлено 8 выражений для восьми узлов. Используя эти узлы, составляется квадратная матрица  $8 \times 8$ . С помощью команды solve строятся и вычисляются конечные значения прогиба пластинки.

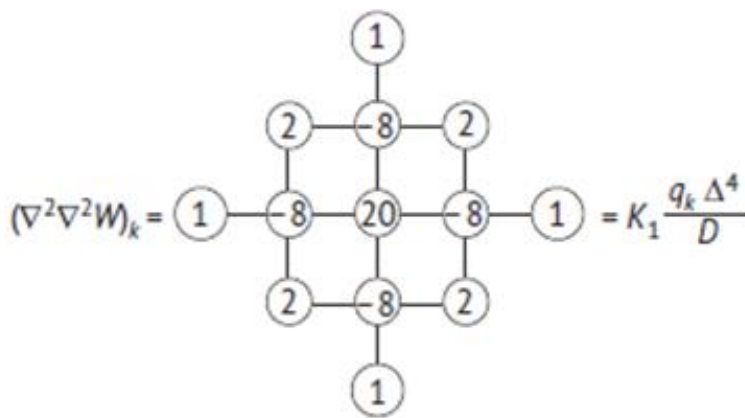


Рис. 5. Конечно-разностная схема задачи

Шаг сетки принимается равным 0,25 м. Схема сетки показана на рис. 6, матрица полученной системы алгебраических уравнений представлена на рис. 7.

					-1	-2	-3	-4	-5	
V	IV	III	II	I	I	II	III	IV	V	3
	4	3	2	1	1	2	3	4	II	2
	5	6	7		8	7	6	5	I	1
	4	3	2	1	1	2	3	4	II	2
	5	6	7			7	6	5		3

Рис. 6. Схема сетки

$$M := \begin{pmatrix} 20 & -16 & 2 & 0 & 0 & 0 & 4 & -8 \\ -8 & 21 & -8 & 1 & 0 & 2 & -8 & 2 \\ 1 & -8 & 20 & -8 & 2 & -8 & 2 & 0 \\ 0 & 2 & -8 & 20 & -8 & 2 & 0 & 0 \\ 1 & 0 & 4 & -16 & 20 & -8 & 1 & 0 \\ 0 & 4 & -16 & 4 & -8 & 20 & -8 & 1 \\ 4 & -16 & 4 & 0 & 1 & -8 & 21 & -8 \\ -16 & 8 & 0 & 0 & 0 & 2 & -16 & 20 \end{pmatrix}$$

Рис. 7. Матрица системы алгебраических уравнений

Результаты расчета прогиба методом конечных разностей:

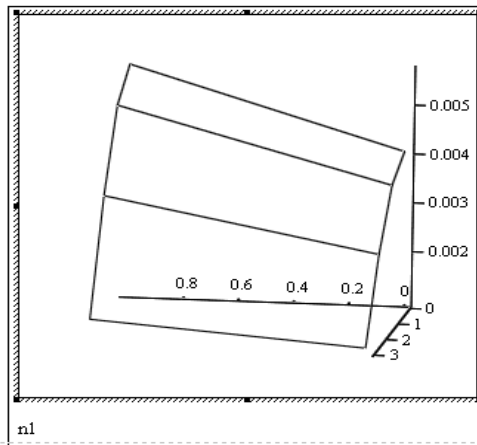


Рис. 8. Результат расчета

Таблица 2

Результаты расчета по конечно-разностному методу

Прогиб в средней точке	Ошибка	Относительная ошибка
$5,67 \cdot 10^{-3}$	$5,332 \cdot 10^{-3}$	0,94

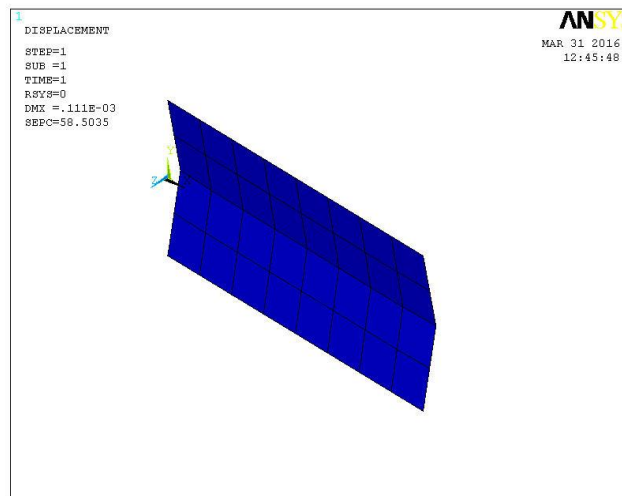
Результат расчета оказался очень неточным. К сожалению, его уточнение без наличия специального программного обеспечения потребует много усилий. Хотя ясно, что это возможно путем увеличения густоты сетки.

Третий метод – Использование программы Ansys метода конечных элементов

Метод конечных элементов – способ решения задач по поиску поля величин путем кусочной замены искомой функции на аппроксимирующую функцию в локальной системе координат. Все это сводится к решению систем алгебраических уравнений [1].

В данном случае для решения задачи в Ansys необходимо задать геометрическую модель пластины, ввести описание свойств материалов, задать внешнее усилие в виде равномерного давления и создать сетку.

Так как программа имеет высокую погрешность при большом размере элемента, то проведено исследование для разных размеров элемента сетки. На рисунках представлены результаты исследований. Погрешность расчета определяется при помощи аппаратных средств пакета Ansys.



с. 9. Размер элемента 0,25 м, погрешность 58 % и матрица 8x8

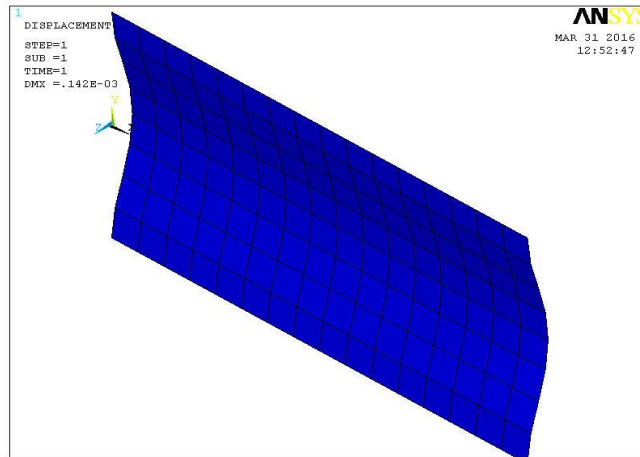


Рис. 10. Размер элемента 0,125 м, погрешность 38 %

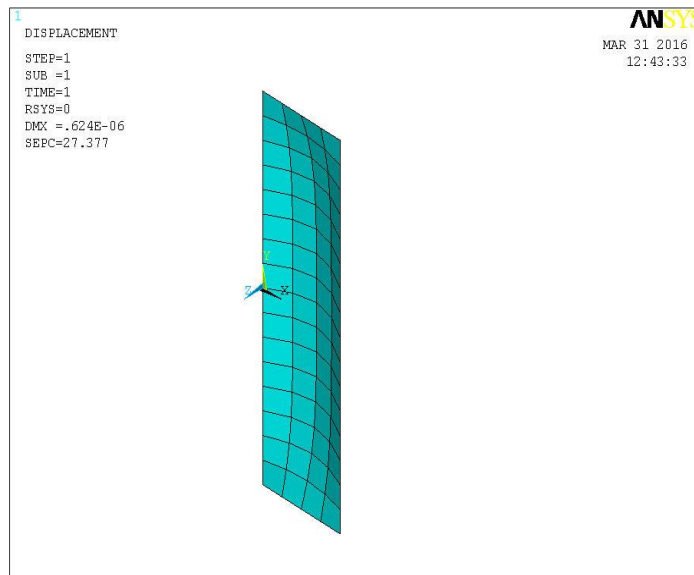


Рис. 11. Размер элемента 0,1 м, погрешность 27 %, матрица 16x16

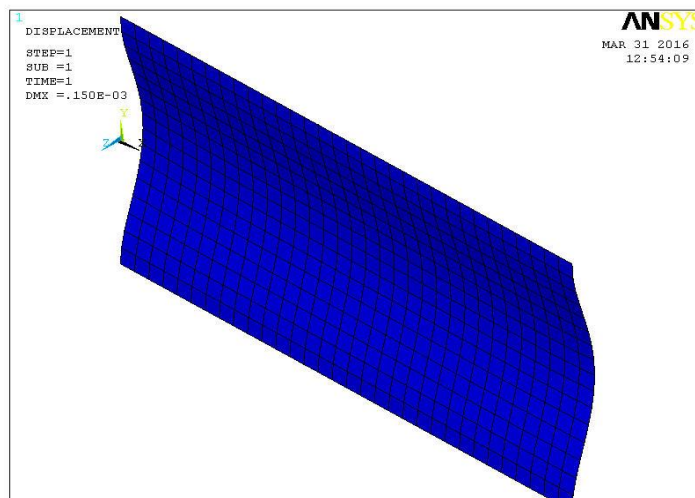


Рис. 12. Размер элемента 0,0625 м, погрешность 21,7 %

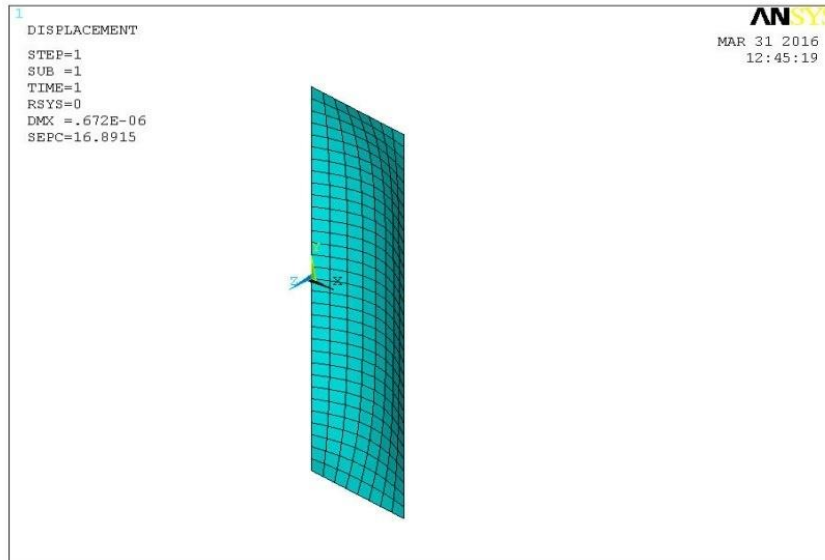


Рис. 13. Размер элемента 0,02567 м, погрешность 17 %

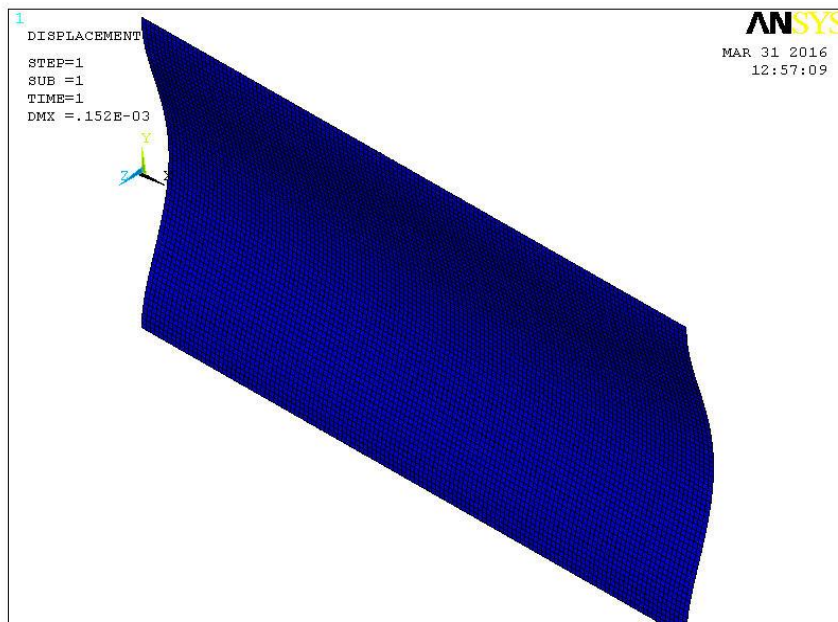


Рис. 14. Размер элемента 0,015625 м, погрешность 6 %

Таблица 3

Зависимость погрешности от размера элемента

Размер сетки, м	Погрешность, %
0,25	58
0,125	38
0,1	27
0,0625	21.7
0,02567	17
0,015625	6



Рис. 15. График зависимости погрешности от размера элемента

В зависимости от размера элемента изменяется ошибка расчета. Чем меньше размер сетки, тем меньше ошибка расчета.

В заключение нужно отметить, что точность метода Фурье оценивается с меньшим количеством членов в этой сумме. Считали когда 9 и когда 14. Получили, что разность невелика. В конечно-разностном методе сложно проводить уточнение, поэтому проведен расчет 1 раз и единожды просчитан прогиб.

Использование пакета Ansys значительно упрощает работу с данными. В нем путем увеличения количества элементов всегда можно достичь требуемой точности. Чем гуще сетка, тем точнее расчет.

Методика инженерных расчетов всегда должна предполагать использование различных методов расчета для анализа точности получаемых результатов.

### Библиографический список

1. Программа для моделирования Ansys [Электронный ресурс]. – Режим доступа <http://www.ansys.com/>
2. Данилина Н.И., Дубровская Н.С., Кваша О.П., Смирнов Г.Л. Вычислительная математика. М.: Высшая школа, 1985. 472 с.

УДК 004.75

**А. А. Рыбальченко**

студент кафедры аэрокосмических компьютерных и программных систем

**С. В. Горбачев**

доц., канд. техн. наук – научный руководитель

## АНАЛИЗ РАСПРЕДЕЛЕННОЙ СИСТЕМЫ ПРЕРЫВАНИЙ НА ОСНОВЕ СТАНДАРТА SPACEWIRE ДЛЯ БОРТОВОГО ВЫЧИСЛИТЕЛЬНОГО КОМПЛЕКСА

Введение

Рассматривается механизм распределенной системы прерываний в стандарте SpaceWire и приводится обоснование необходимости создания программного продукта, осуществляющего расчет времени распространения сигналов распределенного прерывания по сети SpaceWire бортового вычислительного комплекса.



### Стандарт SpaceWire

Стандарт SpaceWire в настоящее время фактически становится базовой технологией при построении высокоскоростных коммуникаций для комплекса бортового оборудования космических аппаратов. SpaceWire представляет собой стек протоколов (вплоть до сетевого уровня), обеспечивающий простой и гибкий набор коммуникационных средств, прозрачных для организации транспортных взаимодействий между терминальными узлами (хост-системами). В дополнение к стеку протоколов SpaceWire ведутся работы по стандартизации следующей версии протокола, обеспечивающего возможность построения систем с управлением процессами в реальном масштабе времени за счет реализации децентрализованного механизма передачи управляющих сигналов распределенных прерываний наряду с коммутацией пакетов данных.

SpaceWire – это наиболее передовая и активно развиваемая сегодня технология для построения распределенных гетерогенных бортовых систем, прежде всего аэрокосмического назначения. Ее поддерживают и внедряют в перспективные космические аппараты Европейское космическое агентство (ESA), космические агентства США (NASA) и Японии (JAXA). В российской космической отрасли развивать и внедрять технологию SpaceWire стратегически важно, поскольку она способствует обеспечению современных тактико-технических характеристик перспективных российских космических аппаратов, конкурентоспособности на мировом рынке космических технологий и услуг, развитию международного сотрудничества в космической области с другими странами, совместимости и унификации разработок аппаратуры космических аппаратов российских предприятий. Технология SpaceWire перспективна и в других областях, где используются встроенные распределенные и параллельные информационно-вычислительные и управляющие комплексы, работающие в реальном времени.

Стандарт SpaceWire регламентирует логические протоколы, физические разъемы и кабели, электрические свойства соединений, которые определяют канал связи SpaceWire, архитектуру коммуникационной сети и обеспечивают средства передачи пакетов информации от исходного узла до требуемого узла назначения через масштабируемую коммуникационную сеть.

Информация по каналам в сети SpaceWire передается в полнодуплексном режиме (в обе стороны канала одновременно) со скоростью 2–400 Мбит/с. Сеть SpaceWire состоит, в общем случае, из некоторого числа узлов-абонентов (SpaceWire nodes) и сетевых узлов – маршрутизирующих коммутаторов (routing switches).

Узлы-абоненты сети SpaceWire – это устройства, передающие и принимающие потоки данных. Они связаны с маршрутизирующим коммутатором или друг с другом дуплексными каналами, называемыми линками (link). Узел оснащен одним или несколькими линк-портами и интерфейсом с источником данных (хост-устройство (host) – процессорный модуль, датчик, исполнительное устройство, периферийный контроллер, и др.). От хост-устройства узел принимает данные, кодирует их и отправляет в свой передатчик, непосредственно подключенный к линку. На другом конце линка данные принимает приемник, который их восстанавливает (декодирует) и передает адресату (другому хост-устройству) или на выходной порт маршрутизирующего коммутатора [1].

### Система распределенных прерываний в сети SpaceWire

Система распределенных прерываний с минимальными задержками по модулям параллельной вычислительной сети является важной системной функцией, особенно для параллельных систем и распределенных комплексов обработки информации.

Механизм распределенных прерываний предназначен для надежной передачи системных сигналов в сети SpaceWire [2].

Прерывание (interrupt) – сигнал, сообщающий о наступлении какого-либо события. При этом выполнение текущей последовательности команд приостанавливается, и управление передается обработчику прерывания, который реагирует на событие и обслуживает его, после чего возвращает управление в прерванный код.

Код прерывания формируется абонентом сети SpaceWire – источником некоторого системного сигнала, который должен быть доставлен с малой задержкой до остальных абонентов сети. Специальные процедуры протоколов канального и сетевого уровня обеспечивают приоритетное распространение этого сигнала по сети. Когда код прерывания доходит до получателя, тот выдает код подтверждения, который должен вернуться к абоненту – источнику прерывания.

Распределение прерываний обеспечивает механизм сигнализации о событиях в сети SpaceWire с низкой задержкой между узлами. Он независим от задержки передачи пакетов и может функционировать даже через линки, в которых имеет место блокировка передачи пакетов данных.

Востребованность инструмента анализа распределенных прерываний в сети SpaceWire

Сигналы прерываний в сети SpaceWire передаются так же быстро, как и метки времени. Известно, что приоритет сигналов прерываний наивысший по отношению к пакетам данных в сети. Для системы передачи меток времени определены фиксированные задержки, которые справедливы и для сигналов распределенных прерываний.

Местонахождение получателя, т.е. обработчика сигнала прерывания, отправителю неизвестно, поэтому сигнал распределенного прерывания распространяется по всем узлам сети. Из-за этого возникают проблемы закливания (loop), которые решаются с помощью установки флагов и настройки тайм-аутов в узлах и маршрутизаторах.

Задержки распространения сигналов распределенного прерывания (мкс) складываются из задержки в обработке маршрутизатором, числа маршрутизаторов и задержки передачи по линку [3].

Исходя из того, что механизм распределенных прерываний в сети SpaceWire используется для реализации достаточно важных системных функций, ставится вопрос о создании инструмента, позволяющего проводить анализ задержек передачи для любой заданной сети на основе протокола SpaceWire с любым количеством и типом узлов, а также протяженностью линков.

Предполагаемый инструмент должен предоставлять достоверную информацию о задержках доставки сигналов распределенных прерываний с учетом размеров сети.

Инструмент анализа может быть полезен системному администратору при проектировании сети SpaceWire для конкретной задачи, так как при настройке сети необходимо конфигурировать счетчики ожидания исходя из размеров сети и количества узлов. Разработчик сети, используя инструмент анализа распределенных прерываний при разработке сети, имеет возможность построить сеть, которая будет непротиворечива с точки зрения распределения кодов прерывания по сети SpaceWire. Помимо этого, правильно настроенные тайм-ауты позволяют решить проблему закливания кодов в сети.

Вывод

Проведен анализ системы распределенных прерываний сети SpaceWire. Очевидно, что разработчику и администратору сети SpaceWire необходимо точно знать, через какой промежуток времени сигнал распределенного прерывания достигнет определенного узла в бортовой сети. Исходя из этого становится очевидной необходимость разработки инструмента анализа распределенных прерываний, использующего в качестве входных параметров размер сети, протяженность линков и количество узлов-абонентов.

### Библиографический список

1. Шейнин Ю.Е. Технология SpaceWire для параллельных систем и бортовых распределенных комплексов.// Электроника: наука, технология, бизнес. 2000. №5. С. 64–74.
2. Koblyakova L. The Analytical Model of Distributed Interrupt Mechanism in SpaceWire Network.// Proceeding Of The 15th Conference Of FRUCT Association: сб. докл. / ITMO university publisher house. СПб., 2014. С. 189-196.
3. Standard ECSS-E-ST-50-52C, "Space engineering. SpaceWire – Remote memory access protocol. European cooperation for space standardization".// ESA Publications Division ESTEC. Noordwijk. 2010.

УДК 004.75

**Д. А. Рябов**

студент кафедры аэрокосмических компьютерных и программных систем

**С. В. Горбачев**

доц., канд. техн. наук – научный руководитель

## АВТОМАТИЧЕСКАЯ КОНФИГУРАЦИЯ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ SPACEWIRE PLUG-AND-PLAY

SpaceWire – это технология для создания коммуникационных сетей (КС) на борту летательных и космических аппаратов. Одной из основных целей стандарта является обеспечение совместимости с различными видами оборудования и многофункциональное использование конечных элементов и подсистем. SpaceWire поддерживает интегрирование и тестирование сложных аэрокосмических систем путём внедрения оборудования непосредственно в аппаратуру обработки данных. Мониторинг и тестирование могут быть проведены без необходимости создания физического интерфейса с аппаратурой обработки данных.

Назначения этого стандарта:

- облегчить проектирование бортовых систем обработки данных;
- сократить затраты на внедрение бортовых систем;
- повысить совместимость подсистем и оборудования обработки данных;
- поддержать многофункциональное использование вычислительного оборудования для решения различных задач;

Plug-and-Play – технология, предназначенная для быстрого определения и конфигурирования устройств.

Цель SpaceWire Plug-and-Play. Основной целью SpaceWire Plug-and-Play является предоставление возможностей для определения или идентификации известных и неизвестных устройств в сети. Протокол также позволяет, с помощью набора функций, поддерживать управление основными элементами сети SpaceWire, таких как определение ошибок на каналах и управление процессом присвоения адресов. Для демонстрации управляющих параметров SpaceWire сервисов и приложений, протокол использует постоянный и расширяемый механизм. SpaceWire Plug-and-Play протокол может быть использован для определения и управления любым протоколом или сервисом, расположенным на сети SpaceWire.

Технология Plug'nPlay (PnP) позволяет:

- Автоматически определять конфигурацию сети после начала работы;
- Отслеживать подключения и отключения отдельных элементов системы (коммутаторов и терминальных узлов), и вносить соответствующие изменения в таблицу маршрутизации;
- Сообщать другим элементам сети об изменении конфигурации

Назначение алгоритма PnP:

- Определять соединение порта коммутатора с другим коммутатором или узлом;
- Определять логические адреса всех узлов, подключенных к коммутатору;
- Конфигурировать и обновлять таблицу маршрутизации при подключении / отключении узлов;
- Обеспечивать другие узлы информацией о сети;
- Реализовать резервирование и устойчивость к отказам в системе;
- Избегать ситуаций блокирования RMAP.

На основе Plug-And-Play протокола, конечные источники являются приложениями, т.е. пользователями SpaceWire. Для пересылки сообщений по сети SpaceWire каждый пользователь использует набор протоколов передачи, где протокол SpaceWire является самым нижним уровнем. Эти протоколы создают уровни стека передачи данных [1]. Устройства, которые будут

настраивать другие устройства, называются контролирующими устройствами, а настраиваемые – периферийными. Периферийные устройства настраиваются с помощью управляющих параметров. Для того чтобы изучать сеть и распознавать петли, необходимым является использование идентификатора устройства. Контролирующее устройство должно считывать информацию с каждого активного соединения в сети и определять по идентификатору было ли ранее изучено периферийное устройство [2].

В протоколе SpaceWire Plug-and-Play используется простая двухуровневая архитектура: сервис сетевого управления и протокол передачи данных. На контролирующем устройстве, благодаря протоколу передачи данных, используется сервис сетевого управления для исследования сети, настройки и идентификации устройств. Данный сервис не обращается напрямую к SpaceWire протоколу, а через драйвер устройства. Это позволяет сервису сетевого управления управлять устройствами, которые не поддерживают стандартный SpaceWire протокол передачи данных или стандарт периферийного устройства сервиса сетевого управления. В тоже время, на периферийном устройстве, имеется возможность осуществлять настройку через настраиваемые параметры [3].

Требования для исследования сети. Для того, чтобы реализовать Plug-and-Play технологию на стандарте SpaceWire необходимо, чтобы каждый коммутатор имел регистр, предназначенный для исследования сети, к которому была бы возможность доступа с использованием протокола RMAP (протокол доступа к удаленной памяти). Он предназначен для конфигурирования узлов сети, а также может использоваться в качестве транспортного протокола, осуществляющего информационное взаимодействие. В нем хранятся данные (статус портов и номер порта), которые используются для доступа к информации. Это соответствует минимальным требованиям, достаточным для реализации алгоритма исследования сети.[2]

Для реализации Plug-and-Play в интеллектуальных узлах, которые называются узлами-менеджерами сети (NNM), создается система с процессором, программируемая алгоритмом PnP. В каждом регионе сети SpaceWire располагается один и более менеджеров сети (NNM). Соответственно каждый коммутатор конфигурируется только одним менеджером. Для чтения и конфигурирования параметров коммутаторов используются команды RMAP.

#### Вывод

Для того, чтобы реализовать Plug-And-Play на имеющемся оборудовании SpaceWire, нужно лишь в некоторых узлах программно исполнить алгоритм PnP. Данный алгоритм не делает каких-либо допущений и ограничений относительно топологии сети, соответственно новые узлы могут быть определены, если в них реализован регистр, к которому имеется доступ с помощью команд RMAP

#### Библиографический список

1. ECSS-E-50-12C. SpaceWire – Links, nodes, routers and networks.// European Cooperation for Space Standardization (ECSS), 31 July 2008.
2. Горбачев С.В., Рождественский Д.А., Суворова Е.А., Шейнин Ю.Е. Масштабируемые архитектуры распределенных систем на технологии SpaceWire на базе платформы "Мультикор" // Вопросы радиоэлектроники. Сер. ОТ, 2006, вып. 2, с. 69–80.
3. Волков П.Л., Горбачев С.В. Применение регионально-логической адресации для организации кластерных сетей SpaceWire // Материалы международной научной конференции, Таганрог-Донецк, 1–6 октября 2001.

УДК 519.216

**Н. Л. Салий**

магистрант кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**Ю. П. Иванов**

доцент, канд. техн. наук – научный руководитель

### ИССЛЕДОВАНИЕ МЕТОДА КОМПЛЕКСНОЙ НЕЛИНЕЙНОЙ ФИЛЬТРАЦИИ СИГНАЛОВ С УЧЕТОМ НАДЕЖНОСТИ ИЗМЕРИТЕЛЕЙ

При оценке качества работы системы оценки навигационных параметров необходимо учитывать взаимосвязь точностных и надежностных характеристик измерителей. Использование комплексной обработки сигналов в случае отказа измерителя позволяет применить методы оптимальной оценки сигналов не только с приемлемой точностью, но и определять состояние измерителя без использования контрольной аппаратуры. Если в рассматриваемой задаче формирования оптимальной оценки с учетом надежности измерителей дополнительно используются показания контрольной аппаратуры о состояниях измерителей и известны апостериорные вероятности, характеризующие апостериорные достоверности, риски изготовителя и заказчика системы контроля, то в этом случае алгоритм квазиоптимальной оценки изменяется незначительно. В связи с этим в работе был исследован ряд алгоритмов оптимальной обработки сигналов с учетом свойств надежности и точности измерителей и проведен сравнительный анализ по точности оценки.

Задана следующая модель измерения [1]:

$$Y_{ki} = X_i + H_{ki}; k = 1, 2; i = 1 \dots N,$$

где  $Z_{ki} = H_{ki} + H_{ki}$ . – входное значение фильтра разностного сигнала.

В работе на основе теоремы Дуба разработан рекуррентный алгоритм определения оптимальной по критерию минимума среднего квадрата ошибки в момент времени усредненной по состояниям измерителей на  $i$ . шаге оценки низкочастотной погрешности, полученной по результатам наблюдения разностного сигнала на интервале измерения  $1 \dots i$ .

$$\hat{H}_{2i}^* = \sum_{m=0}^1 \sum_{m=0}^1 dm_{1,m_2}(z_i, z_{i-1}) \hat{H}_{2i-1,m_2}^*(z_i, z_{i-1}),$$

где  $dm_{1,m_2}(z_i, z_{i-1})$  – апостериорная вероятность;  $\hat{H}_{2i-1,m_2}^*(z_i, z_{i-1})$ . – оптимальная оценка низкочастотной погрешности на выходе фильтра разностного сигнала при условии, что первый измеритель находится в состоянии  $m_1$ . , а второй измеритель – в состоянии  $m_2$ . , и полученные разностные сигналы  $z_i$ . и  $z_{i-1}$  на  $i$  и  $i-1$  шагах [2].

Каждый из измерителей на  $i$ -м шаге может находиться в одном из двух состояний  $m_k(i) = 0, 1; k = 1, 2$ . Состояние измерителя «0» соответствует полной работоспособности измерителя, определяемой заданными средними значениями систематической ошибки  $m_k$ . =  $= 0(i)$ . и стандартного отклонения  $m_k = 0(i)$ . погрешностей измерения, состояние «1» соответствует множеству состояний частичной работоспособности и полной неработоспособности измерителей, определяемых заданными средними значениями систематической ошибки  $m_k = 1(i)$ . и стандартного отклонения  $m_k = 0(i)$ . погрешностей измерения.

При моделировании были выбраны следующие исходные данные [2]:

– корреляционная функция погрешностей измерителя:

$$K_{ki}[m_k] 6^2 [m_k] e^{-\alpha[m_k]^{1-|d|}},$$

– математические ожидания погрешностей измерителей в различных состояниях:

$$M[m_1 = 0] = 0;$$

$$M[m_2 = 0] = 0;$$

$$M[m_1 = 1] = 0,1;$$

$$M[m_2 = 1] = 0,05;$$

– дисперсии погрешностей измерителей в различных состояниях:

$$D[m_1 = 0] = 1;$$

$$D[m_1 = 1] = (20 \dots 50)D_1[m_1 = 0];$$

$$D[m_2 = 0] = 1;$$

$$D[m_2 = 1] = (30 \dots 50)D_1[m_1 = 0];$$

– надежность измерителей в различных состояниях:

$$P_{k0i} + P_{k1i} = 1,$$

где  $P_{k0i}$  – вероятность безотказной работы  $k$ -го измерителя ( $k = 1,2$ ) на  $i$ -м шаге;  $P_{k1i}$  – вероятность отказа  $k$ -того измерителя ( $k = 1,2$ ) на  $i$ -м шаге. При этом выполняется условие нормировки на каждом шаге.

На основе формулы корреляционной функции погрешностей измерителя были исследованы следующие алгоритмы:

1. фильтрация Калмана без учета надежности измерителей;
2. оптимальная «линейная» финитная фильтрация без учета надежности измерителей;
3. оптимальная «линейная» рекуррентная фильтрация с учетом надежности измерителей;
4. оптимальная «линейная» финитная фильтрация с учетом надежности измерителей;
5. оптимальная «нелинейная» рекуррентная фильтрация с учетом надежности измерителей;
6. оптимальная «нелинейная» финитная фильтрация с учетом надежности измерителей.

В работе также получены статистические оценки оптимальных по критерию В. А. Котельникова безусловных достоверностей принятия решений «годен» и «негоден» рисков заказчика и изготовителя для указанных выше алгоритмов (3–6) оценки сигналов в зависимости от длительности интервала наблюдения.

В ходе проведенного исследования, были получены результаты, на основании которых можно сделать следующие выводы:

– среди рассмотренных алгоритмов наибольшей точностью обладает алгоритм оптимальной «нелинейной» рекуррентной фильтрации с учетом надежности измерителей. Больше всего по точности проигрывает алгоритм оптимальной «линейной» финитной фильтрации с учетом надежности измерителей. Между ними по точности распределились алгоритмы оптимальной «линейной» финитной фильтрации с учетом надежности измерителей и оптимальной «нелинейной» финитной фильтрации с учетом надежности измерителей;

– для повышения точности обработки сигналов во всех алгоритмах с учетом надежности измерителей наиболее сильно влияет вероятность безотказной работы измерителей;

– для повышения точности нелинейной обработки сигнала необходимо повышать надежность каждого измерителя (чем она больше, тем меньше дисперсия ошибки оценки);

– при увеличении параметра корреляционной функции  $\alpha$  дисперсия ошибки оценки увеличивается, так как низкочастотная погрешность приближается к белому шуму;

– с увеличением интервала наблюдения разностного сигнала риски изготовителя  $\alpha$  и заказчика  $\beta$  достигнут максимальных значений и затем будут стремиться к установившимся значе-

ниям, определяемым надёжностью и точностью характеристики измерителей и интенсивности восстановления их работоспособности.

**Библиографический список**

1. Иванов Ю. П., Никитин В. Г. Информационно-статистическая теория измерений. Методы оптимального синтеза информационно-измерительных систем, критерии оптимизации и свойства оценок. СПб.: Г1.04 с.
2. Иванов Ю.П., Синяков А.Н., Филатов И.В. Комплексирование информационно-измерительных устройств летательных аппаратов: учеб. пособ. для вузов. Л.: Машиностроение, 1984. 208 с.

УДК 629.7.05

**Г. А. Тарасов**

студент кафедры аэрокосмических измерительно-вычислительных комплексов

**Б. Л. Бирюков**

ст. преп. – научный руководитель

**ТЕМА СТАБИЛИЗАЦИИ БОКОВОГО ДВИЖЕНИЯ БЕСПИЛОТНОГО ЛЕТАТЕЛЬНОГО АПТА В УСЛОВИЯХ ТУРБУЛЕНТНОСТИ**

В настоящее время в авиации все большее внимание уделяется беспилотным летательным аппаратам (БПЛА). Их основное преимущество в сравнении с пилотируемыми в том, что выполнение задач не требует присутствия человека на борту. Современные БПЛА практически вытеснили пилотируемые ЛА в некоторых важных задачах нашей жизнедеятельности, среди которых мониторинг окружающей среды; патрулирование границ; воздушная разведка; помощь при стихийных бедствиях; специализированные сети связи и т.д.

В каждой из вышеперечисленных задач стоит вопрос стабилизации движения, так как постоянно происходит воздействие дестабилизирующих факторов, среди которых одним из главных является воздействие ветра на БПЛА. При отсутствии маневрирования движение ЛА можно разделить на две составляющие: продольное и боковое. В данной работе будет рассмотрен только канал бокового движения, при этом предполагается, что движение в продольном канале стабилизируется отдельно.

Для решения задачи управления боковым движением летательного аппарата рассмотрим [1] его линеаризованные уравнения бокового движения:

$$\left. \begin{aligned} \frac{d\beta}{dt} - \frac{qS}{m} c_z^\beta \beta - \frac{g}{V} \gamma \cos \vartheta - \omega_x \sin \alpha - \omega_y \sin \alpha &= W_z; \\ \frac{d\omega_x}{dt} - \frac{qSl}{J_x} m_x^\beta \beta - \frac{qSl}{J_x} m_x^{\omega_x} \omega_x - \frac{qSl}{J_x} m_x^{\omega_y} \omega_y &= \frac{qSl}{J_x} m_x^{\delta_\alpha} \delta_\alpha + M_{xВ}; \\ \frac{d\omega_y}{dt} - \frac{qSl}{J_y} m_y^\beta \beta - \frac{qSl}{J_y} m_y^{\omega_x} \omega_x - \frac{qSl}{J_y} m_y^{\omega_y} \omega_y &= \frac{qSl}{J_y} m_y^{\delta_\alpha} \delta_\alpha + \frac{qSl}{J_y} m_y^{\delta_\alpha} \delta_\alpha + M_{yВ}; \\ \frac{d\gamma}{dt} - \omega_x + \omega_y \tan \vartheta &= 0; \quad \frac{dz}{dt} = V(\psi - \beta) - U_z; \quad \frac{d\psi}{dt} = \frac{\omega_y}{\cos \vartheta}; \\ \frac{dW_z}{dt} &= -\alpha_v W_z + \mu \cdot n. \end{aligned} \right\}$$

Основное внимание в этой системе уравнений стоит обратить внимание на слагаемое  $W_z$ , состоящее из постоянной составляющей скорости ветра и флуктуационной составляющей, вызванной местной турбулентностью и другими возмущениями.

В соответствии с приведенной системой уравнений, объект управления можем записать в стандартном векторно-матричном виде:

$$\dot{x} = A(t)x + B(t)u + \Psi(t)f,$$

где  $x$  – вектор состояния;  $u$  – вектор управления;  $f$  – вектор внешних возмущений, являющийся гауссовским случайным процессом типа «белый шум»;  $A$  – матрица системы;  $B$  – матрица управления;  $\Psi$  – матрица входа для возмущений.

Для оптимизации данной системы стабилизации будет использован интегрально-квадратичный критерий [2]:

$$J = M \left\{ \int_{t_0}^{t_1} [\varepsilon^m Q(t) \varepsilon + u^m L(t) u] dt + \varepsilon^m(t_1) P^{(1)} \varepsilon(t_1) \right\},$$

где  $\varepsilon(t)$  – ошибка системы.

В этот критерий помимо составляющей, зависящей от ошибки управления, входит составляющая, зависящая от управления, с помощью которой, в частности, можно учесть энергетические затраты на управление. С помощью положительно-определенной матрицы  $Q$  обеспечиваются желаемые весовые коэффициенты для отдельных компонент вектора  $\varepsilon(t)$  по переменным  $\psi, \gamma, z$ , а с помощью аналогичной матрицы  $L$  – весовые коэффициенты по компонентам вектора  $u(t)$ .

Управление с обратной связью по всем компонентам вектора состояния  $x$ , при котором функционал принимает наименьшее значение, имеет вид:

$$u = C^m(t)x,$$

где  $C(t)$  – матричный коэффициент обратной связи, определяемый как:

$$C(t) = -P(t)B(t);$$

где  $P(t)$  – решение матричного уравнения Риккати:

$$-\dot{P}(t) = P(t)A(t) + A^m(t)P(t) - P(t)B(t)B^m(t)P(t) + Q(t);$$

при краевом условии

$$P(t_1) = P^{(1)}.$$

При оптимальном коэффициенте обратной связи  $C(t)$  критерий достигает минимального значения, определяемого выражением

$$J_{min} = tr \left\{ P(t_0)R^{(0)} + \int_{t_0}^{t_1} \Psi R^{(1)} \Psi^m P(t) dt \right\},$$

где  $R^{(0)}$  – ковариационная матрица процесса;  $R^{(1)}$  – матрица чисел, характеризующая интенсивность «белого шума». Запись  $tr \{ \}$  означает след квадратной матрицы.

При получении рассмотренного выше оптимального управления  $u(t)$  предполагалось, что вектор состояния объекта управления  $x(t)$  известен точно.

Учитывая то, что измерители компонент вектора состояния имеют погрешности, в качестве вектора состояния  $x$  должны использоваться их оптимальные оценки. Для их получения может быть использован фильтр Калмана. Допустимость такой замены обосновывается теоремой разделения, на основании которой синтез оптимального стохастического управления для линейной гауссовской задачи при неполной информации о векторе переменных состояния является комбинацией решений задачи оптимального стохастического управления при полной ин-



формации о векторе переменных состояния и задачи оптимального оценивания состояния объекта управления.

В соответствии с выбранным критерием оптимальности структурная схема системы может иметь вид, представленный на рис. 1.

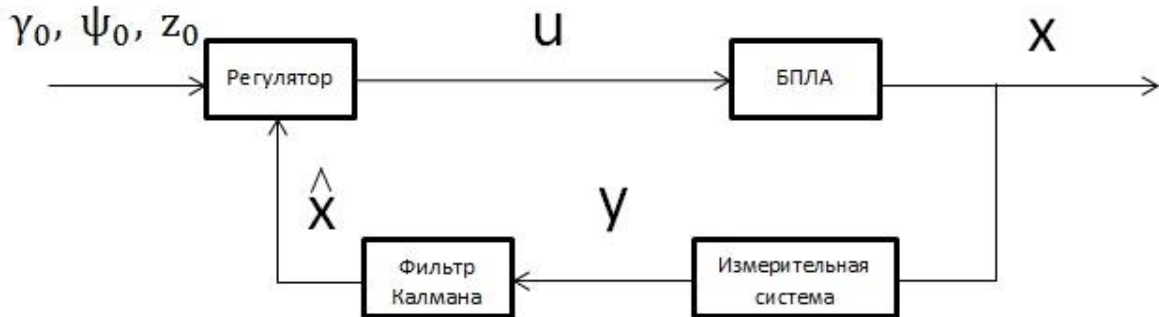


Рис. 1. Структурная схема системы

Применительно к данной модели движения БПЛА было проведено моделирование системы в пакете Mathcad. На рис. 2 приведены диагональные элементы матрицы  $P$  в уравнении Риккати.

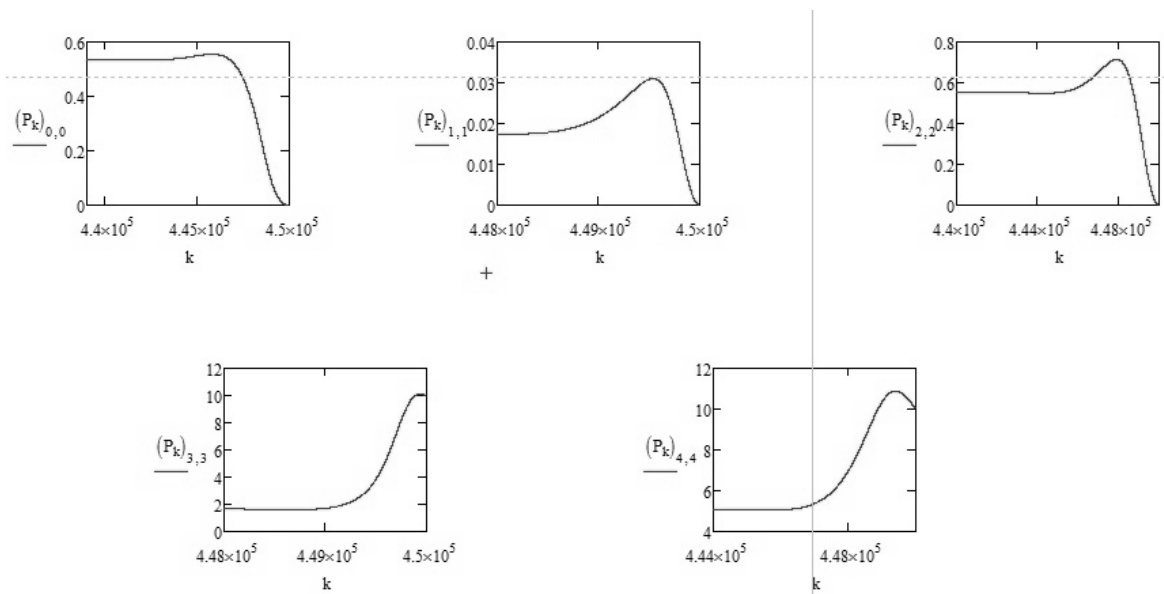


Рис. 2. Зависимость от времени диагональных элементов матрицы  $P(t)$  на конечном этапе процесса управления

ные результаты были получены при безразмерном шаге моделирования по времени  $\Delta t = 2,210^{-5}$ . . Вычисления проведены при краевом условии:

Подставляя в формулу критерия качества полученные в результате моделирования значения матриц, найдем соответствующий минимум, его величина будет равна  $J_{min} = 7,858$ . При расчете данного функционала ковариационная матрица была выбрана следующим образом:

$$R^{(0)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0, \\ 0 & 1 & 0 & 0 & 0, \\ 0 & 0 & 1 & 0 & 0, \\ 0 & 0 & 0 & 1 & 0, \\ 0 & 0 & 0 & 0 & 1. \end{pmatrix}$$

На основе  $P(t)$  были рассчитаны элементы матрицы  $C(t)$ . оптимальных коэффициентов обратной связи. На рис. 3 показаны зависимости элементов этой матрицы по параметрам  $\gamma$  и  $\psi$ .

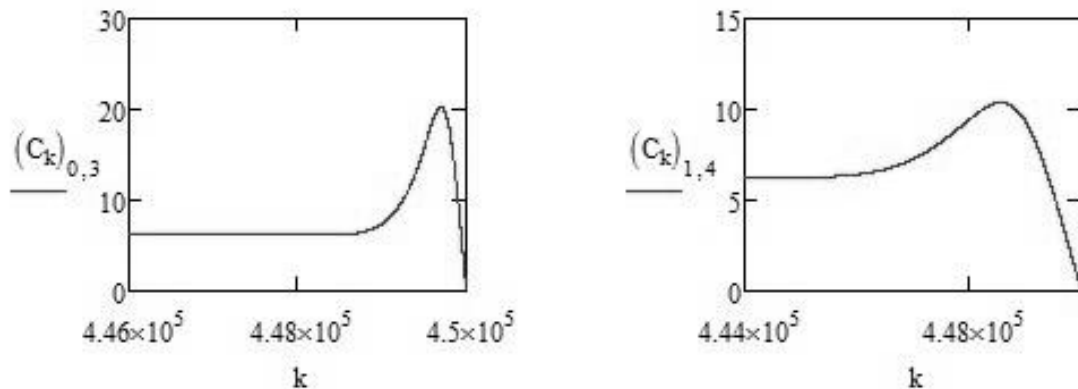


Рис.3. Зависимость элементов  $C(t)$ . матрицы по параметрам  $\gamma$ . и  $\psi$

я полученные зависимости, можно сделать вывод о том, что изменение элементов матрицы коэффициентов наиболее интенсивно происходит на конечном этапе процесса управления, а на других этапах элементы матрицы изменятся незначительно, что необходимо для достижения минимума функционала  $J$ . При этом учитываются энергетические затраты на управление и ошибка системы в конечный момент времени. Присутствие случайных возмущений, действующих на БПЛА, приводит к увеличению значения критерия  $J$ .

### Библиографический список

1. Боднер В. А. Системы управления летательными аппаратами. 2-е изд. М.: Машиностроение, 2010.
2. Александров А. Г. Оптимальные и адаптивные системы: учеб. пособие для вузов по спец. «Автоматика и упр. в техн. системах». М.: Высш. шк., 1989. – 263 с.

УДК 004.414.28

**В. С. Фальзан**

студент кафедры аэрокосмических компьютерных и программных систем

**Ю. Е. Шейнин**

проф., д-р. техн. наук – научный руководитель

### СМАРТ – СИСТЕМЫ В ТУРИЗМЕ

Аннотация

Рассматривается применение геолокационных смарт-систем для построения туристической инфраструктуры и сервисов нового поколения.

### Определение смарт-систем

Смарт-системы обладают встроенными механизмами мониторинга окружающей среды, самостоятельного сбора и анализа данных, принятия решения, а также механизмами обратного воздействия на окружающую среду для приведения ее в соответствие с внутренними целями системы.

Адаптивным способом принятия решений является стратегия поведения системы, направленная на адаптацию системы к изменениям внешней среды, или воздействие на внешнюю среду, приводящее к ее изменению в соответствии с целями системы.

Прогнозирующий способ – это расширенный случай адаптивного, когда для принятия решения используются не только текущие данные, но и внутренняя модель среды, позволяющая оценить развитие ситуации и осуществлять действия, направленные на долгосрочное (спрогнозированное) изменение состояния среды.

Классическая смарт-система включает в себя следующие компоненты (рис.1):

- сенсоры сбора первичной информации;
- блок управления, осуществляющий анализ первичной информации и принимающий решение;
- актуаторы, осуществляющие обратное воздействие системы на окружающую среду;
- линии связи, передающие информацию между вышеперечисленными блоками [1].

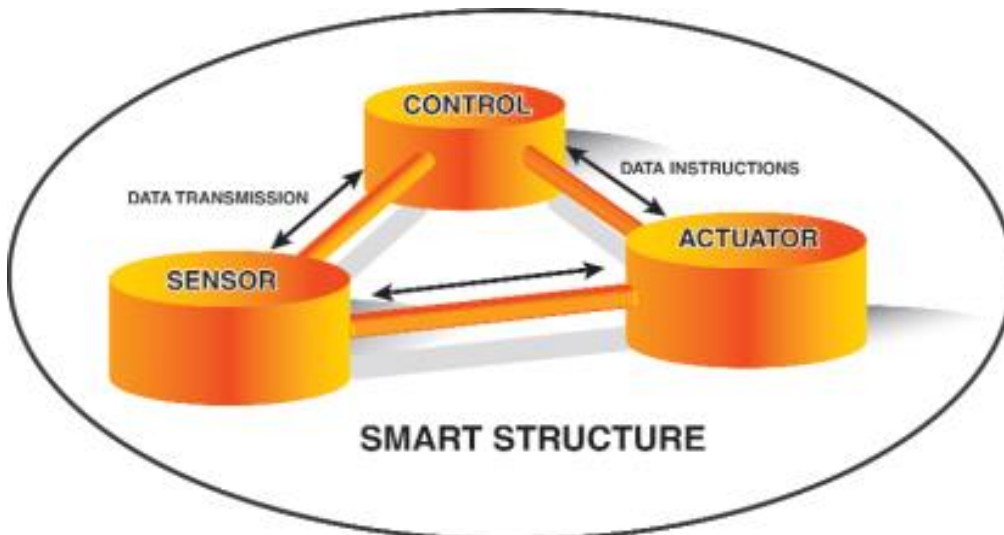


Рис.1. Основные пять компонентов смарт-системы

### Постановка задачи

Основная проблема в технологии смарт-систем – это интеграция множества различных компонентов, разработанных по разным технологиям и произведенных из разных материалов. Фокус – б на проектировании и изготовлении полностью новых рыночных продуктов и услуг для особых применений (например, в медицинских технологиях) и для применений на массовом рынке (например, в автомобильной промышленности).

### Применение

В современном мире рост объема информации вокруг человека идет в геометрической прогрессии, и если в прошлом человек был в состоянии сам обрабатывать информацию, важную для его жизни (или которая может позитивно повлиять на нее), то сейчас это невозможно. Смарт-системы применяются для решения данной проблемы.

Касательно области туризма, смарт-системы позволяют строить сервисы нового поколения, которые за счет своих интеллектуальных действий значительно улучшают качество решений, принимаемых туристами, экономят время и деньги, позволяют глубже погрузиться в зарубежную культуру и, следовательно, увеличивают степень удовлетворения от поездки.

Обзор аналогов

В статье «A worldwide tourism recommendation system based on geotagged web photos» описывается рекомендательная система, которая предлагает туристу места для посещения, основанные на визуальном совпадении и минимальном пользовательском вводе. Пользователь может предоставить либо фото желаемого сценария, либо ключевое слово, которое описывает интересующее его место, и система просматривает базу данных на наличие мест, которые совместно используют эти визуальные характеристики. С этой целью сначала объединяется коллекция крупномасштабных веб – фотографий в группы в зависимости от местонахождения и затем для каждой группы находятся представительные картинки (рис. 2). Рекомендации по туристическим маршрутам предоставляются с помощью сравнения запроса с представительными тегами или картинками под предпосылкой: «Если Вам нравится это место, Вам также могут понравиться эти места» [2].

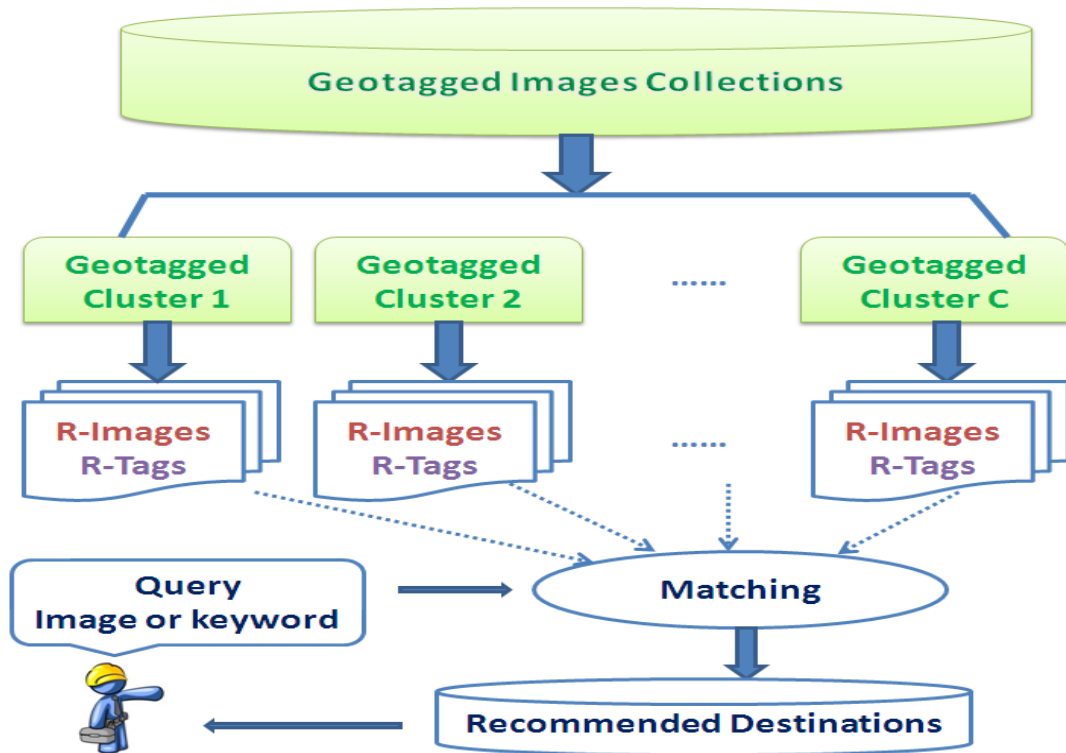


Рис.2. Принцип работы системы, основанной на веб-фотографиях

В статье «STS: Design of Weather-Aware Mobile Recommender Systems in Tourism» рассказывается о мобильной рекомендательной системе с учетом контекста, называемой STS (South Tyrol Suggests), которая учитывает погоду для предоставления более релевантных рекомендаций. Система дает советы по посещению интересных мест в городе Южный Тироль (Италия) благодаря сбору в аккаунте различных условных факторов, включая погоду, в рекомендуемом месте посещения. При использовании в первый раз пользователь вводит свои имя, дату рождения и пол. Затем он отвечает на список из 10 вопросов, благодаря которым система может получить представления о пользователе.

После этого система предлагает пользователю оценить список мест для посещения (рис. 3), чьи оценки, как ожидается, лучше всего улучшат точность последующих рекомендаций. Впоследствии система предоставляет пользователю список из 20 достопримечательностей, которые являются подходящими для текущего пользователя и контекста. Пользователь может получить детальную информацию о каждом из них, а также объяснение рекомендации, исходя из внешних факторов. Например, солнечная погода может послужить хорошей причиной для посещения [3].

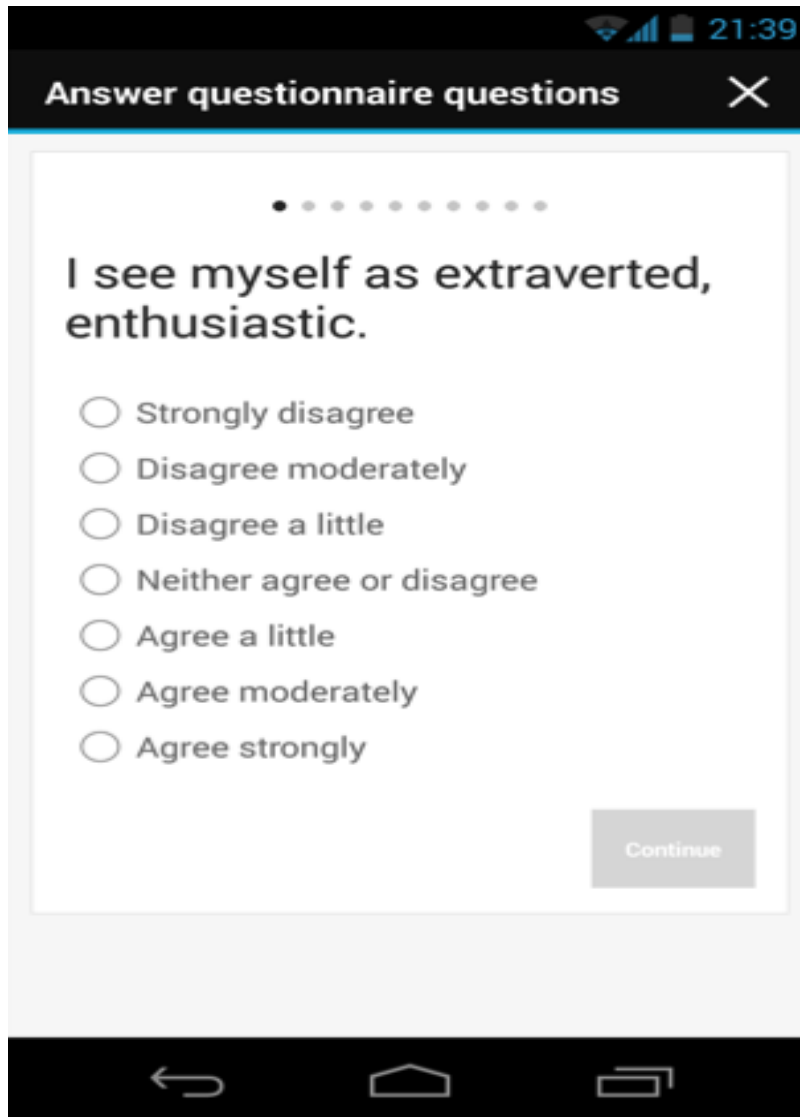


Рис.3. Список из 10 вопросов

В статье «Smart Space-Based Tourist Recommendation System» рассказывается об новой на пространстве умной рекомендательной системе для туристов, позволяющей получать информацию о достопримечательностях вокруг него из различных Интернет – ресурсов (Wikipedia, Flickr и т.д.). Система реализует ранжирование получаемых достопримечательностей согласно предпочтениям туриста и его текущему местонахождению. Туристы могут оценить достопримечательности, которые им нравятся или не нравятся. Основываясь на ранжировании, система объединяет туристов в группы с похожими интересами и использует оценки, принадлежащие к одной и той же группе, для ранжирования интересных мест вокруг туриста. В основе работы лежит технология умного пространства (позволяет предоставлять информацию для совместного использования между различными сервисами системы) [4] (рис. 4).

Статья «Recommendation System for Tourist Attraction Information Service» описывает информационную систему помощи в принятии решений в туризме и набор методов и алгоритмов для генерации рекомендаций для пользователя, которые позволяют значительно увеличивать удобство пользования системой. Система генерирует для пользователя рекомендации, какие интересные места в данный момент лучше посетить, основываясь на предпочтениях пользователя и текущей ситуации на данной территории. Система также позволяет показывать пользователю информацию об интересных достопримечательностях в больших деталях, которая основывается на оценке анализируемых данных, созданных другими пользователями [5] (рис. 5).

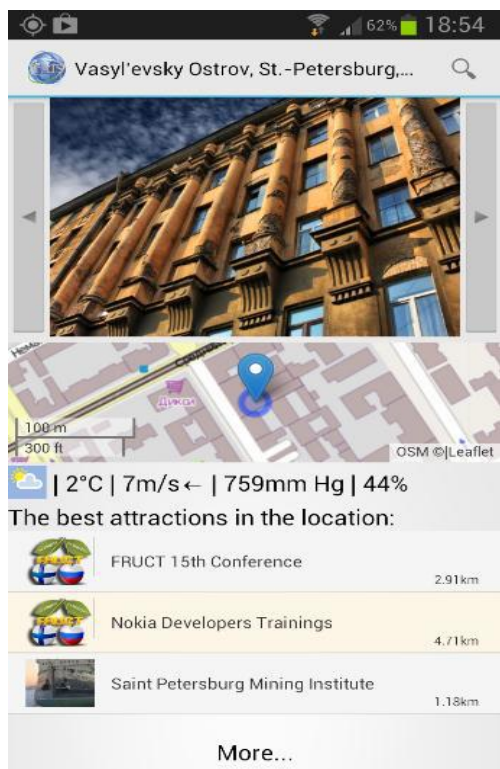


Рис.4. Работа приложения

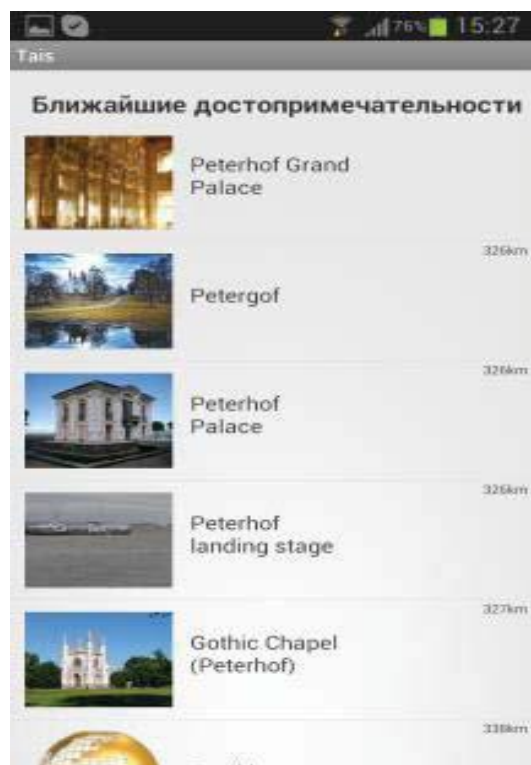


Рис.5. Работа приложения

Статья «Intelligent Tourist Guiding Service Based on Smart-M3 Platform» описывает сервис умного туристического руководства, основанного на информационной платформе Smart – M3 и технологии контекстного менеджмента. Сервис предлагает посетить достопримечательности, основываясь на предпочтениях туриста и текущей ситуации в регионе. Платформа Smart – M3 реализует концепцию умного пространства, представляя инфраструктуру информации совместного пользования в сервисе умного туристического руководства (рис. 6). Умное пространство – это децентрализованная инфраструктура, которая позволяет различным устройствам делиться необходимой информацией между собой. Главное преимущество данного подхода заключается в поддержке туристов в регионе, используя их персональные мобильные устройства. Таким устройствам необходимо лишь подключение к сети Интернет и возможность показывать надлежащую информацию туристам [6].

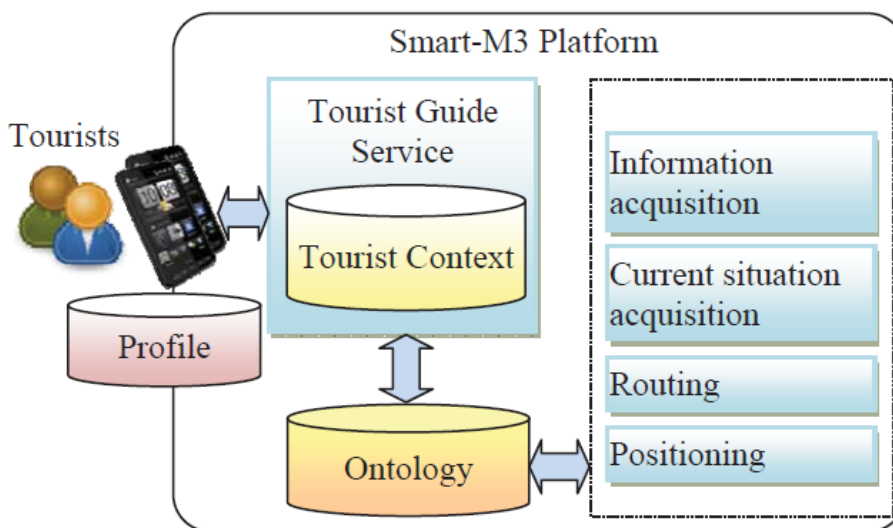


Рис.6. Основная концепция платформы Smart-M3



Статья «Development of a Tourism recommender system» описывает EnoSigTur – умную рекомендательную систему для туристов, связанную с винной отраслью (рис. 7). Через веб – приложение система позволяет пользователям знать о деятельности винного производства в регионе Таррагона (Испания). Пользователям всего лишь необходимо обозначить свои интересы в общих терминах и система будет выбирать наиболее удобные действия для них. Система способна обновлять первоначальную информацию о предпочтениях пользователей путем изучения взаимодействия между пользователем и системой и предложения им более скорректированных рекомендаций. Система также позволяет пользователям планировать свое путешествие посредством предоставления продвинутых сервисов планирования. Мобильное приложение позволит пользователям контролировать запланированное путешествие, пока оно имеет место [7].

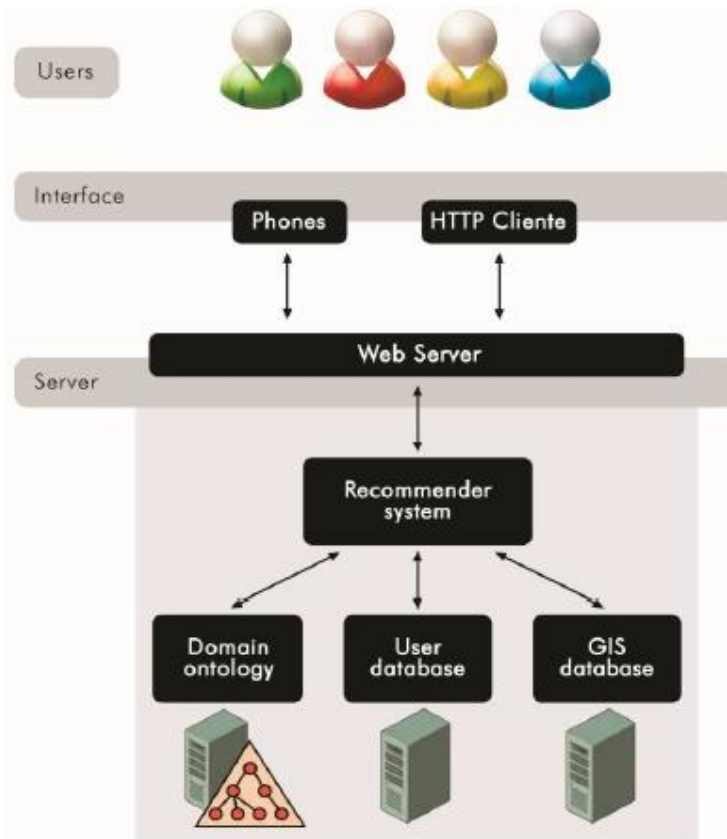


Рис.7. Архитектура системы EnoSigTur

#### Выводы

Проведен анализ существующих смарт – систем с целью выявления текущего состояния соответствующих технологий и уровня реального проникновения данного вида сервисов в выбранной прикладной области. Каждый из рассмотренных аналогов обладает своими преимуществами и недостатками, но в целом все они являются мощными инструментами для улучшения качества туристической поездки.

#### Библиографический список

1. Akhras G. Smart Materials and Smart Systems for the Future// Canadian Military Journal. 2000. pp. 25-31.
2. Liangliang Cao, Jiebo Luo, Andrew Gallagher, and etc. A worldwide tourism recommendation system based on geotagged web photos// 2010 IEEE International Conference on Acoustics, Speech, and Signal Processing. Dallas, 2010. pp. 2274 – 2277.
3. Matthias Braunhofer, Mehdi Elahi, Mouzhi Ge, and etc. STS: Design of Weather-Aware Mobile Recommender Systems in Tourism// Proceedings of the 1st Workshop on AI\* HCI: Intelligent User Interfaces. Turin, 2013. p. 1125.

4. Alexander Smirnov, Alexey Kashevnik, Andrew Ponomarev, and etc. Smart Space-Based Tourist Recommendation System// Internet of Things, Smart Spaces, and Next Generation Networks and Systems ruSMART Conference. St. Petersburg, 2014. pp. 40-51.
5. Alexander Smirnov, Alexey Kashevnik, Andrew Ponomarev, and etc. Recommendation System for Tourist Attraction Information Service// FRUCT Conference. St. Petersburg, 2013. pp.148 – 155.
6. Smirnov A., Shilov N., Kashevnik A., and etc. Intelligent Tourist Guiding Service Based on Smart-M3 Platform// FRUCT Conference. Petrozavodsk, 2013. pp. 121–131.
7. Emili Roger Ciurana Simó. Development of a Tourism recommender system. Polytechnic University of Catalonia, 2012, p. 80.

**УДК 528.9, 681.3.0**

**Е.А. Филиппова**

студентка кафедры аэрокосмических приборов и измерительно - вычислительных комплексов

**В. Г. Никитин**

канд. техн. наук, доц. – научный руководитель

**Д. Д. Никеев**

магистр физики – инженер 1 категории НПК «Пеленгатор»

**Е. В. Гаврилов**

магистр – начальник отд. ОЭС НПК «Пеленгатор»

## **РАЗРАБОТКА АЛГОРИТМА ОПРЕДЕЛЕНИЯ КООРДИНАТ ПОСРЕДСТВОМ СОВМЕЩЕНИЯ ФОТОСНИМКА СЦЕНЫ С ЦИФРОВОЙ КАРТОЙ МЕСТА**

В настоящее время широко используются бесплатформенные инерциальные навигационные системы (БИНС) и спутниковые системы навигации (СНС) летательных аппаратов (ЛА), которые не всегда дают достаточно точную информацию о его местоположении. В БИНС с увеличением времени полета ЛА происходит накопление ошибки в результате двойного интегрирования. В СНС связь может обрываться или задерживаться в зонах, недостижимых в данный момент сигналом спутника или из-за воздействия помех. Развитие оптики, улучшение детализации и качества снимков оптической станции (ОС) на борту ЛА позволяет использовать ее визуальную информацию для задач навигации и целеуказания. Преимуществом этого является работа относительно реальной обработанной информации с поверхности Земли, а не ее модели эллипсоида, как в вышеупомянутых системах. Принцип известных методов коррекции навигационных систем такого рода заключается в совмещении информации видимого или невидимого диапазона с картами местности, которые делятся на корреляционные (КМ) и некорреляционные (НМ) [1]. КМ достаточно распространены и используются для коррекции местоположения ЛА по картам места. Однако, согласно работам [2],[3] характерные изменения яркостно-геометрических параметров вносят высокую вероятность аномальных ошибок, что снижает точность (десятки метров) и надежность метода. Поэтому в настоящее время наблюдается тенденция поиска алгоритмов на основе НМ. Они, как правило, основаны на структурном анализе изображений и поиске характерных черт (линий, областей, точек)(ХЧ). При этом лучшие показатели качества по свойствам ХЧ [1] – инвариантность, устойчивость к шуму, скорость и др. выявлены у НМ по точкам [5], средний – по линиям [4], низкий – по областям.

Особенности информации, получаемой от ОС, включают в себя:

- информацию видимого диапазона (380–730 нм);
- большое число кадров на единицу длины маршрута полета;
- погрешности средств навигации и целеуказания;
- синхронизацию моментов получения информации со средств целеуказания, навигации и кадров изображения.



Учитывая вышеизложенное реализация алгоритма содержит в себе следующие принципы:

1. Преобразование изображения с учетом информации о его угловом положении, полученной посредством ПНС. Для сравнения изображений исходное изображение должно быть снято в горизонтальном полете ЛА, что не всегда возможно. Отклонение от горизонтальности обусловлено погрешностями значений продольного, поперечного и курсового углов ЛА и углов камеры ОС. При таком геометрическом преобразовании искажения сводятся к возможному минимуму и повышается надежность обработки.

2. Фильтрация изображения с использованием морфологических операций с учетом особенностей освещенности сцены и гистограммы оценки фона.

3. Разработанный алгоритм получения координат посредством совмещения снимков камеры и цифровой карты места (ЦКМ) включает следующие этапы:

– предварительную обработку исходного изображения (фильтрация, трансформация, бинаризация и др.);

– нахождение общих ХЧ-изображений (линий), используя преобразование Хафа (Hough Transform [4]);

– получение координат посредством совмещения по найденным ХЧ;

Входными данными алгоритма являются:

– исходное изображение, полученное с ОС в момент времени  $t_n$  (рис.1):

$\{x_{i,j}^{(n)}\}$ ,  $i = 0, \dots, I-1$ ;  $j = \dots, 0$ ;  $n = 0, \dots, N$ ; где  $I, J$  – размеры изображения в пикселях;



Рис.1. Исходное изображение с отмеченной меткой цели

– параметр оптической системы: положение в пространстве (азимут(alfa), угол места(beta));

– цифровая карта местности, полученная в результате обработки измерений средств целеуказания ЛА  $\{p_{i,j}^{(k)}\}$  (рис.2).



Рис.2. Цифровая карта места, полученная по данным ГНС

Общая схема алгоритма представлена на рис.3.

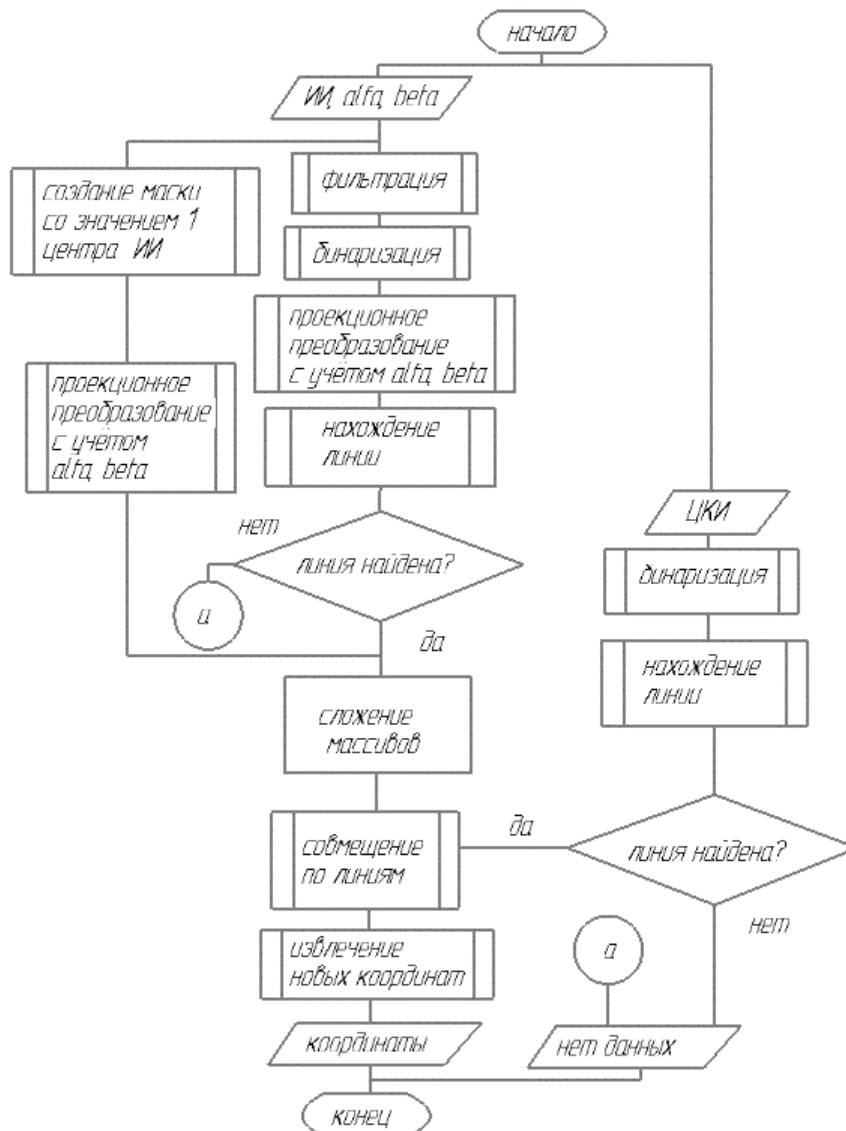


Рис.3 Общая схема алгоритма

Для реализации алгоритма и дальнейшей его оценки была составлена программа в среде MATLAB компании MathWorks. Результаты поиска линий на ИИ и ЦКМ после этапа предобработки представлены на рис.4–5, результат их совмещения по найденной линии на рис.6, где белая область является массивом совмещенной бинаризованной информации со значением 1. После проделанных преобразований программа дает координаты этой точки в системе координат изображения карты и отображает ее на карте места (рис.7):

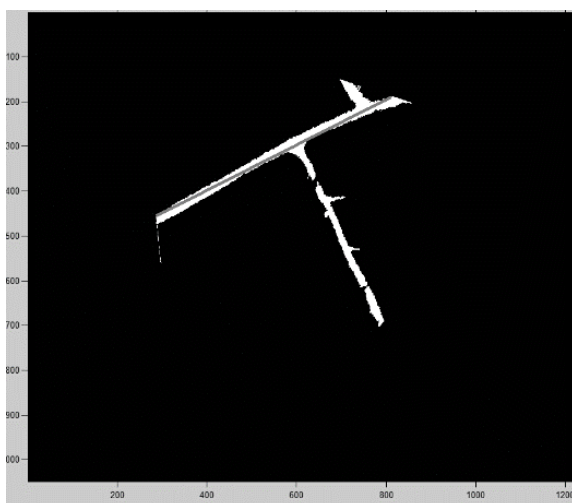


Рис.4. Результат поиска линии на исходном изображении, после предобработки.

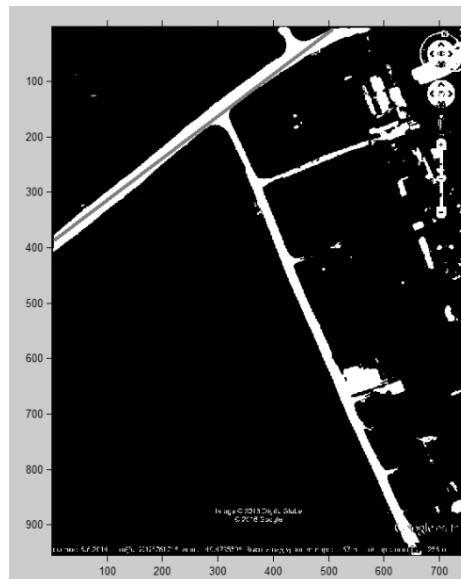


Рис.5. Результат поиска линии на цифровой карте места

После тестирования алгоритма с использованием разного рода реальных сюжетов были выявлены следующие критерии, способствующие повышению точности работы программы:

- фиксированное положение камеры;
- наличие подстилающей поверхности;
- постоянный масштаб съемки;
- постоянная высота и скорость ЛА;
- быстрая скорость обработки вычислителя относительно скорости смены кадров и скорости ЛА;

Для совпадения необходимы следующие критерии:

- наличие массива дорог(линии);
- массив дорог не должен проходить через центр ИИ;
- отсутствие шумов и лишней информации на изображении;
- дневное освещение для камер видимого диапазона.

В связи с невозможностью оценить алгоритм с соблюдением критериев работы алгоритма на всем интервале съемки оценка получена при «грубом» режиме полета ЛА: 7,67 м или 1,7 пикселя, быстродействие 6,3 с (получено на компьютере с характеристиками: процессор Intel(R) Core(TM) i5-4460 CPU®, тактовая частота процессора 3,20 GHz, оперативная память 8 Гб).

Таким образом, предложенный в данной работе алгоритм показал приемлемую точность совмещения, несмотря на «грубый» режим полета, что соответствует требованиям по точности руководства [6]. Полученное быстродействие можно повысить, так как он напрямую зависит от характеристик вычислителя на борту ЛА. Предполагаемая генерация MATLAB кода программы в платформонезависимый код С даст возможность внедрить данный алгоритм в бортовой ком-

плекс ЛА и значительно увеличить скорость нахождения координаты [7]. Разработанный алгоритм создан на базе АО «НПК пеленгатор».

**Библиографический список**

1. Красильщиков М.Н., Себряков Г.Г. Современные информационные технологии в задачах навигации и наведения беспилотных маневренных летательных аппаратов. М.: Книга, 2009 г. 556с.
2. Красовский А.А., Белоглазов И.Н., Чигин Г.П. Теория корреляционно-экстремальных навигационных систем. М.: Книга, 1979г. 448с.
3. Белоглазов И.Н., Джанджгава Г.И., Чигин Г.П. Основы навигации по геофизическим полям. М.: Книга, 1985. 328 с.
4. Richard O. Duda, Peter E. Hart. Use of the hough trasformtion to detect lines and curves in pictures//Communications of the ACM, 1972. 11–15 pp.
5. Harris, С., and M. Stephens. A Combined Corner and Edge Detector//Proceedings of the 4th Alvey Vision Conference, 1988, 147–151 pp.
6. Руководство Р-315 «По минимальным стандартам характеристик авиационных систем (MASPS) для систем улучшенного видения, систем искусственного видения, комбинированных систем искусственного видения и бортовых систем увеличения дальности видения» Авиационного Регистра Межгосударственного Авиационного Комитета (АР МАК), 2012. 86 с.
7. Дьяконов В. П. MATLAB 7 /R2006/R2007. М.: Книга 2008. 768 с.

УДК 004.33

**В. В. Хрусталев**

студент кафедры аэрокосмических компьютерных и программных систем

**С. И. Ковалев**

канд. техн. наук, доц. – научный руководитель

**ПЛАТФОРМА ДЛЯ ИССЛЕДОВАНИЯ РАЗЛИЧНЫХ МИКРОСХЕМ ПАМЯТИ**

Одной из причин преждевременного выхода из строя космических летательных аппаратов (КА) является сбой программного обеспечения, связанный с возникновением ошибок в памяти. На сегодняшний день существует большое количество различных типов памяти, но каждый из них имеет те или иные недостатки. Одной из наиболее передовых и часто используемых в КА является NAND Flash-память [1]. Основным недостатком этого типа памяти является сравнительно небольшой ресурс перезаписей и, как следствие, весьма ограниченное время службы, по достижению которого начинают проявляться кратковременные ошибки, переходящие в полный отказ микросхемы.

В настоящий момент существует большое количество разнообразных предложений по борьбе со сбоями и отказами внутри NAND Flash, начиная с дублирования микросхем памяти и мажоритарного выбора результата до использования кодов, корректирующих ошибки [2].

Тем не менее вопрос о том, какие способы защиты от ошибок в памяти эффективней с точки зрения надежности, скорости, а также финансовой целесообразности, остается открытым из-за нехватки статистической информации, характеризующей сбой в флеш. Данная работа нацелена на разработку специальной платформы, которая позволила бы собрать всю необходимую информацию об ошибках в микросхемах памяти NAND Flash и соответственно, выбрать оптимальный метод борьбы с ними.

Платформа для исследования различных микросхем памяти состоит из трех модулей (плат), взаимосвязанных между собой (рисунок):

- Основная плата;
- Плата с Flash;
- Wi-Fi модуль.

Основная плата включает в себя микроконтроллер, который управляет всей периферией платформы, дополнительную микросхему памяти, цепи стабилизации напряжения и температуры, а также разъемы для подключения двух других модулей. На плате Flash непосредственно установлена тестируемая микросхема памяти, нагревательные элементы и датчик температуры. В качестве Wi-Fi модуля использовано устройство ESP8266.

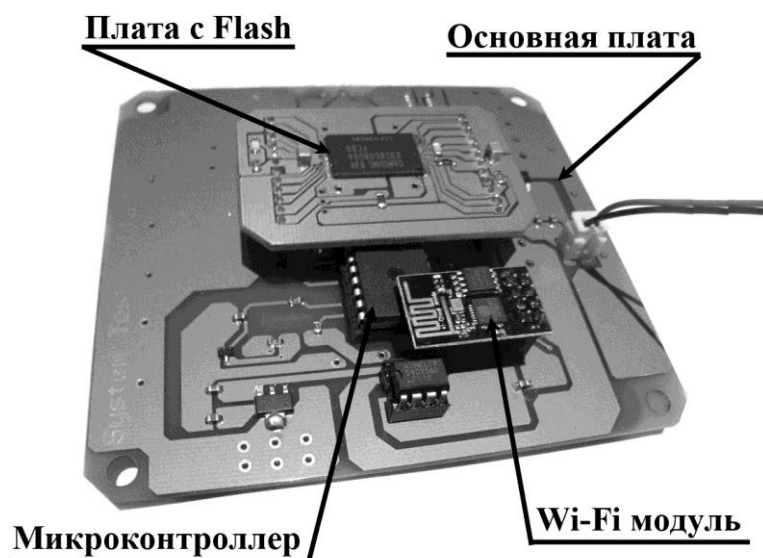


Рис. Платформа для исследования различных микросхем памяти

Такая организация платформы не случайна, она позволяет без особого труда тестировать различные микросхемы памяти, изготавливая для них лишь платы с flash, а не комплекс целиком, собирать данные в едином центре и в режиме реального времени предоставлять доступ к ним всем желающим, а также устанавливать необходимую температуру и напряжение питания NAND Flash.

Нами были изготовлены два экземпляра платформ и написаны две программы для тестирования. Последнее было сделано отдельно двумя программистами для того чтобы исключить алгоритмическую ошибку. В качестве первого экспериментального образца мы выбрали микросхему Samsung K9G8G08U0A двухбитового типа. Этот тип флеш наиболее подвержен ошибкам, так как каждая ячейка может находиться в четырех состояниях, а не в двух, как в однобитной памяти. Первое пробное тестирование страницы со случайно выбранным адресом показало, что количество циклов стирания-записи до первой ошибки составляет около 100 К, что многократно превышает значение, указанное производителем в технической документации, в частности в документации приводится цифра 10К стираний-записей. Данная ошибка довольно быстро восстановилась, т. е. имела вид однократного сбоя. Однако с увеличением числа циклов длительность ошибки, т. е. нахождение ячейки в ошибочном состоянии, плавно увеличивалась и в конечном счете ячейка перешла в состояние полного отказа. Также следует заметить, что параллельно с этой ошибкой возникали и другие кратковременные сбои внутри данной страницы, однако все они были в других байтах. Особенно примечательно то, что все сбои внутри одной страницы имели однонаправленный характер, переходили из низкого состояния в высокое или наоборот, из высокого в низкое. Данная информация была подтверждена при помощи второго экземпляра платформы, работающей под управлением программы, разработанной отдельным программистом, и другой микросхемы.

Таким образом, можно сделать вывод о том, что NAND Flash действительно подвержена ошибкам в памяти. Однако деградация ячеек имеет постепенный характер, ошибочные ячейки периодически возвращаются в рабочее состояние, но с ростом числа перезаписей длительность

ошибочного состояния увеличивается, а рабочего – уменьшается, что в конечном итоге приводит к полному отказу ячейки. Сбои, происходящие внутри одной страницы, асимметричны, причем направление асимметрии зависит от адреса страницы. По крайней мере, такая ситуация характерна для «свежих» чипов памяти, в которых все остальные ячейки, находящиеся вне тестируемой страницы, имеют заполнение вида «все единицы». Это наблюдение наводит на мысль о том, что данная асимметрия связана со способом двоичной нумерации уровней в многоуровневой ячейке Flash-памяти. Разные биты одной ячейки памяти могут принадлежать совершенно разным адресам страниц, что делает ошибки в соседних битах одного байта или даже страницы практически независимыми. Это предположение подтверждается и тем фактом, что за время тестирования не были обнаружены ошибки в смежных битах одного байта. Дальнейшие исследования будут направлены на то, чтобы более точно определить взаимосвязь между ошибками в ячейках памяти, характер асимметрии и условия, при которых она наиболее ярко выражена.

### Библиографический список

1. *Rino Micheloni and other. Inside NAND Flash Memories. 2010. 573 pages.*
2. *Питерсон У., Уэлдон Э. Коды, исправляющие ошибки. М.: Мир, 1964. 593 с.*

УДК 629.7.018.3

**А. О. Чернелевский**

магистрант кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**А. П. Григорьев**

научный руководитель

**В. Я. Мамеев**

доцент, канд. техн. наук – научный консультант

## ЭЛЕМЕНТЫ ДИСКРЕТНОЙ МАТЕМАТИКИ ПРИ ПОСТРОЕНИИ ИНТЕЛЛЕКТУАЛЬНОЙ НАВИГАЦИОННОЙ ТРЕНАЖЕРНО-ОБУЧАЮЩЕЙ СИСТЕМЫ

Переход высшей школы на образовательные стандарты третьего поколения требует кардинальных изменений процесса обучения, существенного повышения гибкости и мобильности образовательного процесса. Таким образом, возникает необходимость формировать учебный план фактически индивидуально для каждого студента. Решение этих задач невозможно без использования современной вычислительной техники, причем требуется автоматизировать не только управление учебным процессом, но и сам процесс обучения. Последнее предполагает создание многочисленных и разнообразных обучающих систем [1].

Важным шагом на пути выявления структуры, присущей знаниям, является построение моделей, в которых в явной форме выделены объекты, образующие эту структуру [1]. В основе этих моделей лежит понятие сети, состоящей из вершин, соединенных дугами. Вершинами этой сети являются понятия (объекты, события, процессы, явления и др.), а дугами – связи или отношения, существующие между этими понятиями. Графическое исполнение элементов сети произвольное. Семантические сети с одной стороны имитируют естественное понимание языка человеком, а с другой – придают фактическим знаниям графовую структурированную организацию, они представляют собой так называемые ориентированные или семантические графы [1, 2]. Построение семантических графов помогает процессу осмысления знаний, способствует их конкретизации, выявлению противоречий, обнаружению недостающей информации и т. п. В качестве примера рассмотрим, как с помощью семантического графа представить учебный материал по параграфу «Навигационный треугольник скоростей» (рис. 1), главы «Учет влияния ветра на полет летательного аппарата» учебного пособия [3].



теризующий индивидуальную траекторию обучаемого (см. рис. 2, в). Данный граф будет учитывать только те связи между УЭ, которые конкретный обучаемый по результатам тестирования не усвоил. Таким образом, можно индивидуализировать процесс обучения на стадии теоретической подготовки и организовать процесс восстановления знаний [5] после тестового контроля [4]. Под восстановлением знаний [5] понимается процедура, при которой по результатам деятельности обучаемого можно не только определить, какие знания недостаточны или ошибочны, но и вернуть его на соответствующий раздел теории или практики, либо дать дополнительные разъяснения.

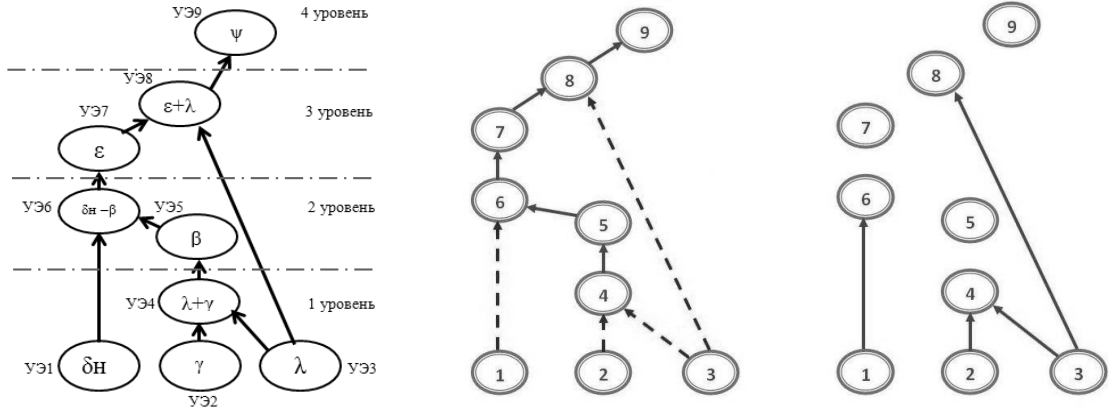


Рис. 2. Графовая структурированная организация: а – семантический граф навигационного треугольника скоростей; б – семантический граф навигационного треугольника скоростей; в – граф по результатам тестирования; г – граф соответствующий

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1)$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (2)$$

$$C = \|A\| - \|B\| = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (3)$$

Также стоит отметить, что семантический граф может быть эффективно использован для ранжирования сложности тестовых заданий [2] при организации адаптивного контроля знаний [4] (см. рис. 2, а).



В данном семантическом графе различают входные, внутренние и выходные концепты. Входные концепты – это понятия или же определения курса ( $\delta_n$  – направление ветра,  $\gamma$  – курс полета,  $\lambda$  – угол сноса и т. д.) (см. рис. 2, б), внутренние концепты используются для определения других понятий курса, являются следующим уровнем графа ( $\beta$  – путевой угол,  $\varepsilon$  – угол ветра и т. д.) и выходные концепты – понятия, не использующиеся для определения других понятий курса ( $\psi$  – курсовой угол). Очевидно, что концепт, связанный с большим количеством других концептов, позиционируется как более сложный. В нашем конкретном случае вопросы, ориентированные на знания входных концептов – 1 уровень (см. рис. 2, б) – считаются наиболее простыми, это, как правило, фундаментальные определения и понятия курса «Воздушной навигации» [3], данным вопросам присваивается наименьший вес. Вопросы, ориентированные на знания концептов высших уровней – 2, 3, 4 соответственно считаются более сложными и, как следствие, имеют больший вес. Таким образом, не трудно моделировать задания различного уровня сложности.

Рассмотрим абстрактную модель обучаемого, представляющую собой вероятностный автомат Мура [6] с двумя состояниями (рис.3).

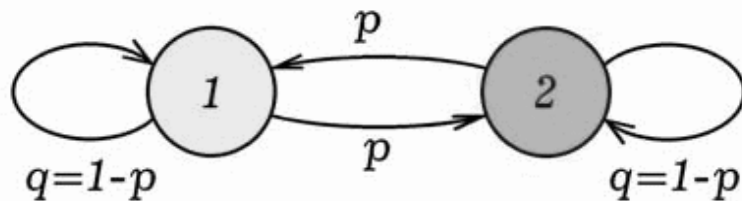


Рис.3. Диаграмма автомата Мура с двумя состояниями

При переходе из одного состояния в другое автомат выполняет некоторую операцию (действие). Будем считать, что автомат обучен, когда из состояния 1 он переходит в состояние 2, а из состояния 2 в состояние 1 и т.д. Если из состояния 1 автомат переходит в состояние 1 или из состояния 2 переходит в состояние 2, то он совершает ошибку. Вероятность правильного действия обозначим через  $p$ , тогда вероятность неправильного действия равна  $q = 1 - p$ . В принципе можно изучать работу вероятностного автомата с большим числом состояний, но при этом всегда он и переходов будет правильным, а остальные – неправильными.

Т как изначально автомат не обучен, то он может совершить 100 различных операций с равными вероятностями, тогда вероятность правильного действия  $p = 0.01$ . Программа, моделирующая ученика, будет содержать цикл, в котором выбор каждой операции осуществляется с помощью генератора случайных чисел. Если случайное число  $x$ , находящееся в интервале  $[0; 1]$ , меньше  $p$ , то автомат Мура совершает правильное действие, если нет – делает ошибку. Процесс обучения приводит к изменению матрицы вероятностей: вероятность правильного выбора  $p$  увеличивается на  $ap$ , где  $a$  – коэффициент научения, а вероятность ошибки уменьшается на ту же величину. Уровень знаний  $Z$  будем считать равным вероятности  $p$  правильного перехода. Чтобы учесть свойство, присущее всем обучаемым – забывание, необходимо на каждом временном шаге уменьшать  $p$  на  $gp$ . ( $g$  – коэффициент забывания) и на такую же величину увеличивать  $q$ :  $p = p - gp; q = q + gp$ . Построенная таким образом программа позволяет промоделировать ситуации: обучение с поощрением, обучение с наказанием, обучение с поощрением и наказанием, подробно описанные в работе [7].

### Библиографический список

1. Лантнев В. В. Модель предметной области и оценка ее сложности в обучающей системе по программированию // Вестник АГТУ. Сер. Управление, вычислительная техника и информатика. 2010. №2.

2. Григорьев А. П., Ивахица Т.А. Графосемантический подход в контроле, диагностике и восстановлении знаний авиационных специалистов: сб. докл. научной сессии ГУАП. СПб.: ГУАП, 2014.

3. Мамаев В. Я., Синяков А. Н., К. К. Петров, Д. А. Горбунов Воздушная навигация и элементы самолётостроения: Учеб. пособие. СПб, 2002 г. 256 с.

4. Глова В. И., Дуплик С. В. Модели педагогического тестирования обучаемых // Вестник Казан, гос. техн. ун-та им. А.Н. Туполева. 2003. №2. С. 74–79.

5. Григорьев А. П., Долгий А. И. Контроль, диагностика и восстановление знаний в дистанционных адаптивных обучающих системах / Monitoring, diagnosis and recovery of knowledge in remote adaptive learning systems. Новые образовательные стратегии в современном информационном пространстве. сб. статей международной ежегодной научной интернет-конференции. СПб.: РГПУ им. А. И. Герцена, 2015. С. 213–216.

6. Пономарев В.Ф. Математические методы и модели в обработке информации управлении. Методические разработки по разделу “Основы теории автоматов”. Калининград: КТИРПИХ, 1992. 72 с.

7. Григорьев А. П., Керн Е. С. Разработка модели содержания и усвоения учебного материала в автоматизированных обучающих системах. 66-я студенческая научно-техническая конференция ГУАП. СПб.: 2013.

**УДК 004.451.44**

**Д. М. Шакуров**

студент кафедры аэрокосмических компьютерных и программных систем

**Е. Р. Иванова**

студент кафедры аэрокосмических компьютерных и программных систем

**С. В. Осмоловский**

аспирант – научный руководитель

## **ВИДЫ И СПОСОБЫ ИЕРАРХИЧЕСКОГО ПЛАНИРОВАНИЯ ЗАДАЧ В СИСТЕМАХ РЕАЛЬНОГО ВРЕМЕНИ НА МНОГОЯДЕРНЫХ ПРОЦЕССОРАХ**

Введение

В настоящее время встраиваемые системы становятся неотъемлемой частью многих современных комплексов, в том числе и систем реального времени (СРВ, real-time systems), обладающих повышенными требованиями к безопасности (safety-critical).

В отраслях промышленности, где используются системы для ответственного применения, таких как авионика, космическая отрасль и автомобилестроение, безопасность совместной работы нескольких программных логических блоков в пределах одной системы может быть достигнута за счет использования иерархического планирования (hierarchical scheduling) прикладных задач (task), из которых состоят программные приложения [1].

Основной целью данной статьи является проведение обзора и сравнительного анализа существующих схем иерархического планирования задач в системах реального времени на многоядерных процессорах, а также определение их ключевых свойств, ограничений и возможных областей практического применения.

Иерархическое планирование

Иерархическое планирование является популярным направлением научных исследований в области СРВ (рис. 1). Этот метод активно используется в аэрокосмической промышленно-

сти для обеспечения изоляции и предотвращения распространения появляющихся ошибок в других частях системы, а также для упрощения анализа и сертификации системы [2].

В иерархическом планировании заложены следующие принципы:

- сложная система разбивается на разделы, которые планируются глобальным планировщиком;
- каждый раздел содержит набор задач, планируемых локальным планировщиком;
- разделы могут иметь интерфейс для работы с глобальными и локальными разделяемыми ресурсами;
- глобальный планировщик обеспечивает совместное планирование всех разделов, гарантируя выполнение временных требований разделов, которое сопровождается тестом планируемости.

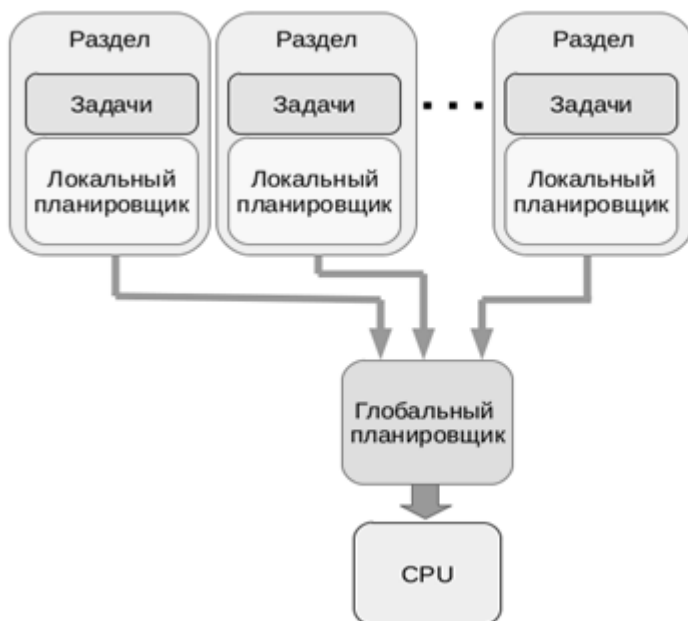


Рис. 1. Метод иерархического планирования

Таким образом, иерархическое планирование обеспечивает временные гарантии и временное и пространственное разделение (изоляция) [3], в частности разделение ресурсов (процессорное время, память, каналы обмена и т.д.) между разделами. Также иерархическое планирование обеспечивает модульность программного обеспечения (ПО), что упрощает сертификацию и повторное использование ПО, делая возможным параллельную разработку и анализ [4]. Метод иерархического планирования схож с механизмом виртуализации: глобальный планировщик выступает гипервизором, а разделы – виртуальными машинами.

Иерархическое планирование позволяет планировать на одной вычислительной многопроцессорной системе приложения различного уровня критичности (mixed-criticality) с различными требованиями реального времени (real-time: hard, soft, non-real time), требованиями по безопасности (mixed-safety) и по защищенности (mixed-security). Планирование возможно производить с помощью различных политик планирования (scheduling policies), например, Rate Monotonic (RM), Earliest Deadline First (EDF), Deadline Monotonic (DM), Least Laxity First (LLF), First Come First Served (FCFS), Round Robin и т.д.

#### Иерархическое планирование на основе серверов

Алгоритмы на базе сервера обладают изоляционными свойствами, что позволяет успешно применять серверы при иерархическом планировании. При этом они становятся составными частями протоколов резервирования ресурсов (resource reservation protocols).

Каждый сервер связан с бюджетом (budget, capacity) процессорного времени и периодом. Бюджет времени – это время работы сервера. Если задача не успевает выполняться в пределах, то происходит заем времени у следующего периода сервера. Бюджет времени восполняется в определенное время пополнения (replenishment time) на рассчитываемую величину пополнения (replenishment amount) [5].

Согласно иерархическому планированию на основе серверов существует один или несколько глобальных планировщиков на первом уровне и несколько серверов на втором уровне. Глобальный планировщик пополняет бюджет серверов, т.е. выделяет им процессорное время, устанавливает для них дедлайны (критический срок, к которому задача должна успеть выполняться) и диспетчирует все готовые серверы согласно своей политике планирования. Локальные планировщики организуют планирование задач, которые закреплены за конкретным сервером. На всех уровнях алгоритмы планирования могут быть произвольными. (рис. 2).

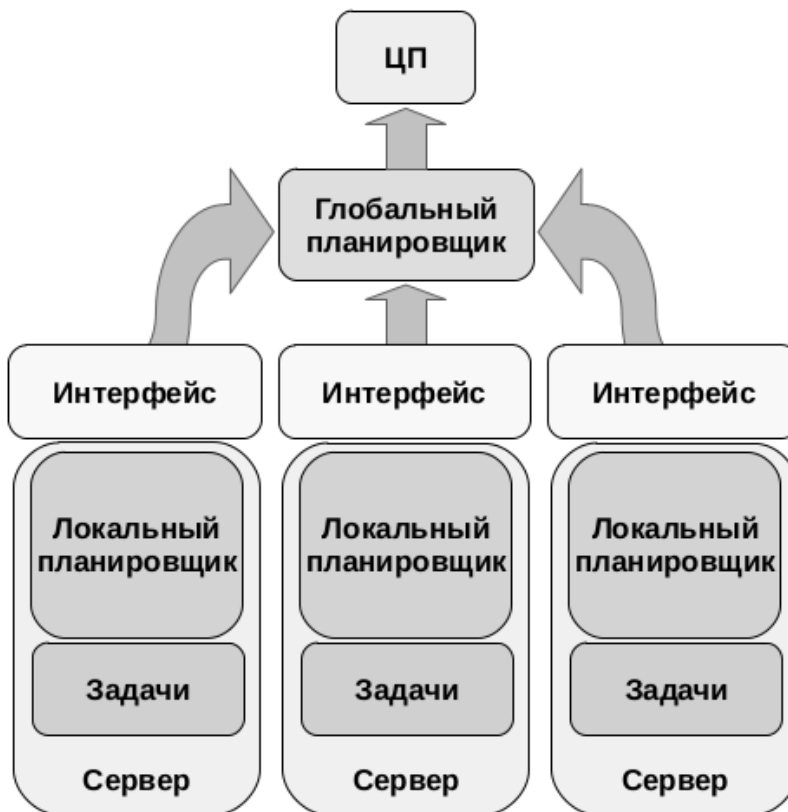


Рис. 2. Метод иерархического планирования на основе серверов

#### Схемы иерархического планирования (СИП)

Под понятием схема иерархического планирования (СИП, Hierarchical Scheduling Framework) понимается реализация метода иерархического планирования.

На сегодняшний день имеются следующие наиболее популярные схемы иерархического планирования для систем реального времени на многоядерных процессорах: СИП с виртуальными кластерами (СИП-ВК) [6], СИП Chesconi [7], СИП Nematı [8], Последовательная СИП (П-СИП) [9].

#### СИП с виртуальными кластерами

Схема иерархического планирования с виртуальными кластерами (СИП-ВК, Virtual Cluster Hierarchical Scheduling Framework), разработанная Shin, является обобщением подхода физической кластеризации с добавлением возможности совместного использования процессоров между различными кластерами. Данная схема позволяет распределять процессоры между кластерами

в процессе работы благодаря интерфейсу, который был предложен Shin и известен как Многопроцессорный Периодический Ресурс (МПР, Multiprocessor Periodic Resource).

СИП-ВК (рис. 3) имеет двухуровневую иерархию планировщиков: межкластерный (inter-cluster, глобальный) и внутрикластерный (intra-cluster, локальный) планировщики. Межкластерный уровень организует динамическое распределение виртуальных кластеров, т.е. производит операцию связывания физического процессора с виртуальным кластером. Все серверы от всех кластеров помещены в очередь в соответствии с глобальной политикой планирования. Очередь задач сортируется по наиболее раннему дедлайну и задачи распределяются по процессорам до тех пор, пока все процессоры не будут заняты. Отметим, что задачи принадлежат определенному кластеру. Внутрикластерный уровень обеспечивает планирование на физическом уровне в кластере. Кластеры не пересекаются и, следовательно, не могут прерывать друг друга. Внутрикластерный планировщик исполняет задачи, закрепленные за кластером используя бюджет планирующих их серверов. В отличие от обычного иерархического планирования, внутрикластерный планировщик также должен использовать алгоритм планирования для многопроцессорных систем, так как у кластера может быть несколько активных серверов.



Рис. 3. СИП-ВК

#### СИП Chesconi

Chesconi предложил собственный вариант СИП, состоящей из двух уровней иерархических планировщиков: локального и глобального (рис. 4). На локальном уровне задачи планируются с помощью совместного многоядерного алгоритма со статическими приоритетами. На глобальном уровне каждый процессор имеет собственный Hard Constant Bandwidth Server (H-CBS) планировщик, который диспетчирует по одному серверу в каждом разделе. Каждый раздел имеет число серверов, равное числу процессоров, и обладает доступом ко всем процессорам. Совместное планирование запускается параллельно с H-CBS планировщиками. H-CBS жестко связан с процессором. Задачи в пределах одного сервера могут перемещаться на другой сервер, который находится на другом процессоре. Сервер имеет в своем распоряжении часть каждого процессора.

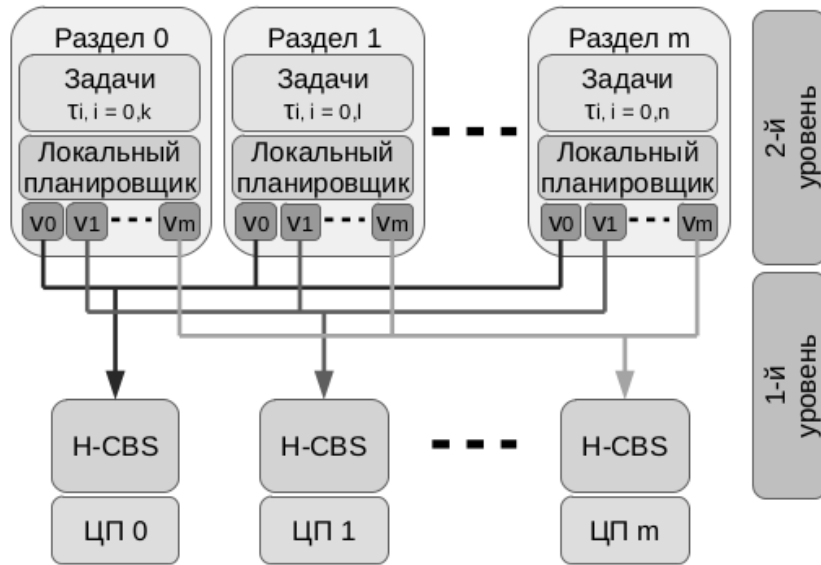


Рисунок 4. СИП Checconi

СИП Nemati

Другая СИП была предложена Nemati (рис. 5). В его схеме серверы планируются согласно совместному многопроцессорному типу планирования (со статическими и динамическим приоритетами), а на локальном уровне каждый раздел планируется согласно распределенному многопроцессорному типу планирования (со статическими и динамическим приоритетами). Каждый раздел имеет максимум один сервер (он может исполняться на любом процессоре), задачи всегда исполняются на одном и том же процессоре.

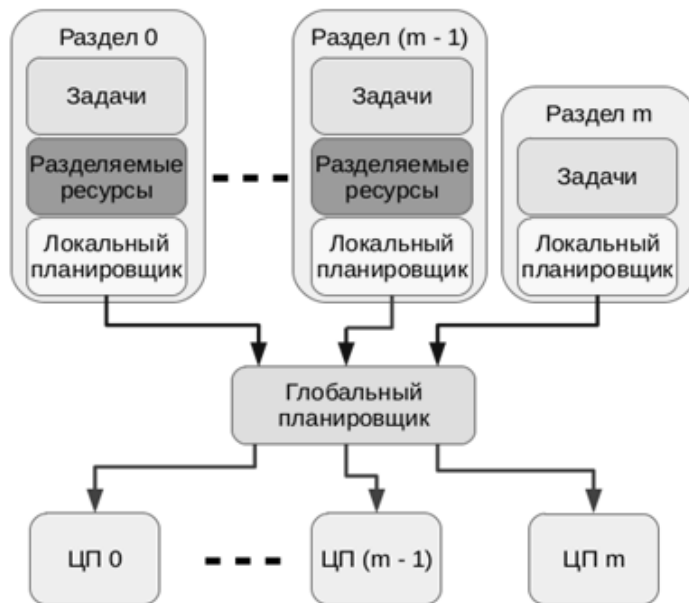


Рисунок 5. СИП Nemati

Последовательная СИП

Еще одна схема – это последовательная СИП (П-СИП, Sequential HSF), предложенная Asberg и др. (рис. 6). Этот подход обеспечивает планирование серверов последовательно, таким образом, планирование задач в каждом разделе происходит с использованием совместных алгоритмов планирования на всех процессорах. Кроме того, данная схема будет выполнять разделы последовательно, занимая каждый процессор одним сервером. Архитектура данной схемы изображена на рис. 6.



Рисунок 6. Последовательная СИП

#### Сравнение схем иерархического планирования

На основании приведенных описаний схем иерархического планирования можно провести их сравнительный анализ.

СИП-ВК имеет наиболее сложный планировщик серверов (на глобальном уровне), так как может быть задействовано наибольшее количество процессоров, занятые процессоры должны быть проверены при планировании сервера. СИП Chessoni является более простой, так как все серверы связываются с процессором статически в режиме офлайн. Действительно, одного глобального планировщика H-CBS может быть достаточно для обработки всех серверов, что делает его похожим на П-СИП. Подход Nematі довольно прост, поскольку существует максимум один сервер на разделы, однако необходимо постоянно проверять доступность процессоров, так как серверы назначаются на них не статически, а динамически. П-СИП проста, так как серверы назначаются на процессоры статически.

Взглянув на локальное планирование, можно сказать, что СИП-ВК и СИП Chessoni имеют схожую схему планирования на локальном уровне. Обе используют совместное многопроцессорное планирование на подмножестве процессоров. СИП Nematі использует обычное распределенное планирование, в то время как П-СИП – совместное многопроцессорное планирование на всех ядрах.

Подводя итог, в СИП-ВК и СИП Chessoni планирование наиболее алгоритмически сложное, чем в П-СИП, а СИП Nematі является наиболее простой.

С точки зрения общих (разделяемых) ресурсов, так как СИП Nematі использует распределенное планирование, то работа с разделяемыми ресурсами внутри раздела становится менее сложной в данном подходе по сравнению с тремя другими схемами.

Краткий обзор результатов сравнения четырех СИП для многоядерных процессоров приведен в таблице.

Схема	Многопроцессорное планирование	Параллелизм серверов	Назначение серверов процессорам	Алгоритмическая сложность
СИП-ВК	Кластерное	Есть	Онлайн	Высокая
СИП Chessoni	Совместное	Есть	Офлайн	Высокая

Схема	Многопроцессорное планирование	Параллелизм серверов	Назначение серверов процессорам	Алгоритмическая сложность
СИП Nemati	Раздельное	Есть	Онлайн	Низкая
П-СИП	Совместное	Нет	Офлайн	Средняя

#### Заключение

Согласно приведенным характеристикам СИП они могут применяться в качестве базовой программной архитектуры в СРВ для ответственных применений. Разработчики программного обеспечения для подобных систем смогут определить, какой тип планирования и СИП лучше использовать для решения конкретных задач, состава и характеристик целевой СРВ.

Иерархическое планирование удовлетворяет значительному количеству требований аэрокосмических стандартов и спецификаций. Отметим, что спецификация ARINC-653 [10] описывает двухуровневую СИП с использованием концепции разделов, а стандарт MILS требует модульности приложений и также оперирует понятием раздела.

Таким образом, иерархическое планирование является актуальным направлением научно-исследовательских работ в области СРВ и обладает большим потенциалом практического использования в системах для ответственного применения во многих отраслях промышленности.

#### Библиографический список

1. Chakma K. A, S. *Debbarma*, N. Kar, N. *Debbarma*, T. *Debbarma* Hierarchical Scheduling Approach for Symmetric Multiprocessing Based Real Time Systems on VxWorks // Lecture Notes on Software Engineering, Vol. 1, No. 1, February 2013, pp. 61-65.
2. *Asberg M.* On the Development of Hierarchical Real-Time Systems // Licentiate Thesis, Mälardalen University, Sweden, 2012, p. 156.
3. *Rufino J., Craveiro J., and Verissimo P.* Architecting robustness and timeliness in a new generation of aerospace systems // Architecting Dependable Systems VII, ser. Lecture Notes in Computer Science, A. Casimiro, R. de Lemos, and C. Gacek, Eds. Springer, 2010, vol. 6420, pp. 146 – 170.
4. *Asberg M., Nolte T., Kato S.* Loadable Task Execution Recorder for Hierarchical Scheduling in Linux // Proc. of the Embedded and Real-Time Computing Systems and Applications (RTCSA), vol. 1, 2011, pp. 380–387.
5. *Осмоловский С. В.* Метод многоуровневого планирования задач в операционных системах реального времени для многоядерных процессоров // Научная сессия ГУАП: Ч.1 Технические науки: сб. докладов 6-10 апреля 2015 г., СПб.: ГУАП, 2015, с. 109-116.
6. *Shin I., Easwaran A., and Lee I.* Hierarchical scheduling framework for virtual clustering of multiprocessors. // Proceedings of the 2008 Euromicro Conference on Real-Time Systems. IEEE Computer Society, 2008, pp. 181–190.
7. *Checconi F., Cucinotta T., Faggioli D., and Lipari G.* Hierarchical Multiprocessor CPU Reservations for the Linux Kernel // Proc. of the 5th International Workshop on Operating Systems Platforms for Embedded Real-Time Applications, June 2009, pp. 15-22.
8. *Nemati F., Behnam M., and Nolte T.* Multiprocessor Synchronization and Hierarchical Scheduling // Proc. of the 1st International Workshop on Real-time Systems on Multicore Platforms: Theory and Practice, in conjunction with ICPP'09, September 2009, pp. 58 – 64.
9. *Asberg M., Nolte, and S. Kato* Towards Hierarchical Scheduling in Linux/Multi-core Platform, T. // Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on 13-16 Sept. 2010, pp. 1-4.
10. ARINC 653: Avionics Application Software Standard Interface (Draft 15). Airlines Electronic Engineering Committee (AEEC), 1996, pp. 14-1–14-7.



УДК 629.735.015

**С. А. Ширяев**

студент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

**М. Е. Тихомиров**

доц. – научный руководитель

## МОДЕРНИЗАЦИЯ БОРТОВОГО УСТРОЙСТВА РЕГИСТРАЦИИ ПОЛЕТНОЙ ИНФОРМАЦИИ

Цель исследования: Анализ особенностей модернизации бортовых устройств регистрации (на примере системы БУР-ЛКВ).

В середине прошлого века Международная организация гражданской авиации (ICAO) приняла постановление об обязательной установке средств объективного контроля на все самолеты и вертолеты. Объективный контроль – это контроль, основанный на обработке информации, полученной с помощью технических средств. Одними из этих средств являются бортовые устройства регистрации (БУР). БУР предназначены для сбора, накопления и сохранения информации о параметрах полета, звуковой и речевой информации. Накопленная в полете информация позволяет не только расследовать причины летного происшествия, но и решать задачи эксплуатационного контроля авиационной техники и контроля техники пилотирования.

Наряду с разработкой нового авиационного оборудования проводится соответствующая модернизация оборудования, находящегося в эксплуатации, для поддержания их соответствия современным требованиям.

Для бортовых устройств регистрации, устанавливаемых на самолетах и вертолетах, актуально:

1. Соответствие характеристик систем современным российским и международным нормативным документам АП-21, АП-23, АП-25, АП-29, КТ-160D, ОСТ 1 01080-95, TSO-C123A, TSO-C124A, ED-55, ED-56A, ED-112.

2. Улучшение технических характеристик по сохранению зарегистрированной информации в случае авиационного происшествия.

3. Минимизация габаритно-весовых характеристик.

4. Совмещение задач по регистрации в одной системе параметрической и речевой/звуковой информации.

5. Увеличение возможностей регистрации параметрической информации.

6. Повышение надежности.

Проводится модернизация бортовой системы сбора и регистрации полетной информации БУР-ЛКВ с твердотельным накопителем, предназначенной для приема и регистрации полетной параметрической информации, сохранения этой информации в случае летного происшествия и решения задач эксплуатационного контроля.

В состав модернизированной БУР-ЛКВ (далее БУР-ЛКВ-01 – рисунок) входят:

- модернизированное бортовое устройство регистрации БУР-ЛК (далее БУР-ЛК-01);
- модернизированный блок согласования БС-ЛК (далее БС-ЛК-01);
- модернизированный пульт управления ПУ-60 (далее ПУ-СН);
- блок микрофонный.

БУР-ЛК представляет собой средство сбора полетной параметрической информации, т. е. многоканальный программируемый преобразователь аналоговых сигналов и разовых команд со встроенным защищенным твердотельным накопителем ЗТН-1, предназначенным для приема, сохранения и воспроизведения кодовой информации.

ПУ-60 представляет собой электромеханический блок, предназначенный для:

- для включения системы БУР-ЛКВ (ручного и автоматического);

- формирования сигнала «Состояние» каждого блока БУР-ЛКВ;
- механической установки даты полета с помощью дисковых переключателей (ДП);
- формирования номера борта ЛА с помощью кодового переключателя (КП);
- формирования цифровых каналов передачи информации между ПУ-60 и БС-ЛК;
- подключения к ПЭВМ при наземном техническом обслуживании БУР-ЛКВ (копирование информации на борту ЛА, градуировка).

БС-ЛК представляет собой средство приема и преобразования аналоговых сигналов, разовых команд и кодовых сигналов в аналоговые сигналы и разовые команды, воспринимаемые БУР-ЛК. Он предназначен для:

- расширения возможностей БУР-ЛК по диапазону и типу регистрируемой информации
- формирования цифровых каналов передачи информации между БУР-ЛК и ПУ-60.

Применение новых композиционных материалов и специально разработанная технология защиты аварийного регистратора БУР-ЛК-01 позволят повысить показатели сохранности записанной информации после воздействия на БУР-ЛК-01 таких внешних факторов, как окружающая температура, ударная перегрузка, пребывание в морской воде и авиационных жидкостях и т. д.

Конструктив системы БУР-ЛКВ также должен быть подвержен модернизации, так как электронная составляющая системы претерпела существенное уменьшение в размере. В основу конструирования БУР-ЛК был заложен принцип блочно-модульного построения изделия. Конструкция БУР-ЛК-01 представляет собой площадку, с одной стороны которой размещена электроника прибора, а с другой – аварийный защищенный твердотельный накопитель (ЗТН). Площадка по своим габаритам соответствует габаритам устаревшего БУР-ЛК в целях сохранения присоединительных размеров. В этом случае при замене БУР-ЛК на модернизированное бортовое устройство регистрации не потребуется доработка конструкции воздушного судна, на которое оно будет устанавливаться. Конструктив БУР-ЛК-01 в отличие от его предшественника включает в себя кронштейн для крепления подводного акустического маяка ПАМ-6к. В соответствии с современными требованиями, описанными в документе ED-112, маяк не должен отделяться от ЗТН при воздействии определенных установленных внешних факторов, а также способ крепления должен гарантировать минимальный риск повреждения маяка. Это позволит повысить эффективность поисков БУР в случае авиакатастрофы над водной поверхностью и возможность обнаружения на глубине до 6000 м. Одной из основных модернизаций является разработка съемного эксплуатационного накопителя в составе пульта управления ПУ-СН, что позволит оперативно осуществлять объективный контроль в целях оценки правильности функционирования оборудования воздушного судна между полетами, а также действий членов экипажа. В результате такой модернизации системы сбора и регистрации полетной информации БУР-ЛКВ улучшатся технические характеристики по сохранению зарегистрированной информации в случае авиационного происшествия, повысится надежность и будет достигнута задача минимизации габаритно-весовых характеристик.

В начале внедрения бортовых устройств регистрации на ЛА для записи параметрической и звуковой/речевой информации использовались отдельные системы. В настоящее время актуально объединение функций записи информации данных систем в одну систему. БУР-ЛКВ является регистратором только параметрической информации. При проведении модернизации в систему вводится дополнительная функция регистрации звуковой/речевой информации. Эта задача решается путем введения необходимых функциональных узлов, специально разработанных для приема и сохранения такой информации. В результате модернизации в данной системе обеспечивается прием речевой информации по трем каналам от переговорной системы воздушного судна и звуковой информации по одному каналу от открытого микрофона, а также последующая запись полученной информации во встроенный защищенный модуль памяти в режиме «кольцо».

Копирование информации из накопителя рассматриваемой системы осуществляется с использованием наземного устройства воспроизведения типа НУВ или ПЭВМ. При копировании

звуковой/речевой информации создаются файлы формата WAV, что позволяет осуществлять воспроизведение звуковой/речевой информации большинством наземных устройств воспроизведения, имеющихся в эксплуатации.

С появлением цифровой электроники встала задача модернизации устаревших БУР, регистрирующих только аналоговые сигналы и разовые команды, с целью внедрения в них возможности регистрации цифрового последовательного кода. Для достижения поставленной задачи необходимо произвести доработку блоков системы БУР-ЛКВ с целью добавления в систему каналов для записи двуполярного последовательного кода с характеристиками по ГОСТ 18977 и РТМ 1495-75 с изм. 3.

Современная электронная база позволяет не только совместить запись параметрической и звуковой/речевой информации в одной системе регистрации полетной информации, а также увеличить возможности регистрации параметрической информации. БУР-ЛКВ-01 имеет возможность обеспечивать продолжительность регистрации цифровой параметрической информации не менее 700 последних часов работы при скорости записи 1024 12-битных слов/с.

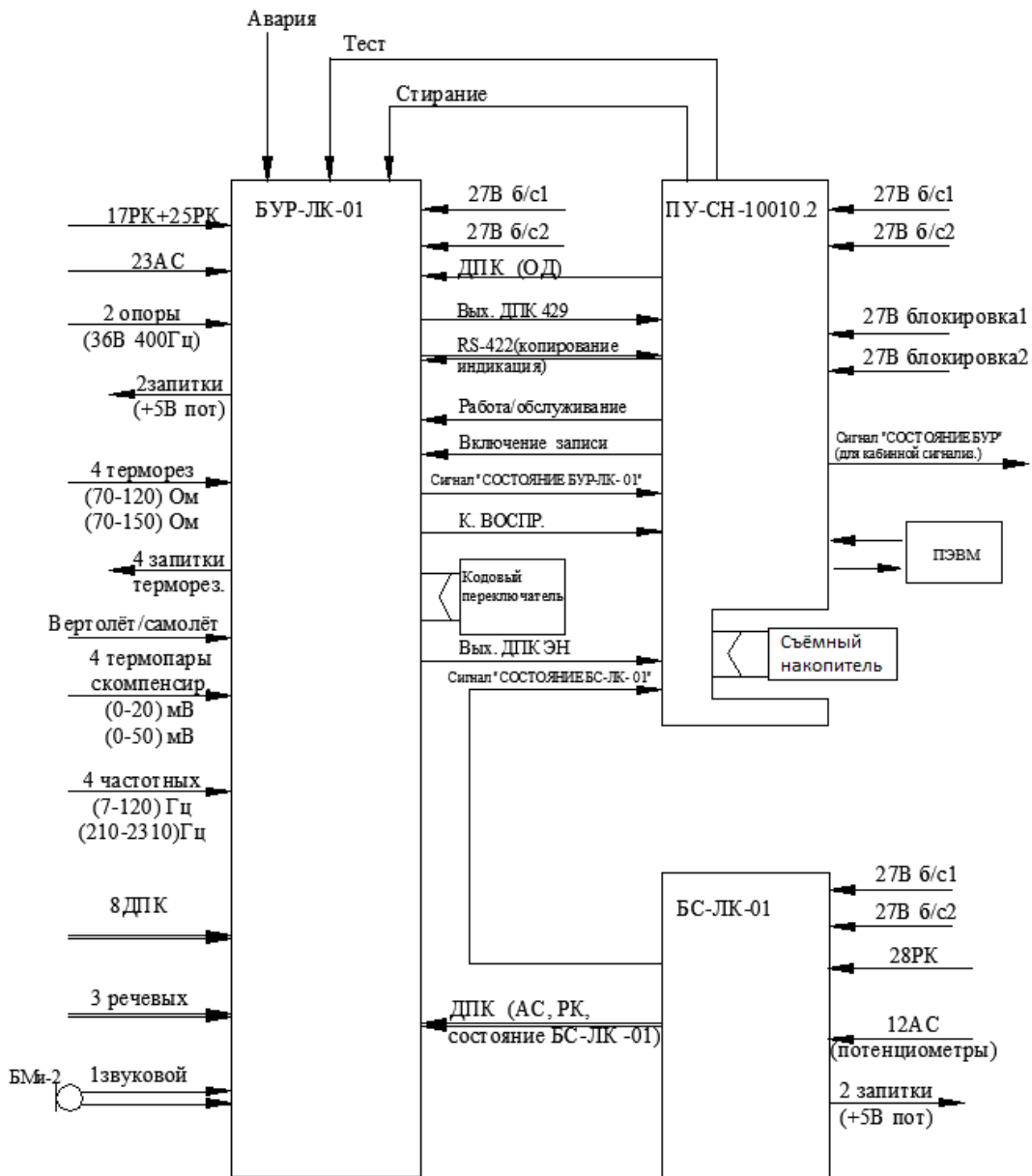
Для повышения надежности, в соответствии с директивными документами ED-112 запись параметрической и речевой/звуковой информации производится на отдельные микросхемы. Микросхемы с целью уменьшения занимаемого пространства следует разместить одна над другой. Результаты анализа последствий авиакатастроф показывают целесообразность установки двух совмещенных защищенных бортовых накопителей на ВС. Метод резервирования увеличивает вероятность сохранения записанной полетной информации. Совмещенные регистраторы могут принести выгоду авиакомпаниям, поскольку они обеспечивают унификацию частей и контрольно-измерительного оборудования, а также сокращение продолжительности технического обучения.

В результате проведения модернизации системы БУР-ЛКВ будут получены следующие основные характеристики, представленные в таблице, соответствующие техническому заданию на систему (таблица).

Сравниваемая характеристика	БУР-ЛКВ	БУР-ЛКВ-01
Сохранение параметрической информации при скорости регистрации 64 слова/с, ч	5,5	400
Максимальная скорость регистрации, слов/с	64	1024
Масса, кг	7,5	6,5
Наличие каналов звуковой/речевой информации	Отсутствуют	Есть
Виды регистрируемой параметрической информации	АС, РК*	АС, РК, ДПК*
Количество каналов приема информации АС	29	48
Количество каналов приема информации РК	29	67
Количество каналов приема информации ДПК	Отсутствуют	8
Наличие съемного эксплуатационного накопителя	Отсутствует	Есть
<b>Система должна обеспечивать сохранение записанной информации после воздействия:</b>		
– ударной перегрузки	До 1000 g в течение 5 мс	До 3400 g в течение 6,5 мс
– окружающей температуры	До 750 °С в течение 15 мин с охватом 50% общей поверхности БУР-ЛК	До 1100 °С в течение 1 ч на 100% поверхности БУР-ЛК-01
– пребывание в морской воде	До 36 ч	До 30 суток
– авиационных жидкостей	В течение 10 мин	В течение 200 ч

\* АС – аналоговый сигнал;  
РК – разовая команда;

ДПК – двуполярный последовательный код.  
 Блок-схема системы БУР-ЛКВ-01 представлена на рис.



Выводы по результатам проведения модернизации системы БУР-ЛКВ-01:

1. Возможности представленной бортовой системы сбора и регистрации полетной информации БУР-ЛКВ-01 позволяют осуществить мониторинг действий экипажа и работу оборудования в процессе полета и сохранить его результаты для анализа как в случае летного происшествия, так и в процессе эксплуатации.
2. Конструктивная, электрическая и интерфейсная совместимость системы БУР-ЛКВ-01 с эксплуатируемыми системами позволяет провести ее внедрение с наименьшими затратами.
3. Наличие части программного обеспечения, адаптируемой к конфигурации самолета, в виде наборов данных позволяет оперативно без доработки блоков осуществлять с помощью программных средств перепрограммирование системы при изменении бортового оборудования и перечней регистрируемых параметров.
4. Возможности представленной бортовой системы сбора и регистрации полетной информации БУР-ЛКВ-01 позволяют осуществить мониторинг действий экипажа и работу оборудования

в процессе полета и сохранить его результаты для анализа как в случае летного происшествия, так и в процессе повседневной эксплуатации.

5. Конструктивная, электрическая и интерфейсная совместимость системы БУР-ЛКВ-01 с эксплуатируемыми системами позволяет провести ее внедрение с наименьшими затратами. Наличие части программного обеспечения, адаптируемой к конфигурации самолета в виде наборов данных позволяет оперативно без доработки блоков осуществлять с помощью программных средств перепрограммирование системы при изменении бортового оборудования и перечней регистрируемых параметров.

#### **Библиографический список**

1. Квалификационные требования КТ-160D, условия эксплуатации и окружающей среды для бортового авиационного оборудования.
2. ED-112. Minimum operational performance specification for crash protected airborne recorder systems). ОСТ 1 01080-95. Общие технические требования. Устройства регистрации бортовые с защищенными накопителями.
3. Техническая документация на бортовую систему сбора и регистрации полетной информации БУР-ЛКВ предприятия АО «НПО «ПРИБОР».

**УДК 681.2.085**

***И. А. Шкаликов***

студент кафедры аэрокосмических приборов и измерительно-вычислительных комплексов

***Д. Ю. Крысин***

ассистент – научный руководитель

### **СТРУКТУРА ЭЛЕКТРОННОГО МАГНИТНОГО КОМПАСА**

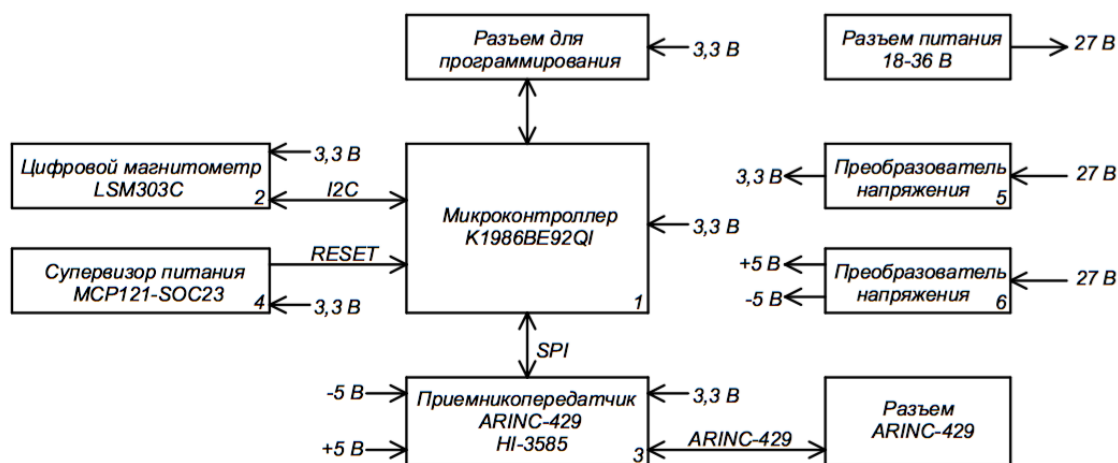
Магнитные компасы были разработаны в III в. до н.э. и продолжают широко использоваться в настоящее время. Для авиации они играют большую роль, так как являются надежным измерителем магнитного курса. Их принцип действия основан на явлении земного магнетизма. Земной шар является естественным магнитом, поэтому обладает магнитными свойствами. В разных точках Земли магнитное поле имеет свою силу и направление. Сила магнитного поля характеризуется напряженностью, которая может быть разложена на две составляющие: горизонтальную и вертикальную [1].

С развитием техники появились различные виды компасов: индукционный, гирополукомпас, астрокомпас и др. Большинство из них характеризуются значительной массой, габаритами, сложностью калибровки, установки на борту летательного аппарата, высокой ценой, трудоемкостью в изготовлении и производстве. Между тем, в настоящее время непрерывно расширяется сфера применения электронных магнитометров, которые являются недорогими и компактными устройствами с точностью измерения, сопоставимой с точностью индукционных датчиков.

По перечисленным причинам была поставлена задача разработки электронного магнитного компаса (ЭМК) для авиационных применений. Были выдвинуты следующие требования: диапазон измерения магнитного поля  $\pm 1600$  мкТл, чувствительность магнитометра – не менее единицы младшего разряда на 0,6 нТл, три оси чувствительности, температурный диапазон от  $-40^{\circ}\text{C}$  до  $+80^{\circ}\text{C}$ , интерфейс передачи данных ARINC-429, напряжение питания модуля 27 В, потребляемая мощность не более 3 Вт, максимально возможное использование отечественной электронной базы, минимальные габариты и цена.

Современный датчик электронного магнитометра часто содержит трехосевой МЭМС акселерометр, который может быть использован для определения угловой ориентации датчика относительно плоскости горизонта, однако эта возможность существует только при отсутствии ускорений объекта.

В результате анализа современного уровня техники была разработана структурная схема ЭМК, представленная на рис.



**1. Микроконтроллер (МК) K1986BE92QI.** Необходимо отметить, что при выборе МК поиск осуществлялся, прежде всего, по продукции российского производства. МК K1986BE92QI, выпускаемый российской компанией Миландр, имеет следующие основные характеристики: 32-разрядный процессор ARM Cortex-M3 с частотой 80 МГц, ПЗУ 128 кбайт, ОЗУ 32 кбайт, периферийные модули: SPI, I2C, UART, USB, напряжение питания 3,3 В, температурный диапазон – 40 °С до +125 °С, количество выводов общего назначения 43.

**2. Датчик трехосевого магнитометра LSM303C.** Данный датчик был выбран по следующим параметрам: диапазон измерения магнитного поля, три оси чувствительности, наличие интерфейсов SPI и I2C. Требуемый диапазон измерения магнитометра был определен исходя из следующих фактов. Напряженность магнитного поля, измеряемая датчиком, складывается из двух составляющих – напряженности естественного магнитного поля Земли и напряженности искусственного магнитного поля устройства. Напряженность магнитного поля на поверхности земли лежит в диапазоне от 22 – 67 мкТл [1]. Значение напряженности магнитного поля, создаваемое современными электронными устройствами, может достигать 1000 мкТл. В связи с этим диапазон магнитометра должен быть таким, чтобы не возникло перенасыщение датчика, т.е. не менее ±1000 мкТл.

В результате анализа продукции наиболее известных производителей был выбран датчик LSM303C производства STMicroelectronics. Его основные характеристики: диапазон измерения магнитного поля ±1600 мкТл, напряжение питания 3,3 В, потребляемый ток 100 мкА, интерфейсы передачи данных SPI и I2C, температурный диапазон от –40 °С до +85 °С, программируемый генератор прерываний.

**3. Приемопередатчик HI-3585 интерфейса ARINC-429.** Данная микросхема выпускается фирмой Holt Integrated Circuits и предназначена для преобразования интерфейса SPI в авиационный интерфейс последовательных данных ARINC-429. Ее основные характеристики: напряжение питания 3,3 ±5 В, температурный диапазон от –40 °С до +125 °С. Приемник и передатчик этой микросхемы подключаются напрямую к шине ARINC-429, что позволяет создать компактное устройство сопрягаемое с большим количеством МК, поддерживающих интерфейс SPI.

**4. Супервизор питания MCP121-SOC23.** Задача этой микросхемы состоит в сбросе МК при достижении минимального порога напряжения питания. Супервизор питания MCP121-SOC23 производства Microchip имеет следующие основные характеристиками: порог срабатывания 3,1 В, импульс сброса шириной 120 мс, активно низкий тип сброса, температурный диапазон –40 °С до +85 °С.

5. Преобразователь напряжения WRB2403S-3WR2. Данный DC/DC преобразователь производства MORNSUN был выбран по следующим параметрам: входное и выходное напряжение, потребляемая мощность, габариты. Так как напряжение питания равно 27 В, был выбран преобразователь с диапазоном входного напряжения от 18 – 36 В. Его ключевые характеристики: мощность 3 Вт, выходное напряжение 3,3 В, температурный диапазон –40 °С до +85 °С, максимальный выходной ток 785 мА.

6. Преобразователь напряжения WRB2405S-3WR2. Данный преобразователь имеет диапазон входных напряжений, аналогичный WRB2403S-3WR2. Отличие в том, что он выдает напряжение питания  $\pm 5$  В. Его основные характеристики: мощность 3 Вт, температурный диапазон –40 °С до +85 °С, максимальный выходной ток  $\pm 300$  мА.

Предложенная структурная схема ЭМК на основе датчика трехосевого магнитометра LSM303С производства STMicroelectronics и 32-разрядного микроконтроллера K1986BE92QI российского производителя Миландр, позволяет получить функциональное и компактное устройство, удовлетворяющее указанным выше требованиям.

#### **Библиографический список**

1. *Браславский Д. А.* и др. *Авиационные приборы*. М.: Машиностроение, 1964. 736 с.

УДК 621.396.6

**К. Р. Афанасьева**

студент кафедры конструирования и технологий электронных и лазерных средств

**В. П. Ларин**

д-р техн. наук, проф. – научный руководитель

## ОСОБЕННОСТИ ТЕХНОЛОГИЧЕСКОГО ПРОЕКТИРОВАНИЯ 3D MID-КОНСТРУКЦИЙ

Технологическое проектирование – сложный процесс, целью которого является разработка рациональных организационно-технологических решений для изготовления качественных изделий. Для того чтобы этой цели достигнуть, при проектировании изделия, в данном случае 3D MID-конструкции [1], нужно учесть ряд особенностей процесса. Именно об особенностях конструкторских решений и технологий изготовления пойдет речь в этой статье, являющейся частью выполненного проекта.

Прежде всего, кратко рассмотрим сам способ производства 3D MID с использованием технологии прямого лазерного структурирования (LDS). Первая стадия – литье компонента под давлением из пластика, каталитически активированного ионами металла. Вторая – формирование структуры проводящих дорожек с помощью лазера. Третья – осаждение слоя химической меди на местах лазерной активации и оловянирование контактных площадок. И четвертая стадия – установка компонентов.

При выполнении этих этапов необходимо учитывать ряд требований при разработке конструкции и технологий. Рекомендуется использование проектных инструментов, находящихся на стыке электрических и механических САПР, которые обеспечат требуемую гибкость и эффективность (модуль Nextra MID для FlowCAD). При проектировании формы для литья 3D MID корпуса литник не должен находиться в критической зоне (зоне расположения проводников, контактных площадок). Также нужно следить за тем, чтобы на критических поверхностях не было линий смыкания потоков, поскольку это может оказаться критичным при механических и термических нагрузках. По возможности стоит выбрать для литья машины с горячеканальной системой и непосредственным впрыском в форму или с литниковой системой с шиберным клапаном, что позволит избежать паразитной металлизации в зоне литника.

На следующем этапе необходимо сформировать структуру системы проводящих дорожек с помощью лазера. Лазерному формированию печатного рисунка предшествует процесс каталитической активации поверхности пластика. В настоящее время существует несколько технологий прямого лазерного формирования печатного рисунка, и предпочтение постепенно перешло к беспалладиевым методам активации поверхности полимерных материалов. Среди этих методов представляет интерес обработка полимерной основы водными растворами, содержащими восстанавливаемые соли неблагородных металлов: меди, никеля, железа, кобальта с последующим их химическим восстановлением [2]. В результате на поверхности полимера образуется слой свободных ядер металлов, каталитически активный в процессе химического осаждения покрытий.

Зазоры (пробельные места) между сформированными лазером элементами печатного рисунка освобождаются от пленки активатора испарением с помощью ультрафиолетового лазера.



Рабочая зона лазерной обработки ограничена и не может быть изменена (рис.1).

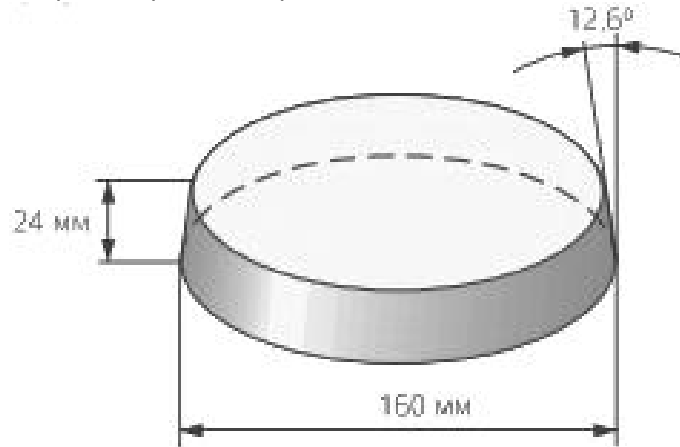


Рис. 1. Размер рабочей зоны лазера (в модели LPKF MicroLine-3)

Ширину проводников и расстояние между ними рекомендуется брать около 150 мкм и больше (рис. 2).

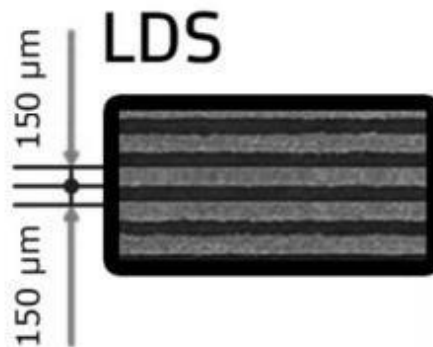


Рис. 2. Стандартные размеры LDS

По экономическим соображениям толщина химически наносимого слоя меди по рисунку лазера без последующего гальванического усиления ограничивается 8 мкм. Если требуется гальваническое усиление для достижения требуемой по токовым характеристикам толщины проводника, то потребуется подсоединение всех цепей 3D MID к общей контактной площадке. При проектировании сквозных отверстий соблюдаются требования к конусным и двухконусным отверстиям. Стоит обратить внимание, что угол конусности не должен превышать максимальный угол падения луча лазера даже по краям рабочей зоны (рис. 3).

Лазерное структурирование, использованное в проекте, обладает определенными преимуществами по сравнению с методами химической и химико-гальванической металлизации:

- нет необходимости в фотолитографии;
- сокращены технологические этапы;
- возможно создание областей с тончайшими структурами (ширина линий и расстояния между ними < 50мкм);
- значительное уменьшение ширины проводниковых линий и расстояний между ними
- высокий эффект дает использование для небольших участков с высокой плотностью интеграции, т.е. большие участки обрабатываются традиционным фотолитографическим способом (класс точности печатного рисунка 3-4), а малые, с высокой плотностью интеграции (например, выше 5-го класса точности), – посредством лазерного структурирования;
- совместимость со стандартными процессами металлического покрытия и процессом травления.

– прямое структурирование на основе данных программного проектирования (CAD)

Монтажные соединители формируются запрессовкой круглых или квадратных в сечении штыревых контактов (рис. 3) в тело корпуса над металлизированными отверстиями с последующей их пайкой в процессе монтажа MID.

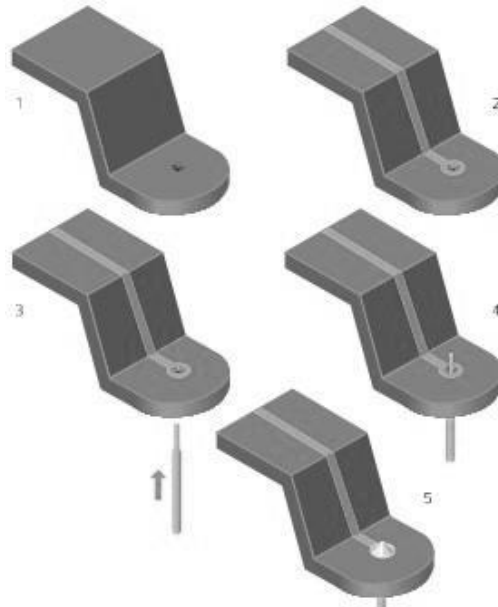


Рис. 3. Контакт-деталь зажимного соединителя

Важным критерием является закругление углов (рис. 4) проводников. Рекомендуемый радиус закругления – более 150 мкм.



Рис. 4. Закругление углов

Размещать проводящие дорожки нужно таким образом, чтобы они не упирались в боковые стенки (рис. 5) [2,3]. Это снизит риск образования отложений и появления паразитной металлизации на стенках. Расстояние от дорожки до стенки более 150-ти микрон при угле стенки 45°, более 250 мкм – при 70°.

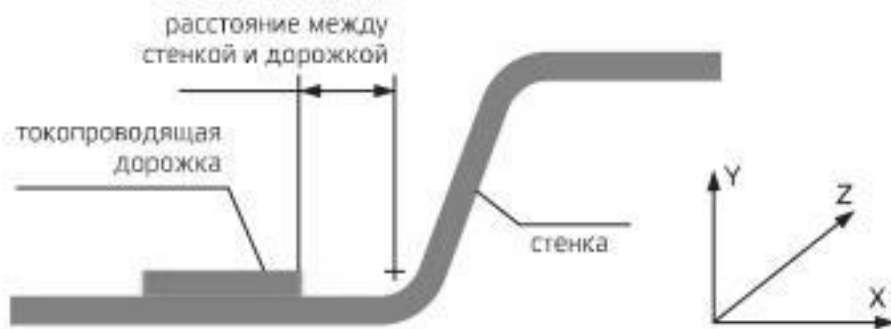


Рис. 5. Расстояние от дорожки до стенки

Максимально допустимый угол падения луча лазера на поверхность корпуса для гарантированного обеспечения активации не должен превышать  $70^\circ$  (рис. 6).



Рис. 6. Угол падения лазерного луча

Необходима вентиляция, чтобы отводить пузырьки водорода, образующиеся в процессе металлизации и закупоривающие отверстия при металлизации. Следует избегать поворотов проводников (дорожек) с углами для уменьшения нагрузок, возникающих при работе MID.

Разница коэффициентов теплового расширения пластмассы и токопроводящих дорожек у 3D MID выше, чем у традиционных печатных плат. Это может привести к увеличению выхода изделий из строя при переменных тепловых нагрузках. Лучше не использовать SMD-резисторы и конденсаторы с типоразмером больше 1206.

Итак, учет всех рассмотренных особенностей выполнения отдельных этапов процесса изготовления 3D MID, позволит избежать ошибок при разработке изделий и выборе технологий изготовления.

### Библиографический список

1. Афанасьева К.Р. Анализ направлений развития MID-технологий // Сборник докладов 68 международной студенческой конференции ГУАП. 2015. С. 91.
2. Бонапартов С.В. Технология 3D-MID. Процесс, возможности, области применения / ЗАО «НИИИТ» / ГК Остек, 2014. С. 21-25.
3. Руководство по разработке трехмерных систем на пластике // Решения для трехмерных схем на пластике / ГК Остек, 2013/14. С. 5-15.

УДК 658.512

**О.В. Баранова**

магистрант кафедры конструирования и технологий электронных и лазерных средств

**Р.И. Сольнищев**

д-р техн.наук, профессор – научный руководитель

### РДМ-СИСТЕМА ПРОЦЕССА ПРОИЗВОДСТВА ЗАМКНУТОЙ СИСТЕМЫ УПРАВЛЕНИЯ НЕЙТРАЛИЗАЦИЕЙ ВЫХЛОПНЫХ ГАЗОВ АВТОМОБИЛЯ

В данной статье будет рассмотрен метод нейтрализации отработавших газов автомобиля на основе замкнутой системы управления нейтрализацией выхлопных газов автомобиля (ЗСУНВГА) с применением катализатора с подогревом фирмы Emites. Также будет рассмотрена автоматизированная система технологической подготовки производства, задачи АСТПП, РДМ-систем для задачи производства ЗСУНВГА.

Процесс катализа отработавших газов автомобиля несовершенен, часть вредных веществ не сгорают и выбрасываются в атмосферу, загрязняя ее. Это происходит из-за недостаточно высоких температур в катализаторе автомобиля некоторое время после включения двигателя. Так же после долговременной работы автомобиля из-за высоких температур в потоке газовой смеси соты катализатора плавятся, забивая катализатор и выводя его из строя.

Были проведены исследования уменьшения отработавших газов автомобиля с использованием катализатора со встроенным подогревателем, произведенным фирмой Emitec. Результаты этих исследований показывают, что при электрическом подогреве катализатора потери топлива меньше, чем при подогреве катализатора отработавшими газами (рис. 1). Что дает положительный результат также в части экономии топлива [1].

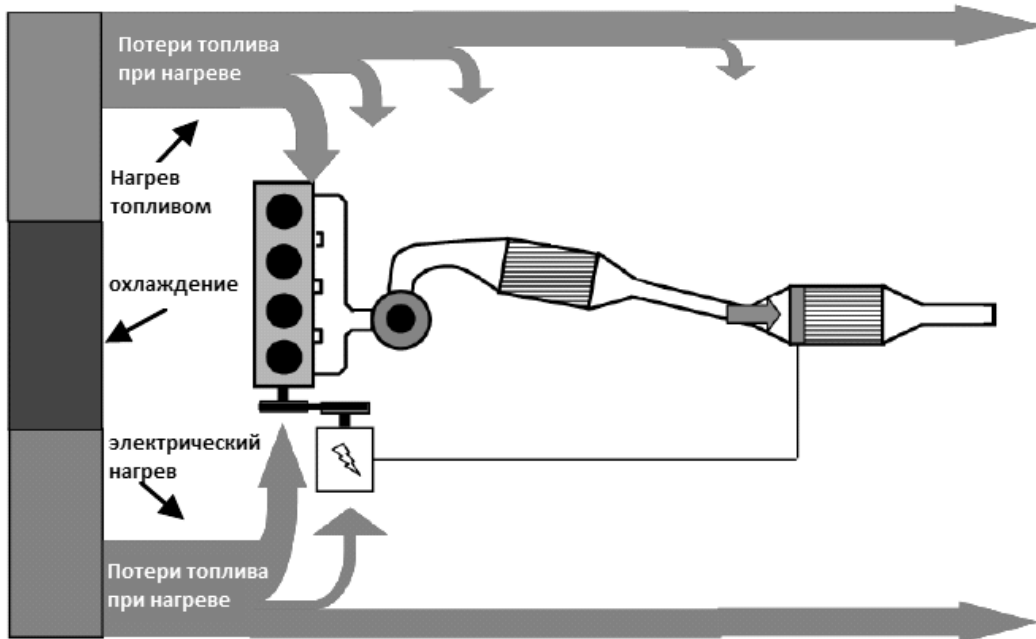


Рис. 1. Результаты исследований сокращения количества отработавших газов на основе подогрева катализатора

Предлагаемый метод нейтрализации выхлопных газов автомобиля основан на применении замкнутой системы контроля над температурой в катализаторе. В качестве контроля используется управление подогревом катализатора, что увеличит скорость протекания химической реакции по нейтрализации вредных веществ и тем самым снизит количество неочищенных газов.

Работа такой системы может осуществляться как локально проводной или беспроводной связью, так и с использованием GPRS для передачи данных о количестве загрязняющих веществ (ЗВ) в ДатаЦентр и ГЛОНАСС для обнаружения локализации автомобиля.

Метод системы управления нейтрализации выхлопных газов автомобиля (ЗСУНВГА) имеет 3 варианта:

- простейшая структура, разомкнутая – только контроль;
- локальная структура, замкнутая – контроль+нейтрализация ВВГ;
- глобальная структура, замкнутая – контроль водителем+нейтрализация ВВГ+контроль диспетчером ПРИРОДНАДЗОРА

Первый вариант имеет катализатор с подогревателем, контроллер для управления нагревателем, датчики измерения ВВГА. Принцип действия такой системы заключается в том, что датчики измеряют количество CO, CH, NOx и передают информацию на индикаторы для визуализации водителя.

Второй вариант, кроме перечисленных в первом варианте элементов, имеет индикатор ВВГА для визуализации водителя. Информация о состоянии выхлопов поступает на индикатор

на приборной панели автомобиля и на контроллер, который изменяет температуру в катализаторе.

Третий вариант включает в себя катализатор с подогревом, контроллер для управления нагревателем, датчики измерения ВВГА, индикаторы ВВГА на приборной панели, модуль ГЛОНАСС, модуль GSPR. Информация с датчиков, стоящих после катализатора, поступают на контроллер, на индикаторы ВВГА и через GPRS передаются в ДатаЦентр. В ДатаЦентре происходит сравнение текущих результатов и норм ПДК загрязняющих веществ, и, в случае, если ВВГА превышают норму ПДК, на контроллер дается команда диспетчером о повышении температуры в катализаторе.

Для поддержания оптимальной температуры в катализаторе катализатор с теплонагревателем конструктивно перемещается за глушитель в конец выхлопной системы [2].

Техническим результатом, достигаемым при осуществлении ЗСУНВГА (рис.2) является повышение эффективности очистки отработавших газов и надежности (ресурса) катализатора.

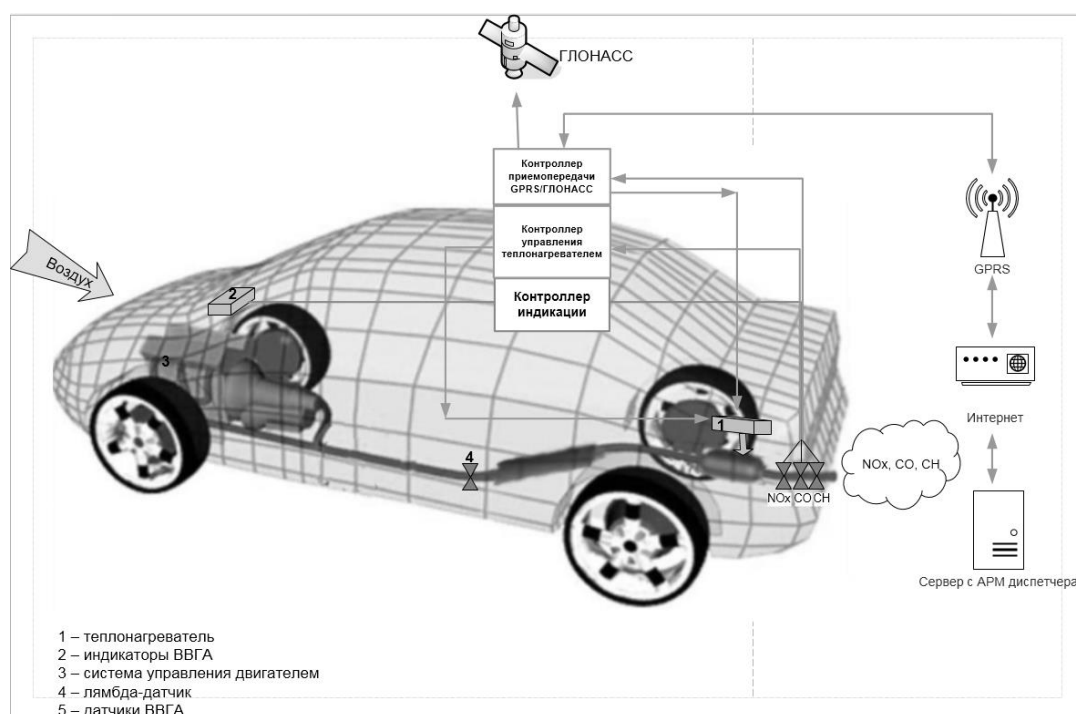


Рис. 2. Схема ЗСУНВГА

При производстве такой системы необходимо определить при помощи каких методов и средств, способов организации производства должна изготавливаться система, определить его себестоимость и эффективность. Эта стадия является технологической подготовкой производства (ТПП). Так как система будет внедряться в разные марки автомобилей, необходимо использование автоматизированной системы технологической подготовки производства (АСТПП) для уменьшения затрачиваемого времени на подбор деталей и финансовую оценку производства. Возрастающие требования научно-технического прогресса предполагают высокую гибкость процесса подготовки с целью более быстрой адаптации к новым типам производства.

Применение средств обработки данных в ТПП дало возможность решить организационные проблемы, такие как, например, управление производственным планированием. Многочисленные разработки систем ТПП как ориентированных на конкретное производство так и не ориентированных на него, позволяют решать различные задачи ТПП. Существенным преимуществом АСТПП является выполнение рутинных процессов и подготовка информации с помощью средств электронной обработки данных. Специалист, работающий с АСТПП, избавится от монотонной, нетворческой работы. Кроме того благодаря большому быстродействию средств элек-

тронной обработки данных появляется возможность исследования различных альтернативных решений и реализации процессов оптимизации [3].

Важная составляющая АСТПП – база знаний, которая содержит в себе нормативно-правовую базу, стандарты и инструкции по которым производится изделие, набор элементов, совместимых друг с другом, схемы и др. Информационная поддержка строится на основе применения так называемых CALS-технологий (непрерывная информационная поддержка жизненного цикла продукта). В качестве одного из базовых инструментов реализации CALS-технологий выступают системы класса PLM (Product Lifecycle Management).

Основными компонентами PLM-системы являются:

- PDM-система (PDM – Product Data Management). Система управления данными об изделии, является основой PLM, предназначена для хранения и управления данными;
- CAD-система (CAD – Computer Aided Design). Проектирование изделий;
- CAE-система (CAE – Computer Aided Engineering). Инженерные расчеты;
- CAPP-система (CAPP – Computer Aided Production Planning). Разработка техпроцессов;
- CAM-система (CAM – Computer Aided Manufacturing). Разработка управляющих программ для станков с ЧПУ;
- MPM-система (MPM – Manufacturing Process Management). Моделирование и анализ производства изделия.

Таким образом, под PLM-системой мы понимаем совокупность программных продуктов (в том числе, от разных поставщиков). PLM-система должна решать задачи как создания инженерных данных (средствами CAD/CAE/CAPP/CAM/MPM-систем), так и задачи управления инженерными данными (средствами PDM-системы).

Как видно из рис. 3, ключевую роль в PLM играет PDM-система, задачей которой является предоставление нужных данных в нужное время в нужной форме в соответствии с правами доступа.

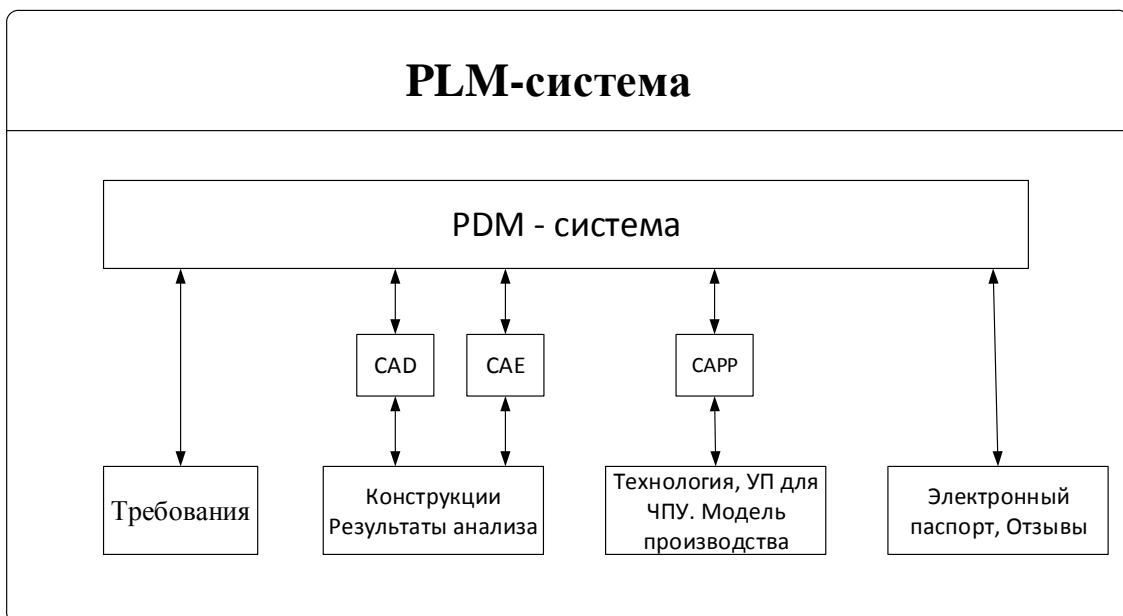


Рис.3. Структура PLM-системы

Наиболее типичные задачи, решаемые при помощи PDM-систем:

- Электронный архив документации (конструкторской, технологической, организационно-распорядительной, проектной, нормативно-технической);
- Электронный документооборот (согласование данных и документов, контроль исполнения);

- Управление разработкой данных и документации (совместная работа в рабочей группе, управление составами и конфигурацией изделий);
  - Компьютерная система менеджмента качества;
  - Электронные справочники (материалы, ПКИ, стандартные изделия и т. д.).
- Преимущества, связанные с управлением инженерными данными:
- Повышение производительности труда сотрудников;
  - Сокращение сроков подготовки производства;
  - Повышение качества продукции и степени удовлетворенности клиентов;
  - Снижение стоимостных издержек;
  - Сопровождение интеллектуальной собственности предприятия;
  - Обеспечение данными АСУП/ERP-системы;
  - Соответствие предприятия требованиям ISO 9000.

Для информационной поддержки АСТПП ЗСУНВГА был проведен анализ технических требований к компонентам автомобиля, собраны нормативно-правовая база, конструкторская и техническая документация изготовления изделия, компоненты, отвечающие требованиям к автомобильной промышленности, чертежи и данные для изготовления катализатора с нагревателем, электрические схемы и результаты моделирования системы. Это позволит сократить время на выбор деталей, поиск справочной информации в данной области, накопить базу технологий для разных марок автомобилей, решить проблему дублирования и неактуальных данных.

#### **Библиографический список**

1. *Manuel Presti, Lorenzo Pace. An Alternative Way to Reduce Fuel Consumption During Cold Start: the Electrically Heated Catalyst// SAE International. 2011. № 11*
2. Пат. 2012145342/06 РФ МПК-F01N 3/28. Каталитический нейтрализатор вредных выбросов автомобиля в атмосферу
3. *Сольнищев Р.И. Автоматизация предприятия систем автоматического управления – М.:Высш.шк., 1991. 335с.*

**УДК 621.382.049.77.002**

**П.В. Белолипецкая**

студентка кафедры конструирования и технологий электронных и лазерных средств

**В. П. Ларин**

проф., д-р. техн. наук, проф. – научный руководитель

### **ОБЕСПЕЧЕНИЕ ПАРАМЕТРОВ ТЕХНОЛОГИЧЕСКОЙ СРЕДЫ ПРИ ИЗГОТОВЛЕНИИ ЭЛЕКТРОННОЙ АППАРАТУРЫ**

Цель работы: проведение анализа задач обеспечения параметров чистых помещений и принципов управления параметрами при реализации технологического процесса изготовления полупроводниковых пластин.

Для изготовления различной электронной аппаратуры требуется выполнение необходимых условий, а именно, то состояние технологической среды, при которой процесс изготовления будет проходить без повреждения изделий. Для поддержания требуемых условий технологической среды на предприятиях создаются специальные чистые комнаты (помещения), где выполняются все необходимые условия.

Технологические особенности изготовления ИМС

Все полупроводниковые приборы имеют четыре характерных признака:

– наличие активной структуры в виде р-п-перехода, контакта металл – полупроводник (барьер Шоттки) или металл – диэлектрик – полупроводник (МДП-структура), обеспечивающей выполнение прибором его основной рабочей функции;

– наличие омических контактов в виде сильно легированных  $p^+$ - или  $n^+$ -областей, имеющих низкое контактное сопротивление для подвода напряжения к активным областям прибора и их электрической связи с внешними цепями;

– монолитный характер конструкции прибора, делающий невозможным разделение его на отдельные элементы; например, эмиттер, база и коллектор транзистора являются неразделимыми областями одного и того же полупроводникового кристалла;

– интегрально-групповой характер технологии, порожденный монолитностью конструкции прибора и позволяющий одновременное (интегральное) проведение на всей пластине определенной группы технологических воздействий, например, создание области эмиттера сразу для всех приборов на пластине.

Все методы полупроводниковой технологии по результату воздействия на материал можно разделить на четыре группы [1]:

I группа – методы удаления материала с использованием не только обычных жидкостных травителей (кислот и щелочей), но и «сухого травления», основанного на применении реактивных парогоазовых смесей (газовое травление) и низкотемпературной плазмы (ионно-плазменное травление).

II группа – методы нанесения материала на поверхность полупроводниковой пластины, такие как:

1. Создание металлических покрытий методами термовакuumного или электронно-лучевого испарения и катодного или ионно-плазменного распыления металлов, выполняющих три функции: а) контактного сплава для омических контактов, б) высокоомного сплава для резисторов, в) низкоомного сплава для электрического соединения элементов схемы.

2. Создание диэлектрических (окисных) слоев различными способами (например, окислением поверхности полупроводника), выполняющих следующие функции: а) активная функция подзатворного диэлектрика в МДП-приборах и изолятора элементов схемы; б) пассивная функция защиты поверхности приборов от внешних воздействий; в) технологическая функция маскирующего покрытия, селективно защищающего поверхность полупроводника от технологических воздействий, например, при локальном формировании контактных площадок к элементам ИМС.

3. Создание монокристаллических слоев методами жидкофазной, газофазной и молекулярно-пучковой эпитаксии O.

III группа – методы изменения свойств материала при диффузионном легировании и ионной имплантации.

IV группа – методы локальной микрообработки, называемые методами литографии, такие как фотолитография (ФЛ), электронная литография (ЭЛ), ионная литография (ИЛ) и рентгеновская литография (РЛ). [4]

Сущность методов литографии состоит в формировании на поверхности пластины элементов прибора или рисунка схемы с помощью специального химически стойкого полимерного материала – резиста, чувствительного к облучению ультрафиолетом (фоторезист при ФЛ), пучком ускоренных электронов (электронорезист при ЭЛ), пучком ускоренных ионов (ионорезист при ИЛ) или рентгеновским излучением (рентгенорезист при РЛ).

При эпитаксии происходит ориентированное наращивание на монокристаллической подложке слоя, продолжающего при своем росте кристаллографическую ориентацию подложки. Экспонирование производится через специальную маску (фотошаблон) с изображением элементов прибора или рисунка схемы, селективно пропускающим облучающий поток. В результате облучения полимерные цепи в освещенных участках резиста либо разрушаются (позитивные фоторезисты), либо задубливаются (негативные фоторезисты), формируя скрытое изображение. Любые



посторонние аэрозольные включения из технологической среды или отклонения от климатических нормативов недопустимы.

Основу современной полупроводниковой технологии составляет **планарный процесс**, под которым понимается совокупность технологических воздействий на материал, осуществляемых с одной стороны исходной пластины полупроводника. Технологический цикл изготовления полупроводниковых приборов и интегральных микросхем включает:

- процесс механической и химической обработки поверхности пластины для подготовки ее в качестве подложки к эпитаксиальному наращиванию монокристаллической пленки или к планарному процессу изготовления приборов;
- эпитаксиальный процесс выращивания на поверхности подложки полупроводниковых слоев с нужными электрофизическими свойствами;
- процесс формирования маскирующего покрытия для последующего проведения локальных технологических воздействий на поверхность полупроводника.

Все рассмотренные процессы, полупроводниковые и пленочные структуры крайне чувствительны к условиям технологической среды. Отсутствие незапланированных примесей, строгое поддержание климатических параметров обеспечивается так называемыми «чистыми» помещениями.

Чистое помещение (cleanroom). Помещение, в котором контролируется концентрация взвешенных в воздухе частиц, чтобы свести к минимуму поступление, выделение и удержание частиц внутри помещения, и позволяющее, по мере необходимости, контролировать другие параметры, например, температуру, влажность и давление. Для некоторых процессов осуществляется контроль и регулирование определенных видов параметров технологической среды непосредственно на технологическом участке. Перед подачей в помещения воздух проходит специальную систему фильтрации, а производственные помещения оборудуются системами кондиционирования. Давление воздуха внутри зданий должно несколько превышать атмосферное для уменьшения самопроизвольного проникновения наружного воздуха (минуя фильтр и вентилятор). Чистая комната имеет рабочее помещение, где размещается технологическое оборудование и выполняются операции; гардеробные помещения для подготовки обслуживающего персонала к работе; переходные и обдувочные шлюзы; помещения для обработки приточного воздуха. В чистых помещениях обеспечиваются не только определенные температура, влажность, запыленность, но и определенные скорость и направление перемещения воздуха, способствующие эффективному удалению пыли, образующейся при работе оборудования и движениях персонала [3].

Обеспечение необходимых требований по микроклимату и чистоте воздушной среды связано со значительными материальными затратами, поэтому целесообразно отдельные операции, к которым предъявляются единые требования, группировать в общем помещении. Установлены для различных операций микроэлектроники классы микроклимата и чистоты производственных помещений. К типовым классифицированным операциям относятся химическая обработка пластин и подложек, окисление пластин, диффузия примесей; скрабирование и разделение пластин, присоединение выводов; герметизация и контроль герметичности, изготовление толстопленочных ИМС. Класс помещения у всех этих операций третий, класс рабочего объема по запыленности – пятый.

Из-за влажности воздуха производственных помещений происходит адсорбция паров воды поверхностью пластин и элементами рабочих поверхностей технологических установок, что при нагреве может приводить к образованию нежелательных окислов. Следует отметить, что адсорбция паров воды поверхностью изделий происходит более активно, чем кислорода воздуха, вследствие малых размеров молекул воды, поэтому влажность в производственных помещениях должна быть минимальной (но не ниже 30%, установленных санитарными нормами). По температурно-влажностным параметрам соответствующим отраслевым стандартом устанавливается

три класса производственных помещений. Параметры технологической среды производственных помещений установлены ОСТ 4.091.172—81 и должны строго поддерживаться.

С точки зрения производственной гигиены воздушная среда производственного помещения характеризуется запыленностью. Присутствие в воздухе механических частиц – аэрозолей является одной из причин появления проколов в защитном слое окисла и выхода из строя элементов и ИМС в целом. Частицы пыли, осевшие на поверхности готовых структур, могут привести к пробоям и коротким замыканиям при электрических испытаниях ИМС или ее эксплуатации. Достичь высокой степени обеспыленности проще в ограниченных, локальных рабочих объемах – боксах или скафандрах. Внутри бокса может быть установлено то или иное технологическое оборудование небольших размеров (центрифуга для нанесения фотослоя, установка совмещения и экспонирования, установка для микросварки, установки межоперационного контроля, ванны для химической обработки и т. д.). Такой бокс может служить для перегрузки пластин из герметичной цеховой тары. Оператор сообщается с рабочим объемом через лицевой проем при поднятой шторке. Достоинством боксов является возможность создать в них необходимую для выполнения операции среду (воздух высокой степени осушенности, заполнение инертным газом и т. д.). [2]

К технологическим средам относятся прежде всего технологические газы. Технологические газы в производстве ИМС используют в разных целях. *Защитные газы* (азот, аргон, гелий) применяют для исключения процессов окисления и коррозии на операциях с высокой температурой (пайка, сварка, герметизация, продувка реакторов эпитаксиальных и диффузионно-окислительных установок, транспортировка газов-реагентов и т. п.). Газы-реагенты используются в качестве диффузантов, окислителей, травителей, восстановителей и т. д. Такие газы, как аргон, азот, кислород, могут применяться в качестве плазмообразующих в процессах ионно-плазменного распыления, плазмохимической обработки, вакуум-плазменного травления.

В качестве примера рассмотрим решение задачи обеспечения требуемых условий технологической среды, выполненное при проектировании технологической операции изготовления полупроводниковой пластины.

1 этап. Выбираем задачи, подлежащие решению для данной технологической операции:

Выбор такой конструкции чистых помещений, которая обеспечивает максимальную защиту продукции. Чаще всего в чистых помещениях для микроэлектроники используется однонаправленный (ламинарный) поток нисходящего воздуха. Для обеспечения однонаправленности потока, как правило, применяются перфорированные фальшполы.

Обеспечение стабильности параметров среды чистых помещений, узкого диапазона поддержания влажности и температуры.

Использование оборудования для очистки воды, применяемой в процессе обработки пластин.

Разработка системы распределения технологических жидкостей, которая позволяет предотвратить перекрестные загрязнения.

Разработка системы распределения газов, которая должна быть герметичной и не выделять загрязнений.

Обеспечение возможности поддерживать необходимые значения температуры и влажности воздуха в чистом помещении.

Применение высокотехнологичных систем управления и контроля: датчиков для контроля содержания токсичных и опасных материалов, систем мониторинга параметров технологической среды и др.

Создание «гибкого» производства, которое позволит вносить изменения в систему чистых помещений без существенного ущерба для производственных процессов и с минимальными затратами [4].

2 этап. Выбираем класс чистого помещения.

Общепринятая классификация делит чистые помещения на шесть основных классов: класс 1, 10, 100, 1000, 10 000 и 100 000.

Каждый класс определяет максимально допустимое количество частиц в заданном объеме воздуха. Чистота помещения по взвешенным в воздухе частицам обозначается классификационным числом  $N$ .

Максимально допустимая концентрация частиц  $C_n$ , частиц/м<sup>3</sup>, с размерами, равными или большими заданного размера  $D$ , для данного класса чистоты определяется по формуле:

$$C_n = 10^N \left( \frac{0,1}{D} \right)^{2,08},$$

где  $N$  – классификационное число ИСО, которое не должно превышать значения 9;  $D$  – заданный размер частиц, мкм. Промежуточные числа классификации ИСО могут быть определены с наименьшим допустимым приращением, равным 0,1. 0,1 константа, мкм;  $C_n$  (концентрация частиц) округляется до целого числа, при этом используется не более трех значащих цифр [5].

3 этап. Выбираем диапазон, систему контроля и регулирования климатических параметров помещения.

Для контроля процесса изготовления необходимо знать и определять влажность и температуру в помещении. Это требуется не только на этапе изготовления плат, но и для комфортного нахождения сотрудников на рабочем месте. Температуру и относительную влажность воздуха измеряют разными средствами измерений температуры и влажности (в том числе психрометром, термометром, термографом, гигрографом):

Методика измерения температуры

Данный вид измерений выполняется при необходимости соблюдения повышенных требований к точности поддержания температурного режима. Рабочую зону следует разделить на секторы равной площади. Для измерений могут использоваться: а) термометры; б) приборы с изменяющимся сопротивлением от температуры; в) термисторы. Разрешающая способность датчика должна быть не менее 1/5 абсолютной величины разности между номинальным значением температуры и наиболее удаленным значением температуры от номинального значения. Датчик должен иметь действующий сертификат калибровки.

Измерение влажности. Контроль влажности выполняется с целью проверки способности системы вентиляции и кондиционирования воздуха поддерживать значения влажности воздуха (выраженной в виде относительной влажности или точке росы). Измерение влажности следует выполнять, как минимум, в одной точке в каждой зоне с заданными требованиями к влажности. Наиболее широко используются датчики влажности с тонкой диэлектрической пленкой; и датчики точки росы.

Воздух рабочей зоны должен соответствовать ГОСТ 12.1.005.

Вода, применяемая для промывки печатных плат в процессе их изготовления, должна соответствовать категории 2 по ГОСТ 9.314. Для приготовления и корректировки растворов химической и гальванической металлизации, а также для промывки готовых печатных мат перед контролем электрических параметров следует применять дистиллированную или деионизованную воду. [6]

И в заключение хотелось бы сказать, что грязный воздух в помещении, где находится оборудование, ведет к сокращению интервалов сервисного обслуживания, более частым поломкам и отказам, ускоренному загрязнению фильтров в оборудовании (их немало и они не дешевы), увеличению процента брака, сокращению срока жизни плат, а иногда и к их поломке. Чтобы избежать выше перечисленных проблем, как раз и необходимо работа с использованием чистой комнаты. Чистое помещение способствует контролю влажности, температуры и давления в технологической среде помещения, отсутствию пыли и других различных загрязнений. Важно соблюдать правильную конструкцию чистой комнаты.

**Библиографический список**

1. Ханке Х. И., Фабиан Х. Технология производства радиоэлектронной аппаратуры. М.: Энергия, 1980.
2. Тупик В.А. Технология и организация производства радиоэлектронной аппаратуры. СПб: СПбГЭТУ "ЛЭТИ". 2004.  
URL <http://dl10cg.rapidshare.de/files/31510061/4078542704/tehnologiya.i.organizaciya.proizvodstva.radioel-ektronnoj.apparatury.pdf.rar>
3. Баранов Е. А., Шевченко Е. Л., Калачик Т. С. Регенерации отработанных травильных растворов в производстве печатных плат. М.: ЦНИИ Электроника, 1981.
4. <http://sole.x2108.narod.ru/tipp1.htm>
5. Ивченко В.Г. Конструирование и технология ЭВМ. Конспект лекций. / (Таганрог: ТГРУ, Кафедра конструирования электронных средств). 2001.
6. Конструкторско-технологическое проектирование электронной аппаратуры: учебник для вузов. М.: Изд. МГТУ им. Н.Э. Баумана, 2002.

УДК 621.45.043

**Е. А. Бондарец**

студентка кафедры конструирования и технологий электронных и лазерных средств

**О. Л. Смирнов**

доц., канд.техн.наук, доц. научный руководитель

**МОДАЛЬНЫЙ АНАЛИЗ**

Модальный анализ - это режим работы в ANSYS, направленный на определение собственных форм и частот конструкции на основе результатов измерений и анализа вынужденных механических колебаний. Данное исследование поможет провести испытание пластины в соответствии со стандартами качества. Стоит задача рассчитать собственные частоты колебаний пластины в ненагруженном состоянии и провести анализ собственных частот этой же конструкции в преднагруженном состоянии. Для исследования создана модель в SolidWorks стальной пластины толщиной 0,003м (рис.1).

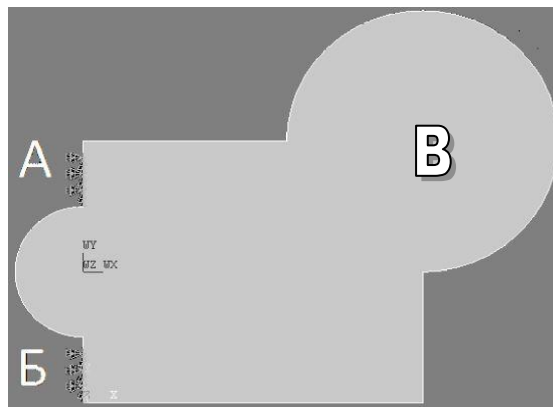


Рис. 1. Форма модели пластины

В ANSYS Workbench выбираем систему анализа (в нашем случае Modal и Static Structural). Необходимо задать материал. По параметрам создаем библиотеку модулю юнга  $E_X = 2e11$ Па, коэффициент Пуассона  $PRXY = 0,3$ ,  $DENS = 7800$  кг/м<sup>3</sup> и применяем ее в работе. После чего запускаем редактор модели, где видим сделанную нами пластину и начинаем производить над ней работу. Для этого сгенерируем сетку, которая разобьет пластину на ячейки для решения системы дифференциальных уравнений. Необходимо указать тип закрепления модели – Fixed Support во

вкладке Modal на поверхности А и Б и во вкладке Static Structural выбрать Force на поверхности В (рис. 1), так как необходимо продавить пластину с усилием 1000Н, берем обратный знак, чтобы деформация произошла в обратную сторону. Далее производим определение параметров модели.

Собственные частоты колебаний пластины в ненагруженном состоянии представлены на рис.2 – 5, а собственные частоты колебаний в преднагруженном состоянии на рис.6 – 9.

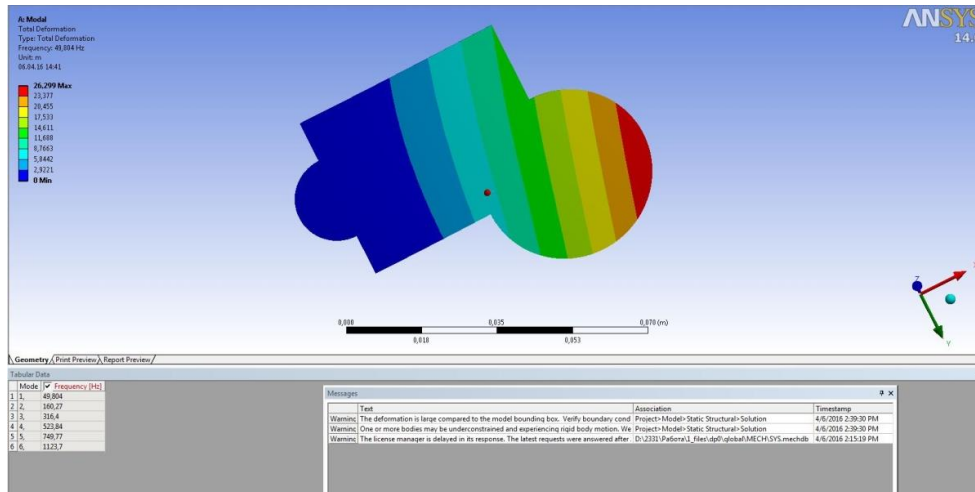


Рис. 2. Распределение перемещения по пластине при частоте до 26 Гц

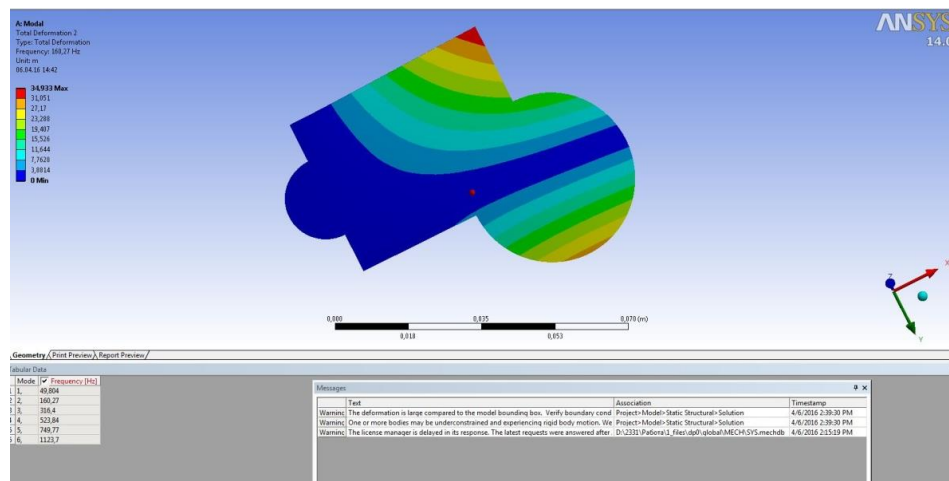


Рис. 3. Распределение перемещения по пластине при частоте до 32 Гц

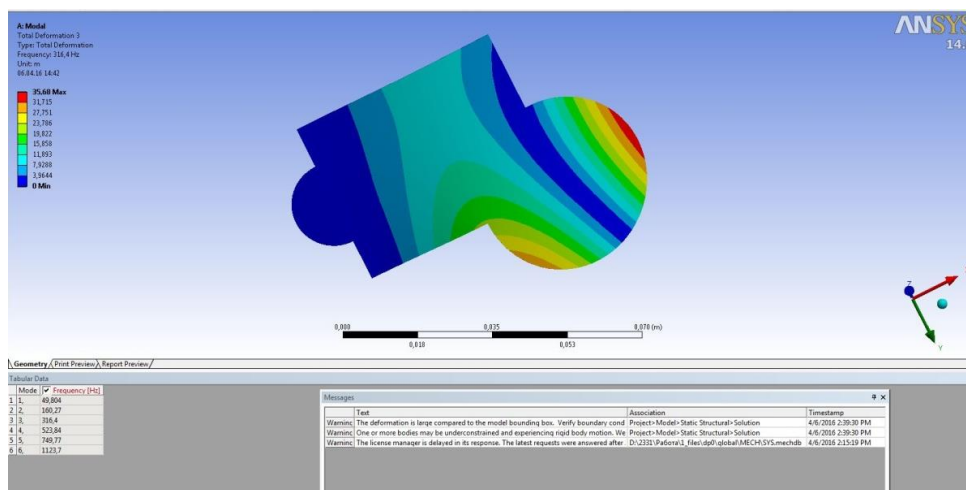


Рис. 4. Распределение перемещения по пластине при частоте до 36 Гц

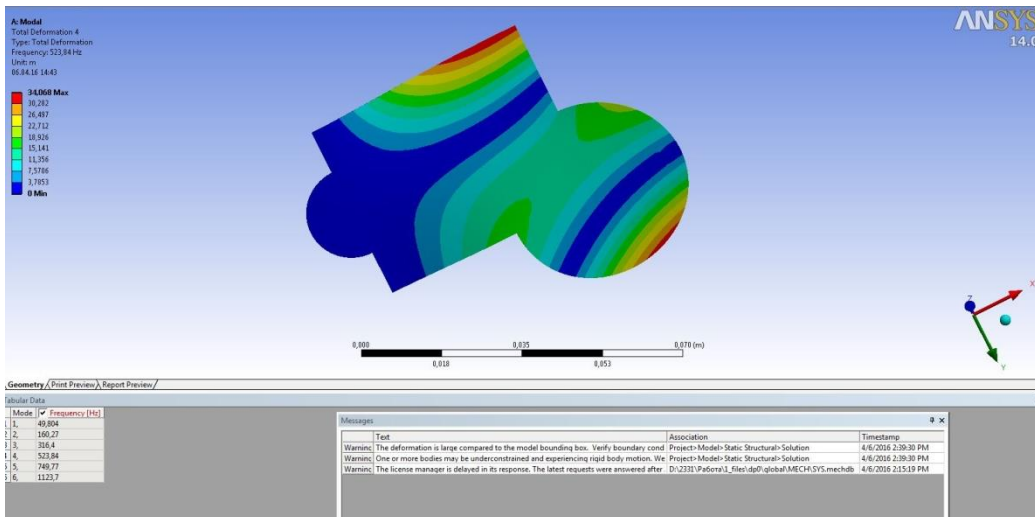


Рис. 5. Распределение перемещения по пластине при частоте до 30 Гц

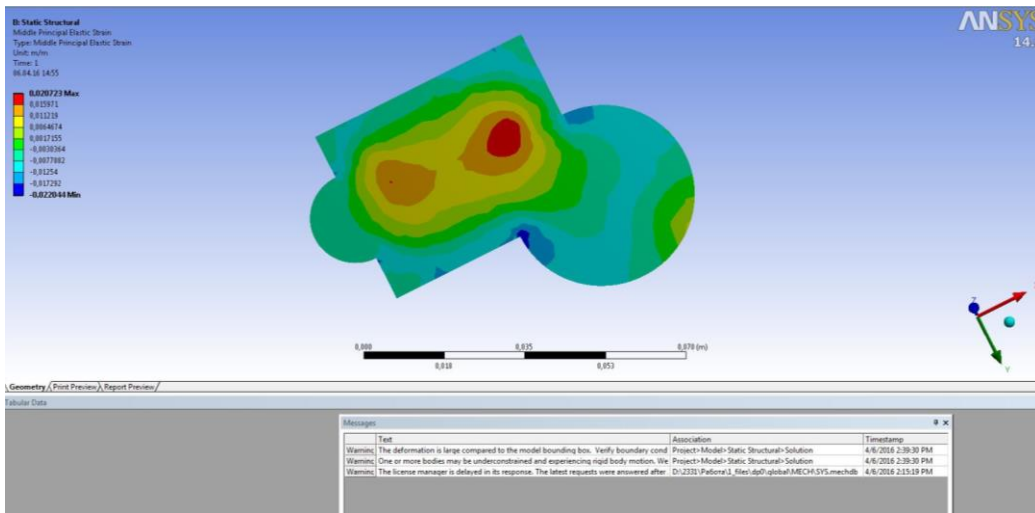


Рис. 6. Средний коэффициент упругости

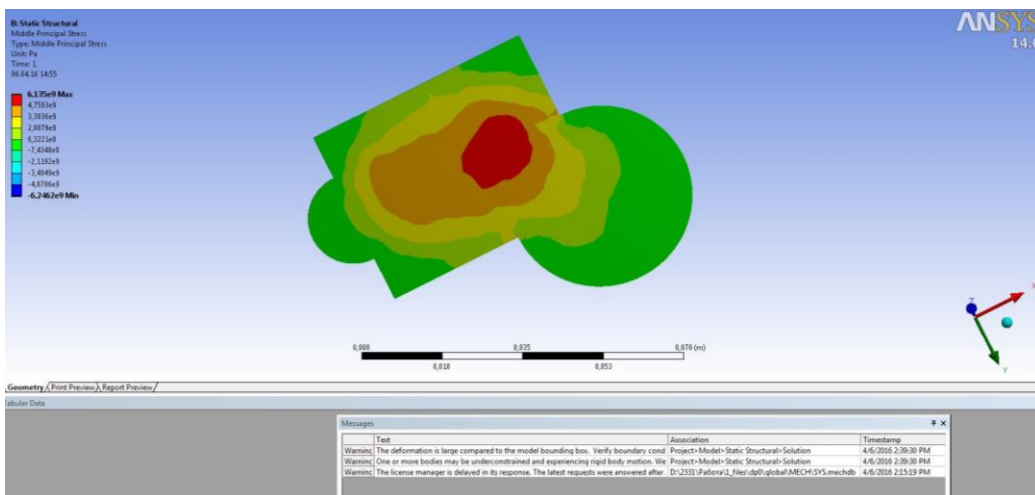


Рис. 7. Средний коэффициент напряжения

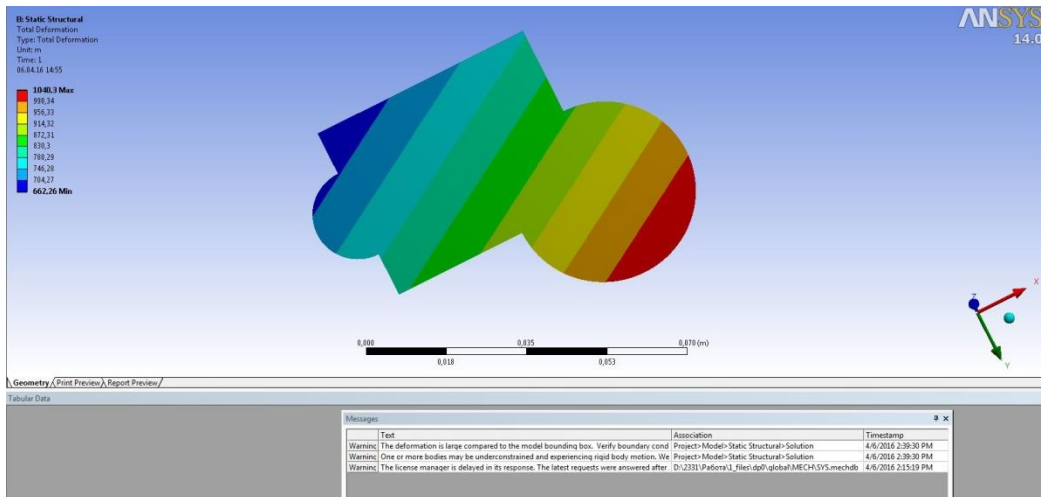


Рис. 8. Общая деформация пластины

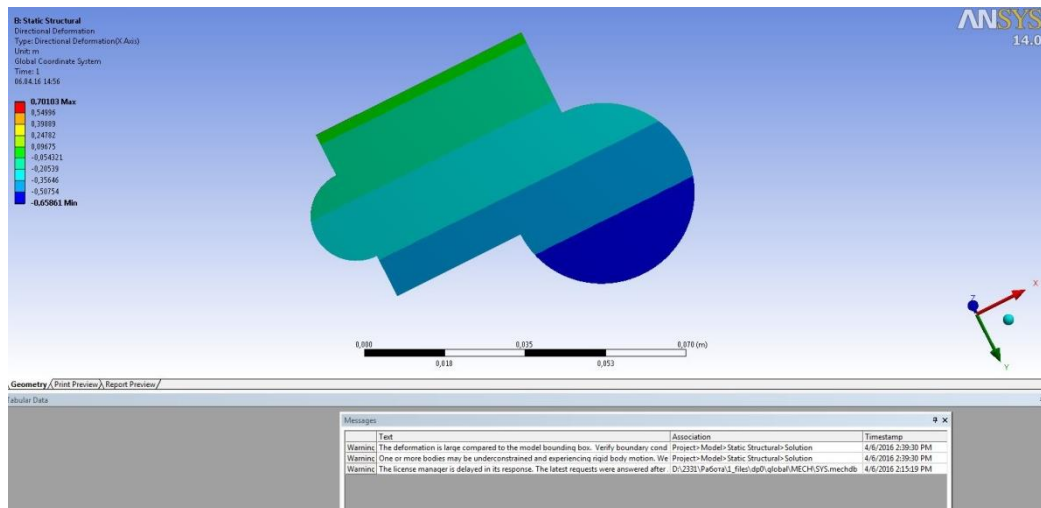


Рис. 9. Направленная деформация пластины

Из данного исследования можно сделать вывод, что с помощью программы Ansis Workbench можно произвести расчет собственных частот, что позволит выяснить, как будет вести себя пластина в резонансных режимах с точки зрения перемещений, деформации и напряжений.

### Библиографический список

1. Смирнов О.Л. Использование комплекса ANSYS при проектировании приборов и электронных приборов и электронных средств. Методические указания к лабораторным работам. ГУАП, 2014. 138 с.
2. Козлов А.А., Смирнов О.Л. Моделирование и исследование конструкций электронных узлов с использованием SOLIDWORKS и ANSYS. Методические указания к лабораторным работам. ГУАП, 2015. 59 с.

УДК 517.958:52/59

**С. А. Бондарчук**

студентка кафедры конструирования и технологий электронных и лазерных средств

**О. Л. Смирнов**

доц., канд. техн. наук, доц. научный руководитель.

## ИССЛЕДОВАНИЕ ПЛАСТИНЫ НА СЛУЧАЙНУЮ ВИБРАЦИЮ

При разработке, производстве и эксплуатации машин, станков и других объектов инженерам и техникам приходится решать вопросы по обеспечению прочности при воздействии случайной вибрации. Для решения таких проблем необходимо проводить измерение, анализ и контроль параметров вибрации.

На практике избежать вибрации не удастся никогда. Ее наличие обусловлено присутствием зазоров и поверхностных контактов отдельных деталей и механизмов, неуравновешенных элементов и деталей. Очень часто вибрация многократно увеличивается из-за присутствия резонансных явлений.

Случайная вибрация – это вибрация с непрерывным частотным спектром. Случайная вибрация очень сложное явление, кроме среднего квадратического значения, никакие параметры, которые ее характеризуют определить с достаточной точностью невозможно.

Задача исследования – анализ случайной вибрации пластины с жестким защемлением её торцевых сторон и получением полей среднеквадратичных значений перемещений, скоростей и ускорений.

Для проведения исследования случайной вибрации необходимо создать модель пластины в SolidWorks (рис.1)

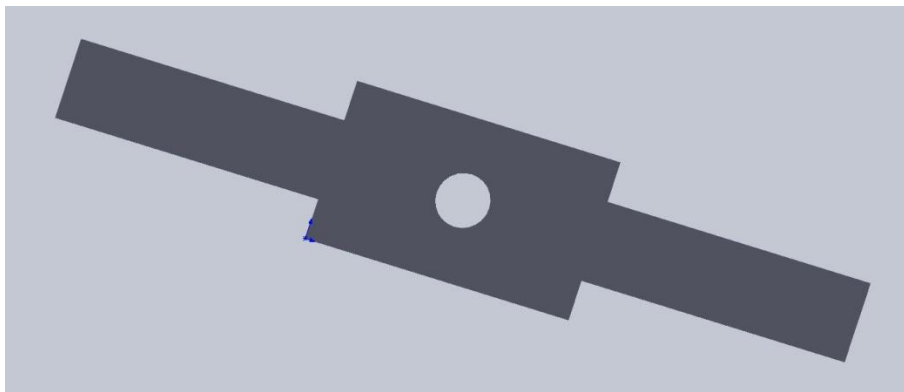


Рис.1. Форма модели пластины

В программе Mechanical для начала исследования необходимо задать параметры пластины: это материал – сталь, модуль юнга  $E_x = 2e11$  Па и коэффициент Пуассона  $\nu_{xy} = 0,27$ .

Далее нужно разбить пластину на конечные элементы, это делается для того, чтобы решить систему дифференциальных уравнений.

Первое исследование заключается в определении общей деформации при сканировании в диапазоне частот (рис.2, 3), затем – задание ускорений (рис. 4) и определение направленных перемещений – рис. 5, скорости и ускорения (рис 6, 7) результаты определения нормального напряжения (рис. 9).



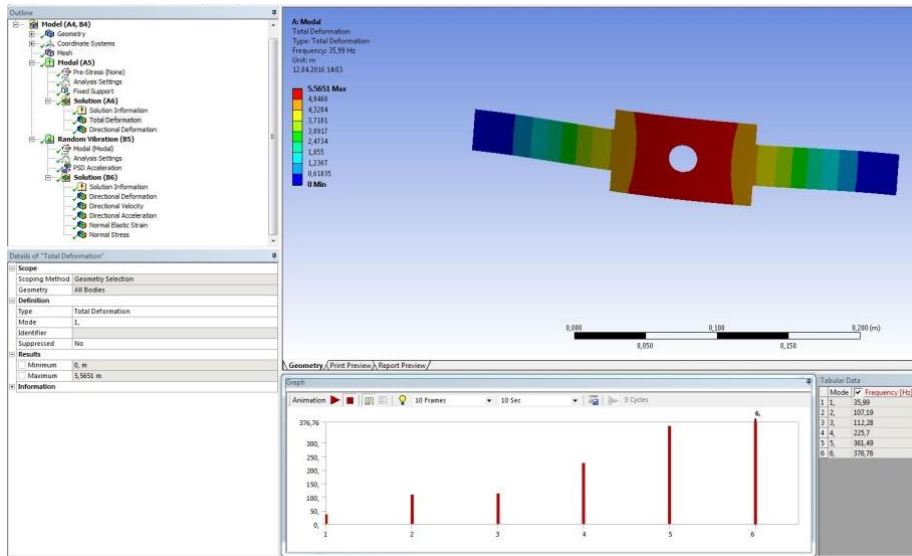


Рис. 2. Общая деформация пластины

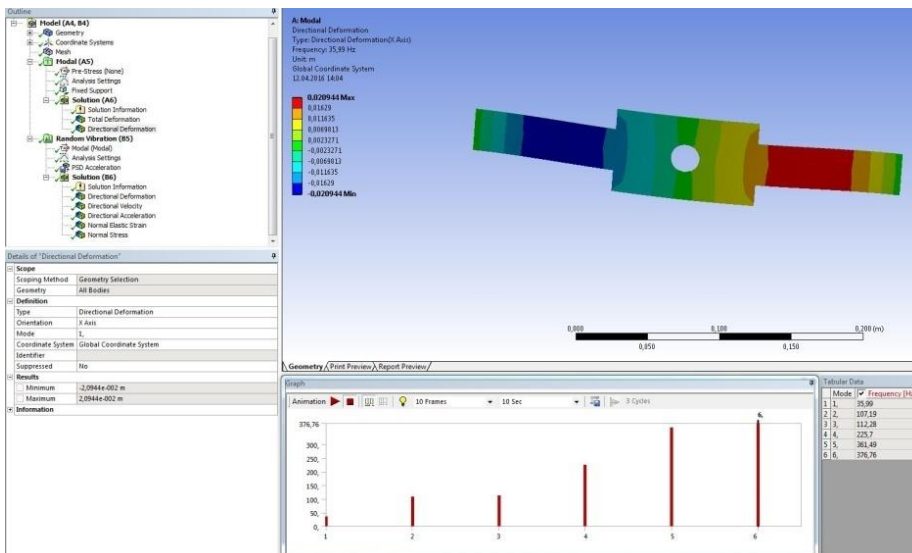


Рис. 3. Определение направлений деформации пластины

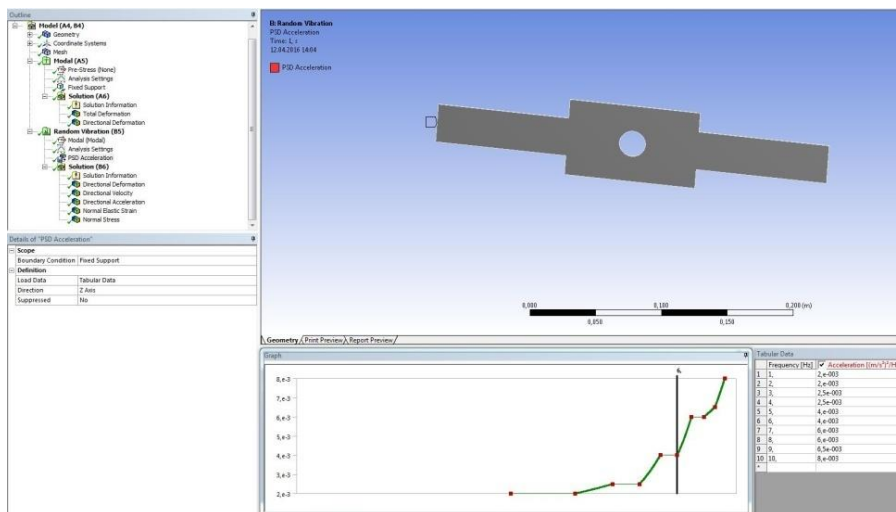


Рис. 4. Задание ускорений

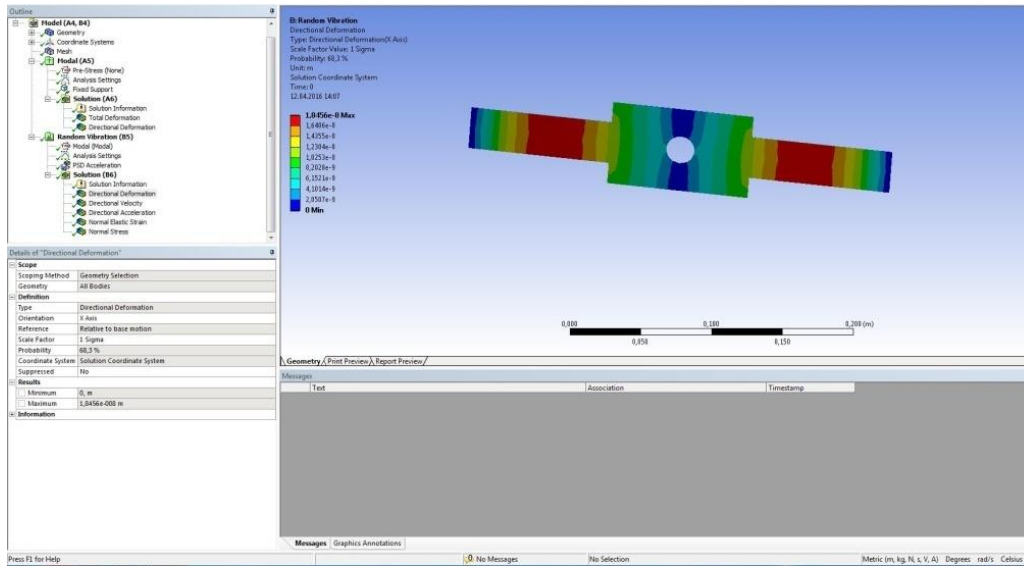


Рис. 5. Определение направленного перемещения пластины

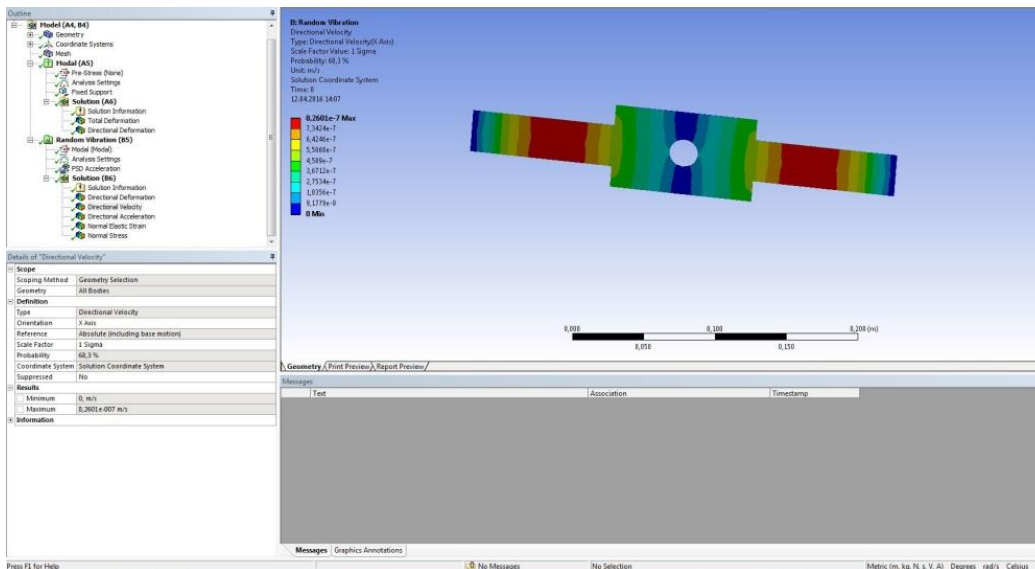


Рис. 6. Определение направленной скорости

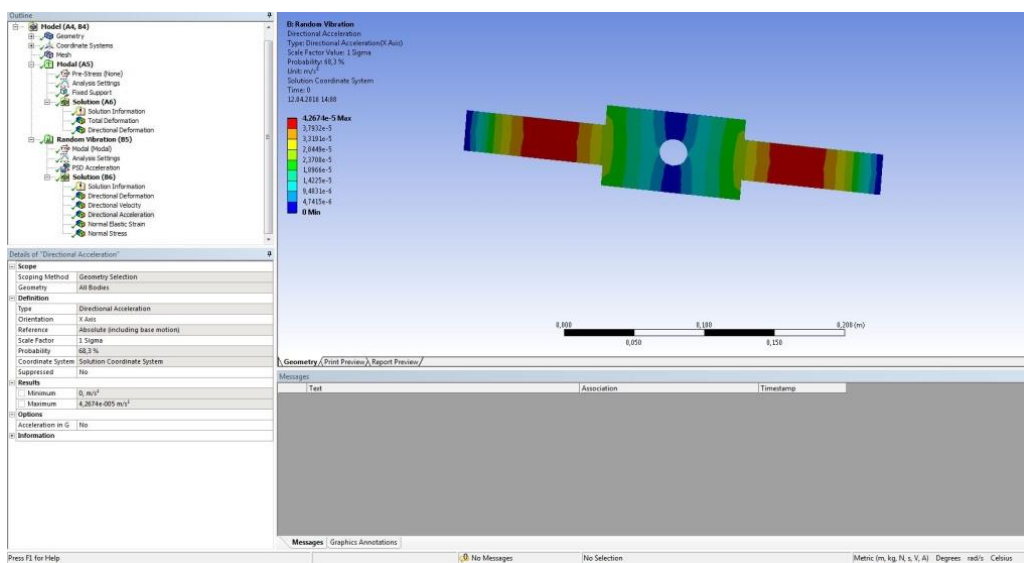


Рис. 7. Определение направленного ускорения

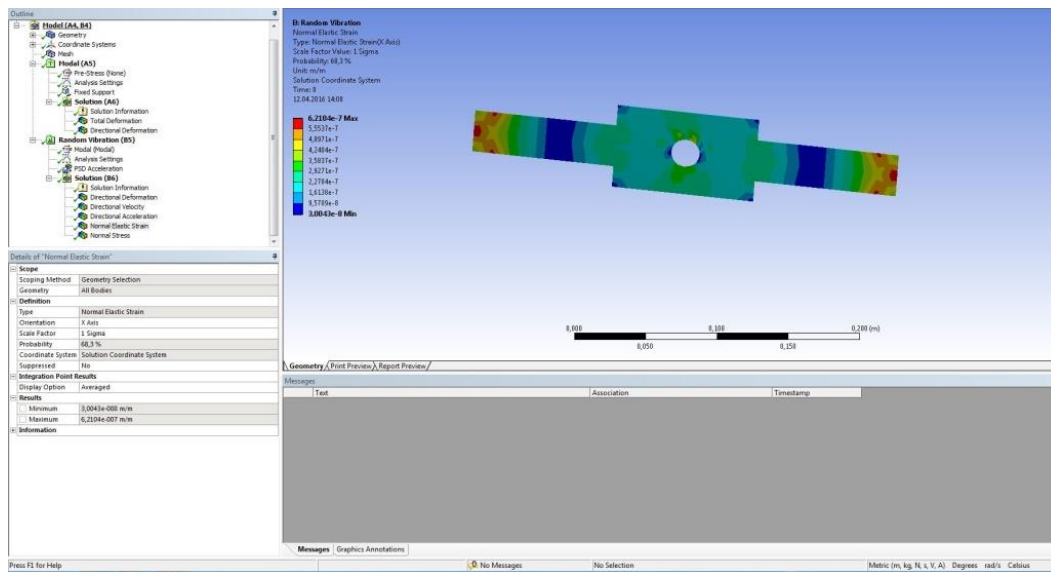


Рис. 8. Изменение структуры материала пластины

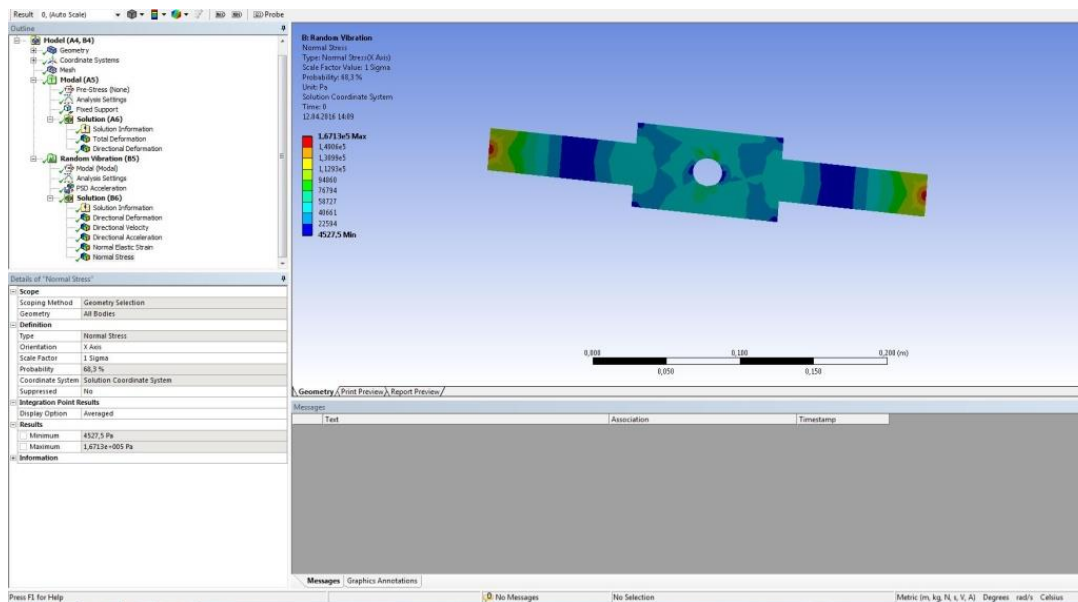


Рис. 9. Результаты определения нормального напряжения

Из данного исследования можно сделать вывод, что при помощи программы ANSYS Mechanical можно провести такие исследования как: определения направленной деформации, направленного перемещения, скорости и ускорения. Можно графически показать изменение формы структуры материала нормальное, а также упругое. Есть возможность рассчитать нормальное напряжение.

### Библиографический список

1. Смирнов О.Л. Использование комплекса ANSYS при проектировании приборов и электронных приборов и электронных средств. Методические указания к лабораторным работам. ГУАП, 2014. 138 с.
2. Козлов А.А., Смирнов О.Л. Моделирование и исследование конструкций электронных узлов с использованием SOLIDWORKS и ANSYS. Методические указания к лабораторным работам. ГУАП, 2015. 59 с.

УДК 629.735.33

**А.А. Борискин**

магистрант кафедры проектирования и технологии электронных и лазерных средств

**Д.К. Шелест**

проф., д-р. техн. наук – научный руководитель

**МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ОДНОРОДНОЙ ПРЯМОУГОЛЬНОЙ ПЛАТЫ, СВОБОДНО ОПЕРТОЙ ПО КРАЯМ, С ВНЕШНИМ ДЕМПФИРУЮЩИМ СЛОЕМ**

При изгибных колебаниях конструкций ячеек с внешним ДС (рис.1) основными видами деформации слоев являются изгиб и растяжение [1]. Деформации сдвига значительно меньше и ими обычно пренебрегают. При нанесении ВП материала на основную конструкцию, например плату или шасси, нейтральный слой конструкции смещается в направлении ДС. Вид деформации конструкции показан на рис.2. Так как основными видами деформации являются изгиб и растяжение- сжатие, то потенциальная энергия каждого слоя:

$$\Pi_i = \Pi_{iu} + \Pi_{ip}, \tag{1}$$

где  $\Pi_{iu}$ ,  $\Pi_{ip}$  – составляющие потенциальной энергии в  $i$ -м слое, обусловленные деформациями изгиба и растяжения-сжатия соответственно.

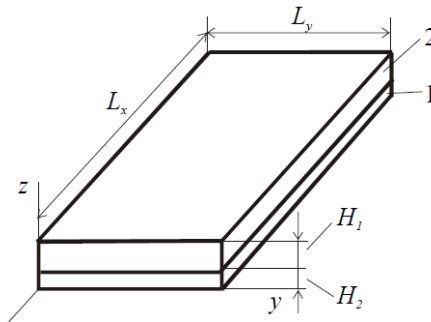


Рис. 1. Плата с внешним демпфирующим слоем: 1 – демпфирующий слой, 2 – несущий слой

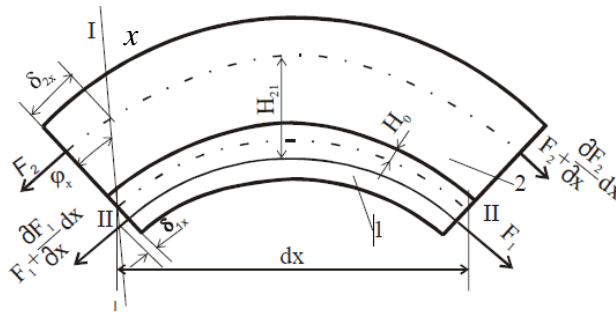


Рис. 2. Вид деформации платы: 1 – демпфирующий слой; 2 – несущий слой

Величины  $\Pi_{iu}$  и  $\Pi_{ip}$  можно найти по известным формулам теории упругости [2]:

$$\Pi_{iu} = \frac{D_2}{2} \int_0^a \int_0^b \left\{ \left( \frac{\partial^2 W}{\partial^2 x} + \frac{\partial^2 W}{\partial^2 y} \right)^2 + 2(1-\nu_i) \left[ \frac{\partial^2 W}{\partial^2 x} \frac{\partial^2 W}{\partial^2 y} + \left( \frac{\partial^2 W}{\partial x \partial y} \right)^2 \right] \right\} dx dy; \tag{2}$$

$$\Pi_{ip} = \frac{E_i H_i}{2(1-\nu_i)} \int_0^a \int_0^b \left\{ \left( \frac{\partial U_i}{\partial x} \right)^2 + \left( \frac{\partial V_i}{\partial y} \right)^2 + 2\nu_i \frac{\partial U_i}{\partial x} \frac{\partial V_i}{\partial y} \right\} dx dy, \tag{3}$$

где  $W = W(x, y)$  – деформация изгиба;  $U_i, V_i$  – продольные деформации  $i$ -го слоя в направлении  $x$  и  $y$  соответственно.

Так как для прямоугольных плат, свободно опертых по всем краям, форма колебаний при резонансе определяется выражением

$$W(x,y) = A_{if} \sin k_x x \sin k_y y, \quad (4)$$

где волновые числа  $k_x = l\pi / a$ ,  $k_y = l\pi / b$ , то выражение в формуле (2)

$$\int_0^a \int_0^b \left[ \frac{\partial^2 W}{\partial x^2} \frac{\partial^2 W}{\partial y^2} - \left( \frac{\partial^2 W}{\partial x \partial y} \right)^2 \right] dx dy = 0.$$

Представим связь между деформациями изгиба  $W(x,y)$  и растяжения – сжатия  $U_i, V_i$  в виде

$$U_i = R_{ix} \frac{\partial W}{\partial x}; \quad V_i = R_{iy} \frac{\partial W}{\partial y}, \quad (5)$$

где  $R_{ix}, R_{iy}$  – неизвестные коэффициенты связи деформаций.

Подставляя (4) и (5) в (2) и (3) получаем:

$$\Pi_{ip} = \frac{D_i}{8} (k_x^2 + k_y^2)^2 A_{if} ab; \quad (6)$$

$$\Pi_{ip} = \frac{E_i H_i}{8(1-\nu_i^2)} (R_{ix}^2 k_x^4 + R_{iy}^2 k_y^4 + 2\nu_i R_{ix} R_{iy} k_x^2 k_y^2)^2 A_{if} ab. \quad (7)$$

Коэффициенты связи деформаций найдем из статических и геометрических соотношений при изгибе. Для изгиба в направлении оси  $X$  можно записать

$$U_1 + U_2 = \phi_x H_{21}; \quad (8)$$

$$\frac{\partial F_{1x}}{\partial x} - \frac{\partial F_{2x}}{\partial x} = 0. \quad (9)$$

Продольные силы  $F_{ix}$  найдем из следующих соображений. В каждом слое, как показано на рис. 3, действуют нормальные напряжения  $\sigma_x$ , обусловленные продольными и изгибными деформациями.

Силу  $F_{ix}$  найдем интегрированием этих напряжений по площади поперечного сечения:

$$F_i = \int_0^{H_{i1}} \int_{-H_{i2}}^{H_{i1}} \sigma_x dy dx.$$

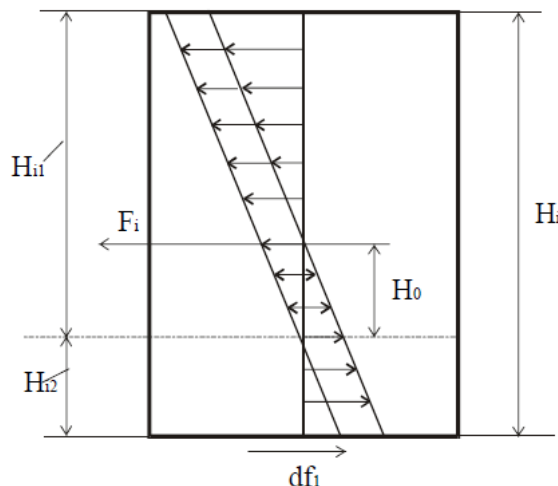


Рис. 3. Силы, действующие в  $i$ -м слое при изгибе двухслойной плат

Подставляем в это выражение формулу обобщенного закона Гука

$$\sigma_x = -\frac{E_i Z}{1-v_i} \left( \frac{1}{\rho_x} + v_i \frac{1}{\rho_y} \right),$$

где  $\rho_x, \rho_y$  – радиусы кривизны плат в направлении осей  $X$  и  $Y$ ;  $Z$  – расстояние от некоторого слоя до нейтрального слоя.

Интегрируя, получаем:

$$F_i = -\frac{E_i Z}{1-v_i^2} \left( \frac{1}{\rho_x} + v_i \frac{1}{\rho_y} \right) \left( \frac{H_{i1}^2 - H_{i2}^2}{2} \right). \quad (10)$$

Для малых прогибов справедливы соотношения:

$$\frac{1}{\rho_x} \approx \frac{\partial^2 W}{\partial x^2} = 0; \quad \frac{1}{\rho_y} \approx \frac{\partial^2 W}{\partial y^2} = 0.$$

Учитывая выражение (4), получаем:

$$\frac{1}{\rho_x} = -k_x^2 W; \quad \frac{1}{\rho_y} = -k_y^2 W.$$

Подставим их в выражение (10). Замечая, что  $H_{i1} + H_{i2} = H_i$  и обозначая  $k_i = \frac{E_i Z}{(1-v_i^2)}$ , выражение (10) приведем к виду:

$$F_i = -\frac{E_i Z}{1-v_i^2} \left( 1 - v_i \frac{k_y^2}{k_x^2} \right).$$

Заметим, что выражение  $H_{i1} - H_{i2} / \rho_x$  определяет относительную продольную деформацию  $\varepsilon_{ix} = \partial U_i / \partial x$  средней плоскости  $i$ -го слоя при действии силы  $F_i$ .

Учитывая соотношение  $\partial^2 U_i / \partial x^2 = -k_x^2 U_i$ , получаем:

$$\frac{\partial F_{ix}}{\partial x} = -k_i b k_x^2 U_i \left( 1 + v_i \frac{k_y^2}{k_x^2} \right).$$

Подставляя это выражение в (10), запишем:

$$k_1 U_1 \left( 1 + v_i \frac{k_y^2}{k_x^2} \right) - k_2 U_2 \left( 1 + v_i \frac{k_y^2}{k_x^2} \right) = 0, \quad (11)$$

Решая совместно уравнения (8) и (11), находим:

$$U_1 = \frac{H_{21} k_{2x}}{1 + k_{2x}} \phi_x; \quad U_2 = \frac{H_{21}}{1 + k_{2x}} \phi_x, \quad (12)$$

где

$$k_{2x} = \frac{E_2 H_2 (1 + v_1^2) (1 + v_2 k_y^2 / k_x^2)}{E_1 H_1 (1 + v_2^2) (1 + v_1 k_y^2 / k_x^2)}. \quad (13)$$

Сравнивая (12) и (5) и принимая во внимание, что  $\phi_x = \partial W / \partial x$

$$R_{1x} = \frac{H_{21} k_{2x}}{1 + k_{2x}} \phi_x; \quad R_{2x} = \frac{H_{21} k_{2x}}{1 + k_{2x}} \phi_x. \quad (14)$$

Выражения (1), (6), (7), (14) являются математической моделью однородной прямоугольной платы, свободно опертой по краям, с внешним демпфирующим слоем.

**Библиографический список**

1. Ю.В.Зеленев, А.А.Кирилин, Э.Б.Слободник, Е.Н.Талицкий Виброзащита радиоэлектронной аппаратуры полимерными компаундами; Под ред. Ю.В.Зеленева. М.: Радио и связь, 1984. 120с.
2. Бабаков И.М. Теория колебаний. М.: Наука, 1968. 560 с.

УДК 621.8.036:621.382.2:004.9

**И.В. Дегтярь**

студент кафедры конструирования и технологий электронных и лазерных средств

**О.Л. Смирнов**

доц., канд. техн. наук. доц. – научный руководитель

**СТАЦИОНАРНЫЙ ТЕПЛОВОЙ АНАЛИЗ ПЛАСТИНЫ**

Цель работы: провести стационарный тепловой анализ конструкции (рис.1) . Провести нелинейный тепловой расчет твердотельной (Solid) пластины с двумя отверстиями, нагреваемой конвекцией с теплопроводностью, как функцией температуры. Эта пластина служит теплоотводом в приборах. Построить графики по произвольно заданному пути и получить график температур и теплового потока.

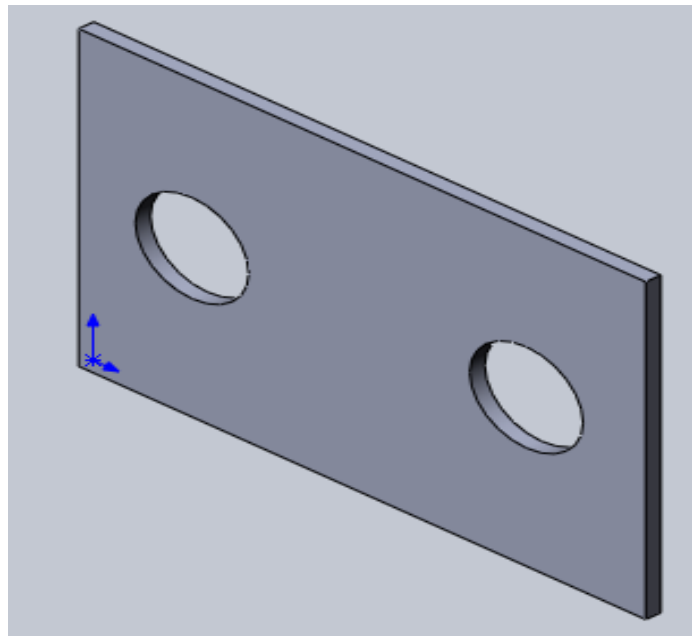


Рис. 1. Модель пластины с двумя отверстиями

Действующая лабораторная работа по стационарному тепловому анализу пластины выполнена в среде Ansys на языке APDL, Возникла задача разработать аналогичную лабораторную работу в среде Ansys Workbench. В Ansys Workbench в списке всех процессов выбираем стационарный тепловой анализ. В Проекте запускаем редактор библиотек, выбираем нужную нам библиотеку и материалы для пластины.

Модель пластины изначально была выполнена в SolidWorks. Чтобы Ansys смог читать модель, сохраняем ее в формате .STEP, и импортируем в проект геометрии Ansys.

Далее проводим разбиение на конечные элементы (сетка). Добавляем два источника тепла для температурного анализа на торцах пластины с температурами 20° и 110°. Добавляем проведение теплового расчета, общего теплового потока и направленного теплового потока. В

заклучение производим решение общей модели. Получили результаты теплового расчета, общего теплового потока и направленного теплового потока (рис. 2 – 4).

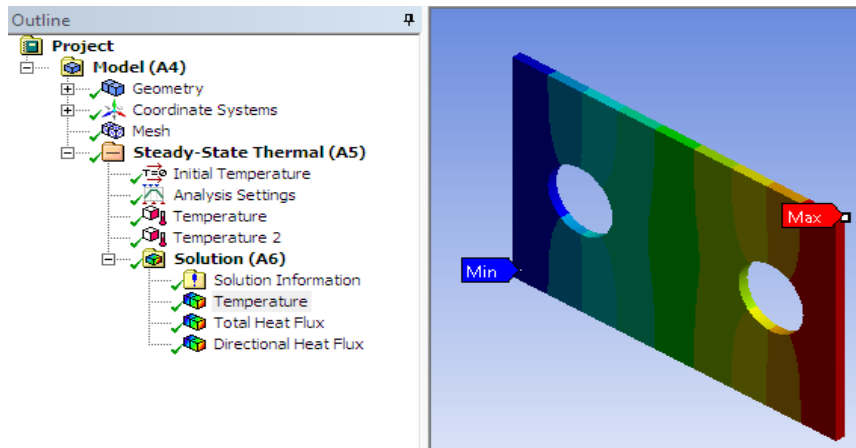


Рис.2. Иллюстрация теплового расчета

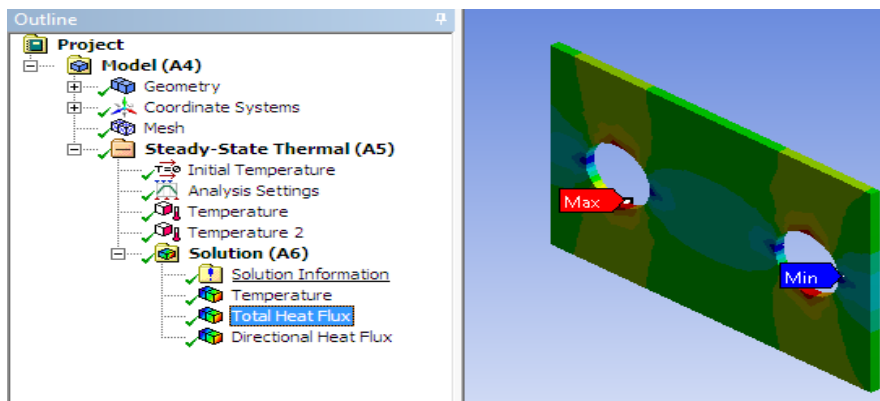


Рис.3. Определение общего теплового потока

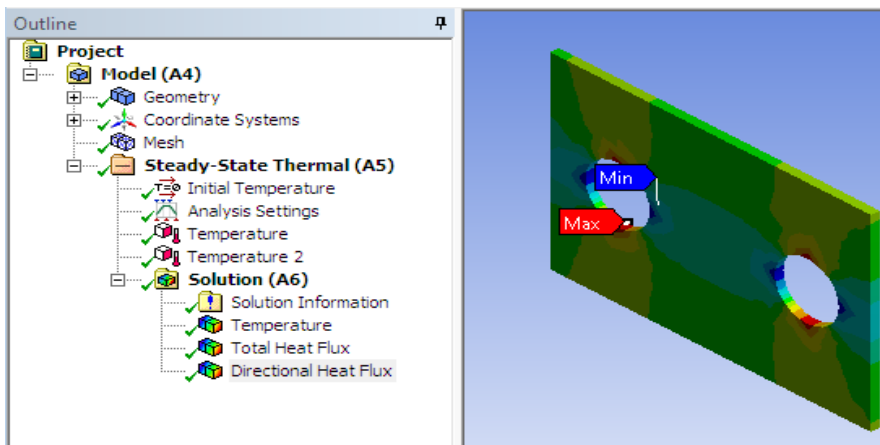


Рис.4. Определение направленного теплового потока

### Библиографический список

1. Смирнов О.Л. Использование комплекса ANSYS при проектировании приборов и электронных приборов и электронных средств. Методические указания к лабораторным работам. ГУАП, 2014. 138 с.
2. Козлов А.А., Смирнов О.Л. Моделирование и исследование конструкций электронных узлов с использованием SOLIDWORKS и ANSYS. Методические указания к лабораторным работам. ГУАП, 2015. 59 с.



УДК 621.314; 621.3.036.2

**Д.А. Жданов**

студент кафедры микро- и нанотехнологий аэрокосмического приборостроения

**И.А. Гарютин**

старший преподаватель – научный руководитель

### ОБЗОР ВОЗМОЖНОСТЕЙ СОВРЕМЕННЫХ ПАКЕТОВ МОДЕЛИРОВАНИЯ ТЕПЛОВЫХ РЕЖИМОВ

Одной из важнейших задач, возникающих сегодня у разработчиков радиоэлектронных средств, является задача контроля температуры, в том числе и перегрева электронных компонентов. С развитием компьютерных технологий стали появляться новые возможности в моделировании тепловых режимов для разработки более эффективных методов охлаждения. В данной статье рассматриваются современные возможности программных пакетов моделирования тепловых процессов.

Наиболее популярным пакетом, используемым во многих университетах, является MATLAB. Это высокоуровневый язык и интерактивная среда для программирования, численных расчетов и визуализации результатов. Расширением, используемым для моделирования и симуляции широкого спектра термодинамических систем из различных областей, является Thermolib [1]. Это расширение предоставляет набор блоков Simulink для моделирования систем и набор функций MATLAB для термодинамических расчетов. Thermolib содержит обширный набор термодинамических и термохимических блоков, которые хорошо интегрированы в окружение MATLAB и Simulink. Фундаментальные термодинамические блоки помогают пользователям создавать свои собственные компоненты. База данных включает множество распространенных в природе веществ и эта база может легко расширяться пользователем. Thermolib может рассчитывать поведение реального газа с помощью уравнения Пенга-Робинсона. Такой подход реализует хорошую точность в окрестности критической точки и при конденсации. Для еще более подробных расчетов, включающих воду и пар, доступны термодинамические свойства по IAPWS-IF97 (промышленное образование, принято транснациональным объединением для свойств воды и пара). Широкий набор демонстрационных примеров, охватывающих процессы сгорания, циклы подвода и отвода тепла, топливные ячейки, газовые турбины, контроль тепловыделения батарей и другие важные компоненты помогают пользователю быстро приступить к работе.

Довольно важным дополнением является возможность совместного использования в MATLAB пакетов SimPowerSystems и Thermolib. Это позволяет моделировать элементы силовой электроники и связывать тепловые процессы с электрическими, протекающими в различных цепях.

SolidWorks – программный комплекс САПР для автоматизации работ промышленного предприятия на этапах конструкторской и технологической подготовки производства. Данный пакет больше известен как инженерная CAD-система трехмерного моделирования. Дополнительные расчетные модули SolidWorks Simulation позволяют проводить тепловые расчеты. Основные особенности:

- расчет теплового потока вблизи адиабатных стенок или в твердотельных телах указание различных типов источников тепла;
- определение зависящих от температуры свойств материала;
- определение термических нагрузок и граничных условий;
- назначение моделям различных твердотельных материалов, которые хранятся в инженерной базе данных;
- определение собственных материалов путем назначения им значений для физических свойств, таких как теплопроводность, теплоемкость и т.д.;

- расчет теплоты излучения;
- создание эпюры температуры, температурных градиентов и тепловых потоков;
- создание исследования переходного термического процесса;
- определение зависящий от времени термической нагрузки посредством импортирования предопределенной кривой времени;
- вычерчивание графиков термических результатов относительно времени.

Данный пакет позволяет описать распределение тепловых потоков в несложных по составу изделиях, но не очень годится для сложных радиотехнических систем. Хотя и позволяет рассчитать тепловые характеристики охлаждающего радиатора или процесс теплопереноса при использовании вентиляторов.

ANSYS имеет довольно широкую область применения. В авиационной и космической технике его используют для: определения температурного и термонапряженного состояния, моделирования стационарных и нестационарных задач теплообмена, решения задач охлаждения и деформации материалов из-за нагрева, рассмотрения влияния термоциклирования на прочностные характеристики. В машиностроении: определение полей температур и тепловых потоков, как в отдельных деталях, так и целых агрегатах или конструкциях; моделирование нестационарных и стационарных задач теплообмена, моделирование тепловых контактов и тепловых зазоров, расчет термонапряженного состояния устройства, расчет переходных режимов работы агрегатов, моделирование процессов нагрева или остывания. Также ANSYS используют для моделирования охлаждения электроники. Сюда входят: анализ теплового состояния микрочипов, печатных плат и электронных устройств; определение теплопроводности печатных плат; учет джоулевых потерь; выбор системы охлаждения; расчет термонапряженного состояния печатных плат и электронных блоков. Программные продукты ANSYS для теплового анализа:

- ANSYS Mechanical. Позволяет решать широкий спектр задач механики деформируемого твердого тела и теплообмена с учетом нелинейных свойств материалов, пластичности и контактного взаимодействия.

- ANSYS CFD (ANSYS CFX + ANSYS Fluent). Предназначен для моделирования ламинарных и турбулентных потоков, расчета процессов теплообмена (конвекции, теплопроводности, излучения), процессов горения, моделирования многофазных потоков и решения задач акустики.

- ANSYS Icerak. Используется для расчета температурного состояния электронных устройств и компонентов.

Существуют и другие программы, выбор которых для пользователей зависит от того, какую задачу требуется решить. Из рассмотренных пакетов, MATLAB наиболее приспособлен для решения серьезных теоретических вопросов. Пакет SolidWorks Simulation имеет больше практическое значение с возможностью посмотреть на 3D-модели распределение тепловых полей и перегрев. Для решения задачи моделирования тепловых режимов средств ПЭС и распределения температурных полей внутри компонентов более подходящим вариантом является ANSYS, так как содержит все нужные для этого функции.

### Библиографический список

1. <http://matlab.ru>
2. Руководство по основным методам проведения анализа в программе ANSYS. 399 с.
3. Шалумов А.С., Ваченко А.С., Фадеев О.А., Багаев Д.В. Введение в ANSYS: прочностной и тепловой анализ / Ковров: КГТА. 2002. 52 с.
4. Алямовский А.А. SolidWorks/COSMOSWorks: Издательство ДМК Пресс, 2004. 216 с.
5. Чен К., Джиблин П., Ирвинг А. – MATLAB в математических исследованиях: Издательство Мир, 2001. 346 с.

УДК 621.355

**К. В. Желудева**

магистрант кафедры конструирования и технологий электронных и лазерных средств

**В. П. Ларин**

д-р техн. наук, профессор.) – научный руководитель

### **ФАКТОРЫ, ОПРЕДЕЛЯЮЩИЕ ВЫБОР АККУМУЛЯТОРНЫХ БАТАРЕЙ ДЛЯ БОРТОВЫХ СИСТЕМ ЭЛЕКТРОСНАБЖЕНИЯ**

Из всех видов энергии, известных человечеству, электрическая энергия является наиболее универсальной. Электрическая энергия легко преобразуется в другие виды энергии, ее легко передавать по проводам к потребителю, а также распределять между потребителями. Ряд данных преимуществ определяет предпочтение электрической энергии в выборе необходимого вида энергии.

Приведение в действие всех элементов устройств и оборудования летательного аппарата (ЛА) (двигательная система ЛА, органов управления, систем связи, приборного комплекса и т.д.) невозможно без электрической энергии. Поэтому система электроснабжения (СЭС) является одной из важнейших систем, обеспечивающих работоспособность ЛА.

Бортовая СЭС представляет собой совокупность первичных и вторичных источников тока, аппаратуры преобразования энергии и стабилизации выходного напряжения с необходимой автоматикой контроля и управления [1]. Для бортовых СЭС ЛА наибольшее распространение в качестве одного из видов первичных источников тока получили аккумуляторные батареи (АБ).

Как известно, аккумуляторные батареи – химические источники электрической энергии многоразового действия. За основу работы АБ принят принцип возникновения разности потенциалов (напряжения) при фиксированной нагрузке между двумя пластинами (полюсами аккумулятора), погруженными в раствор электролита. Таким образом, АБ состоит из положительного и отрицательного электродов, электролита и корпуса. Заряд и разряд АБ обуславливается прямыми и обратными химическими реакциями окисления-восстановления электродов. [2]

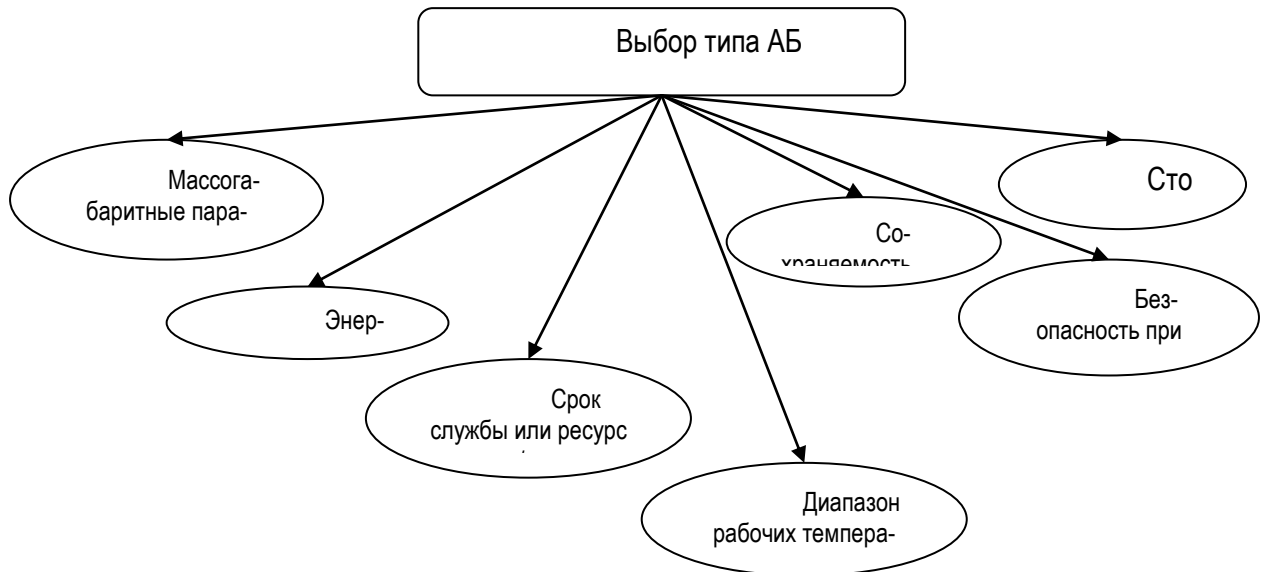
В настоящее время в качестве химических источников тока распространены такие АБ, как свинцово-кислотные, никель-кадмиевые Ni-Cd, никель-водородные Ni-H, серебряно-кадмиевые Ag-Cd, литий-ионные Li-Ion, серебряно-цинковые Ag-Zn. Выбор того или иного типа АБ обуславливается рядом факторов:

- низкая стоимость АБ;
- большая энергоемкость аккумуляторов;
- уменьшение веса и размеров (большая удельная энергия);
- большой срок службы или ресурс АБ (годы/циклы);
- хорошая сохраняемость (низкий саморазряд);
- работоспособность в широком диапазоне температур.

Перечисленные факторы, влияющие на выбор типа аккумуляторных батарей, представлены на рисунке.

В таблице приведены основные технико-эксплуатационные характеристики наиболее распространенных типов аккумуляторов герметичного исполнения.

Применительно к АБ бортовой СЭС одними из важнейших факторов являются массогабаритные параметры: вес и размеры АБ должны минимизироваться. Поэтому при выборе АБ стоит значительное внимание уделять такой характеристике, как удельная энергия Вт·ч/кг. Исходя из этого фактора, наилучшими показателями обладают литий-ионные и серебряно-цинковые АБ. Однако при этом последние значительно проигрывают по таким показателям, как срок службы и относительная стоимость за единицу энергии.



Факторы, обуславливающие выбор типа АБ

Характеристики наиболее распространенных типов аккумуляторов

Тип аккумулятора	Свинцово-кислотный	Никель-кадмиевый	Никель-водородный	Литий-ионный	Серебряно-цинковый
Удельная энергия, Вт·ч/кг	15-40	20-50	50-60	90-150	100-200
Рабочее напряжение, В	2	1,2	1,25	3,6	1,5
Саморазряд, % (сут.)	5 (30)	20 (28)	40 (3)	До 15 (28)	До 15 (30)
Отдача по емкости, %	-	72	90-92	98	-
Срок службы, годы/циклы	3-10/500	5/500	7/1000	8/1000	2/200
Относительная стоимость единицы энергии	0,8-1,8	1,5-2	-	2,5-4	15
Диапазон рабочих температур	От -20 до +40	От -40 до +45	От -20 до +30	От -10 до +50	От -40 до +40

Немаловажным фактором являются температурные условия, характеризующие относительную работоспособность АБ. С понижением температуры эффективность заряда при постоянном напряжении уменьшается из-за роста внутреннего сопротивления аккумулятора. В таблице приведены показатели диапазона рабочих температур для выбранных типов аккумуляторов.

Стоит понимать, что при достижении своего порога низкой и высокой температуры, АБ могут значительно снижать или полностью терять свою относительную работоспособность.

Так для серебряно-цинковых АБ понижение температуры от  $+20^{\circ}\text{C}$  до  $0^{\circ}\text{C}$  незначительно влияет на величину отдаваемой емкости, снижение до  $-10^{\circ}\text{C}$  снижает величину отдаваемой емкости примерно на 50%. Дальнейшее понижение температуры ведет к более резкому падению емкости. [3]

При повышении температуры на  $10^{\circ}\text{C}$  выше нормальной температуры эксплуатации происходит снижение работоспособности на 20% у никель-кадмиевых АБ, и на 50% у свинцово-кислотных.

Таким образом, различные типы АБ при одной температуре эксплуатации имеют разных показатели работоспособности, что необходимо учитывать при выборе аккумулятора.

Немаловажное значение при выборе АБ имеют значение такие параметры как, коэффициент отдачи и коэффициент полезного действия аккумулятора.

Коэффициент отдачи определяется как отношение количества электричества в кулонах, отданного аккумулятором при полном разряде, к количеству электричества, полученному при заряде. Коэффициент полезного действия аккумулятора определяется как отношение количества

электричества в кулонах, которое он отдает потребителю, разряжаясь до установленного предела для продолжения нормальной работы последнего, к количеству электричества, полученному им при заряде. [2]

При этом значение коэффициента полезного действия всегда меньше значения коэффициента отдачи.

Итак, при выборе типа аккумуляторных батарей необходимо учитывать ряд важных факторов. В настоящее время, наибольшее распространение в качестве АБ бортовой СЭС как первичного источника тока получили литий-ионные АБ. Однако, в качестве накопителя электрической энергии в разработке беспилотного гиперзвукового летательного аппарата – Нурег (НХRV) используются серебряно-цинковые батареи.

### Библиографический список

1. Шиняков Ю.А. Гуртов А. С., Гордеев К. Г., Ивков С. В. Выбор структуры систем электроснабжения низкоорбитальных космических аппаратов. Вестник Самарского государственного аэрокосмического университета. 2010. №1(21). С.103–113.
2. Лаврус В.С. Батарейки и аккумуляторы. Наука и Техника, 1995. 47 с.
3. Петровичев М.А. Давыдов Е. И. Системы оборудования летательных аппаратов: лабораторный практикум. Самара, 2002. 78 с.

УДК 621.313.29

**А.А. Канатова**

магистрант кафедры конструирования и технологий электронных и лазерных средств

**А.Г. Михайлов**

доц., канд. техн. наук – научный руководитель

## ВОЗМОЖНОСТЬ ИСПОЛЬЗОВАНИЯ ЭЛЕКТРОДВИГАТЕЛЯ ТИПА ДПР В МЕХАНИЗМЕ ПОДАЧИ ПОРОШКОВОЙ ПРОВОЛОКИ

Механизм подачи порошковой проволоки применяется в составе сварочных полуавтоматов и обеспечивает подачу проволоки по гибкому шлангу в зону сварки. Обычно он состоит из электродвигателя, редуктора и системы подающих и прижимных роликов.

Выделяют три вида механизмов подачи проволоки:

- толкающий;
- тянущий;
- толкающе-тянущий.

Привод толкающего механизма расположен в корпусе полуавтомата. Он проталкивает проволоку (диаметром не более 2 мм) сквозь механизм подачи проволоки по рукаву к сварочной горелке. Толкающий механизм является наиболее распространенным механизмом подачи проволоки в полуавтомат.

Привод тянущего механизма расположен в ручке горелки. Он притягивает к себе проволоку через рукав и направляет в сварочную горелку. Это менее распространенный механизм подачи проволоки.

Толкающе-тянущий механизм имеет два привода: один расположен в корпусе прибора, а второй в ручке сварочной горелки. Но этот механизм практически не используется: его применяют только тогда, когда рукав, соединяющийся с горелкой, очень длинный, например, для выполнения сварки в стесненных условиях [1].

Для мягких проволок из алюминия и его сплавов или порошковой проволоки с высоким коэффициентом трения (за счет неровностей поверхности проволоки), а также для проволок малого диаметра (менее 0,8 мм), способных сминаться в канале, полуавтоматы толкающего типа

неприменимы. В этих случаях следует использовать полуавтоматы тянущего типа. Это позволяет избежать потери продольной устойчивости проволоки и связанного с этим повышенного сопротивления проталкиванию. Кроме того, тянущий механизм обеспечивает уменьшение шлангового усилия протяжки, и прижимные ролики меньше деформируют мягкую проволоку. Однако длина шлангов этих полуавтоматов ограничивается малой мощностью электродвигателя (до 100 Вт) и не превышает 1 м. Более длинные шланги требуют увеличенной мощности и массы двигателей, а, следовательно, большой массы горелки.

Для механизма подачи порошковой проволоки, подача которой оптимальна механизмом тянущего типа, очень важно использовать электродвигатель с минимальными массогабаритными показателями из-за ограничения массы горелки.

В ходе работы был исследован российский рынок микродвигателей (рис. 1). В настоящее время в качестве исполнительного двигателя тянущего механизма активно используют электродвигатели постоянного тока зубцового исполнения и асинхронные электродвигатели. Нами был выбран коллекторный двигатель постоянного тока ДПР-62-Н1-07А (далее – ДПР), конструктивно отличающийся от двигателей зубцового исполнения.

Двигатели этой серии выполняют закрытого исполнения со степенью защиты от попадания посторонних частиц и воды IP44 (рис. 1, а). Основными элементами конструкции являются: постоянный магнит 1, якорь (ротор) 2, магнитопровод (корпус) 3, коллектор 4. При изготовлении корпуса (ротора) обмотку укладывают на цилиндрический каркас и заливают компаундом, получая таким образом ротор в виде полого тонкостенного немагнитного стакана. После заливки концы секций обмотки якоря соединяют с пластинами коллектора.

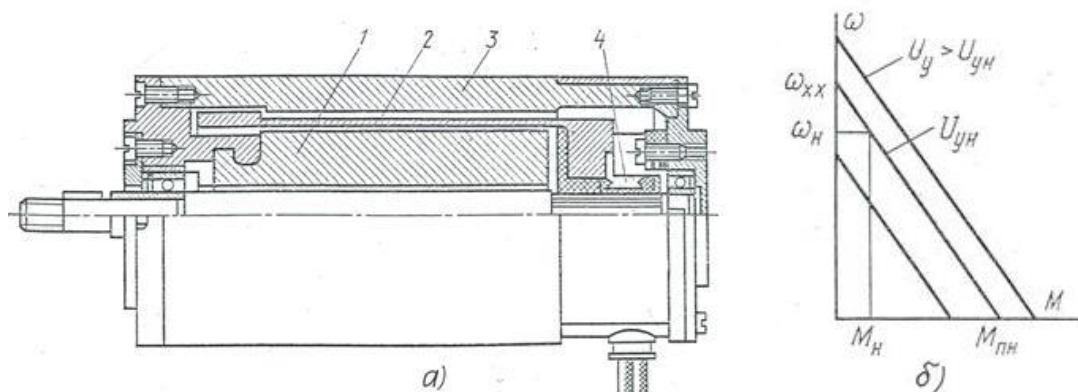


Рис. 1 – Микродвигатель постоянного тока с полым якорем серии ДПР:  
а – конструкция двигателя; б – механические характеристики

Выбранный электродвигатель обладает рядом преимуществ, благодаря которым целесообразно его использование в качестве исполнительного двигателя тянущего механизма подачи порошковой проволоки.

Во-первых, двигатель ДПР имеет максимальное значение отношения момента вращения к массе (номинальный момент – 19,6 мН·м, масса – 0,4 кг), т. е. двигатель ДПР высокомоментный и имеет при этом минимальные габариты и массу, поэтому может применяться в тянущем механизме, не увеличивая размеры автомата и не сильно утяжеляя ручку горелки.

Во-вторых, как и все двигатели постоянного тока, ДПР имеет линейные механические характеристики (рис. 1, б) – зависимость частоты вращения ротора от момента нагрузки на валу двигателя и напряжения управления. Благодаря этому ДПР легко регулируется в широком диапазоне, что важно при изменении скорости подачи проволоки в зону сварки.

Кроме того, положительным качеством ДПР является высокое быстродействие, обусловленное малым моментом инерции якоря, не содержащего ферромагнитных материалов и малой индуктивностью обмотки якоря. Постоянная времени двигателей типа ДПР находится в пределах 0,01–0,02 с.

По величине ресурса двигателя с полыми якорями значительно превышают ресурс двигателей с зубцовыми якорями при той же мощности и частоте вращения. Из-за очень малой величины индуктивности якорной обмотки и больших воздушных зазоров наработка у современных двигателей с полыми якорями находится в пределах от 1000 до 5000 ч – в зависимости от мощности и частоты вращения.

Недостатками конструкции двигателя постоянного тока является наличие коллектора и значительного немагнитного промежутка в магнитной цепи двигателя, который складывается из двух воздушных зазоров и толщины полого ротора. Это требует увеличения намагничивающей силы возбуждения. Однако, так как в данной конструкции двигателя на якоре отсутствуют ферромагнитные массы, а на статоре – обмотки возбуждения, а следовательно, отсутствуют потери в них, то коэффициент полезного действия таких двигателей (30–45%) находится примерно на том же уровне, что и в других двигателях.

По теплостойкости двигателя с полыми якорями значительно уступают двигателям с зубцовыми якорями. Если у последних теплоустойчивость обусловлена теплостойкостью эмалевого провода (в пределах 200–250°C), то у двигателей с полым якорем – теплостойкостью конструкции якоря. Поскольку обмотка залита эпоксидным компаундом, их теплостойкость определяется температурой начала деформации этого компаунда и находится в пределах 125–135°C. По этой причине предельная допустимая повышенная температура окружающей среды у двигателей с полым якорем ниже, чем у зубцовых.

Есть еще один фактор, резко ограничивающий возможность форсирования двигателей с полыми якорями по мощности, особенно при кратковременных и повторно-кратковременных режимах работы. Из-за малой массы и теплоемкости полых якорей, скорость достижения полыми обмотками предельно допустимой температуры при перегрузках по току во много раз больше, чем у обычных, зубцовых якорей. Это резко ограничивает допустимую длительность их работы в режимах форсированной мощности в сравнении с двигателями, имеющими зубцовые якоря.

Таким образом, можно сделать вывод, что двигатели с полым якорем, благодаря максимальному значению отношения момента вращения к массе, высокому быстродействию, минимальному статическому моменту трения и хорошей равномерности вращения являются практически идеальными исполнительными элементами для высококачественных широкодиапазонных регулируемых электроприводов, в том числе и для тянущего механизма подачи порошковой проволоки. Но возможность применения таких двигателей ограничена теплостойкостью ротора (температура ротора не должна превышать 120°C), поэтому в каждом частном случае необходимо производить тепловой расчет двигателя для конкретных значений диапазонов токов и скоростей вращения и проверять возможность работы двигателя при повышенных температурах и возможность работы двигателя без перегрева в повторно-кратковременном режиме, если необходимо, чтобы полуавтомат работал в режиме точечной сварки.

### Библиографический список

1. *Васильев В.И.* и др. Введение в основы сварки: учеб. пособие. Томск: Томский политехнический университет, 2011. 317 с.

УДК 537.877

**Н.В. Корчигин**

студент кафедры конструирования и технологий электронных и лазерных средств

**Д.К. Шелест**

д-р техн. наук, профессор – научный руководитель

## АНАЛИЗ ЗАДАЧИ ЭЛЕКТРОМАГНИТНОЙ СОВМЕСТИМОСТИ ПРИЕМО-ПЕРЕДАЮЩЕЙ СИСТЕМЫ

Электромагнитная совместимость (ЭМС) является одним из основных факторов при проектировании современной радиоэлектронной аппаратуры (РЭА). В наши дни количество РЭА, используемой в одном месте, стремительно увеличивается, поэтому необходимо увеличить электромагнитную защищенность узлов от воздействия посторонних шумов оказывающих влияние на надежную и стабильную работу РЭА. Необходимость электромагнитной защищенности была вызвана бурным ростом внедрения микроэлектроники практически во все виды электронных средств. Стабильное состояния ЭМС будет выполнено только в том случае, когда электротехнические и электронные приборы будут способны выполнять заданные им функции не реагируя на помехи, созданные электротехническими изделиями, и помехи, созданными природными явлениями. При проектировании РЭА должен охватываться широкий круг вопросов: от специфических особенностей систем до конструирования и технологии производства, причем с учетом не только технических аспектов но так же экономических, метрологических и организационных.

Для решения задач ЭМС разработано большое количество подходов и решений, которые можно разбить на несколько групп, под общим понятием «меры по обеспечению ЭМС»[1].

Различают следующие меры:

- *организационно-технические* (организационное, правовое и техническое регулирование в области использования радиочастотного ресурса);
- *системно-технические* (выбор принципов построения радиосистем с учетом требований ЭМС и организация работы отдельных устройств в составе системы, комплекса);
- *схемотехнические* (технические приемы, относящиеся как к улучшению параметров ЭМС конкретной РЭА, так и к использованию специальных устройств, предназначенных для компенсации влияния помех);
- *конструкторско-технологические* (уменьшение нежелательных электромагнитных помех, создаваемых источниками помех, восприимчивости рецепторов и, главным образом, устранение путей распространения помех путем более совершенного конструктивного исполнения самих узлов).

Поскольку данная тема является довольно обширной и одной статьи будет недостаточно для раскрытия всех нюансов данной тематики, сосредоточимся на рассмотрении коммутационных помех печатных узлов и их конструкторско-технологических решениях.

Когда возвратный путь в линии передачи не является однородной плоскостью, или диэлектрический материал достаточно асимметричен, тогда индуктивная связь значительно превосходит емкостную, и помехи будут определяться взаимной индуктивностью контуров. Это явление обычно проявляется в малых локальных зонах, таких, как выводы микросхем, соединители, области печатной платы, в которой возвратный путь прерывается щелью. В этом случае связь может быть смоделирована сосредоточенной индуктивностью.

Помехи, которые генерируются в пассивной линии за счет этой индуктивности, проявляются только при наличии изменения тока  $dl / dt$  в активной линии за время длительности фронта, поэтому эти помехи называются коммутационными помехами. Подскок напряжения заземления – одна из форм проявления коммутационных помех в особых случаях, когда существует некоторый



обходной возвратный путь. Можно указать три способа уменьшения этого эффекта. Для этого следует:

1. Увеличить число возвратных путей так, чтобы не было возможности найти еще один неучтенный возвратный путь.
2. Увеличить ширину и уменьшить длину возвратного пути с целью минимизации его парциальной самоиндукции.
3. Располагать сигнальный путь рядом с возвратным путем для уменьшения их парциальной взаимной индукции.

Даже в том случае, если нет альтернативного возвратного пути, перекрестная помеха между двумя или более линиями передачи будет определяться взаимной индуктивностью. Взаимная индуктивность контура в корпусах микросхем и соединителях часто является доминирующим фактором.

Оценка взаимной индуктивности двух контуров может быть проведена на простой модели [2]. Когда сигнал проходит по одной паре штыревых контактов в соединителе (активные выводы), происходит внезапное изменение тока в контуре при прохождении фронта сигнала. Это вызывает (за счет взаимной индуктивности) появление индуцированного напряжения в соседнем, пассивном контуре. Это напряжение  $V_b$  примерно равно

$$V_b = M \frac{dI_A}{dt} = M \frac{V_A}{t_R Z}, B$$

где  $M$  – взаимная индуктивность между активным и пассивным контуром, нГн;  $I_A$  – изменяющийся ток в активной линии, В;  $Z$  – типовое значение волнового сопротивления (около 50 Ом) для рассматриваемого сигнала в активном и пассивном контуре, Ом;  $V_A$  – напряжение сигнала в активной линии, В;  $t_R$  – фронт сигнала, нс.

Приемлемое значение коммутационной помехи зависит от общей помехоустойчивости. Опыт говорит, что это значение должно составлять от 5 до 10% от размаха сигнала [2]. Если приемлемый уровень помех определен, то можно рассчитать максимально допустимую индуктивность  $M$  между активной и пассивной цепями:

$$M = \frac{V_b}{V_A} t_R Z, нГн$$

где  $Z$  волновое сопротивление, Ом;  $V_A$ ,  $V_b$  – соответственно напряжение сигнала в активной и пассивной линиях, В;  $t_R$  – фронт сигнала, нс.

Для примера примем  $V_b / V_A = 0,05$ ;  $Z = 50 \text{ Ом}$ ;  $t_R = 1 \text{ нс}$ . Тогда получим максимально допустимую взаимную индуктивность  $2,5 \text{ нГн}$ . При уменьшении фронта скорость изменения тока  $dI / dt$  увеличивается, и следует уменьшить взаимную индуктивность для удержания уровня коммутационных помех в заданных пределах.

Практическое правило: для удержания коммутационных помех на приемлемом уровне между парой сигнальных и возвратных путей, взаимная индуктивность контуров между ними должна быть менее  $2,5 \text{ нГн}$  умноженного на (фронт сигнала (нс)).

При фронте  $0,5 \text{ нс}$  взаимная индуктивность не должна быть более  $1,25 \text{ нГн}$ , что представляет значительные трудности при создании корпусов микросхем и соединителей. Можно назвать три геометрии, которые приводят к снижению взаимной индуктивности:

- сокращение длины контура, что типично для корпусов CSP;

– разнесение активного и пассивного контуров, что трудно выполнить в миниатюрной аппаратуре;

– сближение прямого и возвратного путей, что снижает парциальную самоиндукцию, а также и волновое сопротивление; это, в свою очередь, может привести к новым проблемам, связанным с неоднородностью линии передачи.

Практическое правило: минимальная длительность фронта,  $нс$ , ограниченная допустимым уровнем коммутационных помех, составляет  $t_R > M / 2,5нГц$ . Это определяет максимальную тактовую частоту на уровне  $10 / t_R = 250МГц / MнГц$ . Период равен  $10t_R$ . Например, если взаимная индуктивность между парой сигнальных путей составляет  $1нГц$ , максимальная рабочая частота будет  $250МГц$ . Если будет пять активных пар, которые индуцируют помехи в пассивную линию, и каждая из них имеет взаимную индуктивность  $1нГц$  то максимальная частота будет  $250МГц / 5 = 50МГц$ .

### Библиографический список

1. Пудовкин А.П., Панасюк Ю.Н., Чернышова Т.И. Электромагнитная совместимость и помехозащищенность РЭС: Учебное пособие / ТГТУ Тамбов, 2013. 92 с.
2. Ефанов В.И., Тихомиров А.А. Электромагнитная совместимость радиоэлектронных средств и систем: Учеб. Пособия / ТУСУР Томск, 2012. 228 с.

УДК 621.371.39.002.2:621.9.048.7

**А. И. Кривалова**

студентка кафедры конструирования и технологий электронных и лазерных средств

**В. П. Ларин**

проф., д-р. техн. наук, проф. – научный руководитель

## НАПРАВЛЕНИЯ РАЗВИТИЯ ТЕХНОЛОГИЙ ПЛАЗМЕННОЙ ОЧИСТКИ И ТРАВЛЕНИЯ

Микроэлектроника – бурно развивающаяся область электронной техники. Основной тенденцией развития электроники и микроэлектроники является непрерывное повышение степени функциональной и конструктивной интеграции, достижение высокой точности микронных и субмикронных элементов и высокой плотности компонентов на плоскости и в объеме корпуса. Это достигается за счет перехода от больших интегральных схем к сверхбольшим и ультра большим, уменьшения размеров элементов микросхем, использования методов «система на кристалле» и «система в корпусе».

В данной статье изложены результаты анализа по направлениям использования и перспективам развития плазменных технологий в производстве электронных и микроэлектронных изделий. Эти технологии являются сравнительно новыми и относятся к развивающимся. В задачи статьи входит рассмотрение возможностей известных методов, направлений исследований по достижению более эффективных результатов и различные плазменные операции с использованием низкотемпературной неравновесной газоразрядной плазмы (плазма низкого давления) и плазмы атмосферного давления.

Плазма – это частично или полностью ионизированный газ, образованный из нейтральных атомов (или молекул) и заряженных частиц (ионов и электронов). Различают несколько видов плазмы:

– изотермическая плазма – когда энергии (температуры) всех составляющих плазму частиц равны ( $T_e \approx T_i \approx T_g$ ) и все процессы обмена являются равновесными.

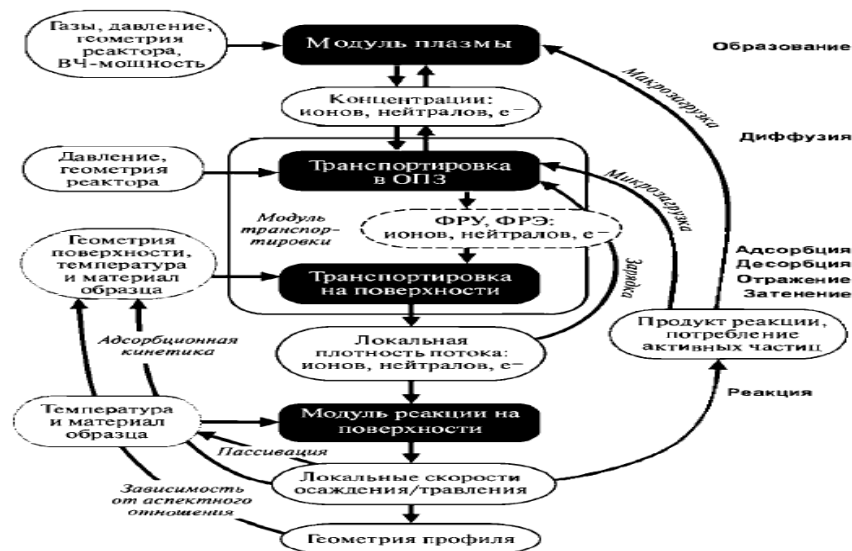
– неізотермическая плазма (плазма низкого давления) характеризуется тем, что средняя энергия (температура) электронов во много раз превышает энергию ионов и нейтральных частиц  $T_e \gg T_i \approx T_g$ .

– плазма атмосферного давления генерируется под нормальным давлением и пригодна к поточному применению.

Жидкостные процессы технологической обработки материалов в процессе создания микроструктур не могут обеспечить изготовления интегральных микросхем (ИМС) с субмикронными размерами, поэтому ведутся поиски альтернативных технологических вариантов на базе «сухих» процессов. Технологии с использованием низкотемпературной неравновесной газоразрядной плазмы в этом плане оказались наиболее перспективными. При сравнении плазменных технологических процессов с жидкостными можно выделить следующие их преимущества: увеличение разрешающей способности с уменьшением размеров элементов ИМС, обеспечение высокой селективности и анизотропии процессов, более высокий уровень безопасности работы и экологической чистоты производства; возможность непрерывного контроля хода и окончания процесса, возможность создания автоматизированных технологических циклов, не требующих участия оператора [1].

Модель процесса плазменного травления, взятая за основу в проводимых исследованиях, представлена на рис. [2]. Модель адаптирована к описанию операций, выполняемых при встраивании активных компонентов в подложку по технологии внутреннего монтажа [3].

Модуль плазмы, учитывающий внешние вводимые (входные) параметры, такие как рабочее давление, расход газов и газовых смесей, мощность на подложкодержателе и конструкция реактора, должен вычислять концентрации и энергии (температуры) образованных в плазме частиц (ионы, электроны, радикалы, нейтральные молекулы и атомы), которые являются входными параметрами модели. Полученные значения выходных параметров вводятся в модуль транспортировки, который включает в себя транспортировку частиц в области пространственного заряда (ОПЗ) или в плазме и транспортировку частиц по поверхности, как входные.



Рисунок

Модуль транспортировки в ОПЗ учитывает рабочее давление и конструкцию реактора и предсказывает, полагаясь на концентрации частиц в плазме и их температуру, функцию распределения по энергиям и по углам, и абсолютные потоки плазменных частиц на границе между ОПЗ плазмы и поверхностью образца.

Модуль транспортировки реагента по поверхности учитывает такие эффекты, как затенение, отражение или адсорбцию плазменных частиц вблизи образца и на его поверхности. Локальные расходы плазменных частиц – это входные параметры модуля реакции на поверхности.

Модуль реакции на поверхности описывает реакции плазменных частиц с поверхностью, учитывая материал и температуру пластины. Входными параметрами модуля являются локальные скорости травления или осаждения, которые впоследствии приводят к изменению геометрии поверхности. Геометрия поверхности представляет собой профиль травления, который является окончательным результатом моделирования процесса плазменного травления.

На правой стороне рисунка указаны три основных явления – образование, транспортировка (диффузия, дрейф, адсорбция, отражение, десорбция и затенение) и реакция, которые вносятся и определяются своим модулем моделирования. Существуют каналы обратной связи, по которым выходные параметры одного модуля влияют на граничные условия предыдущих модулей. Эти каналы обратной связи представлены обратными стрелками на рисунке.

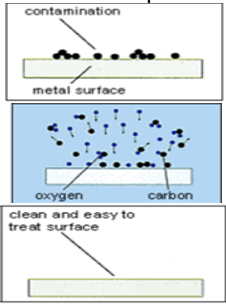
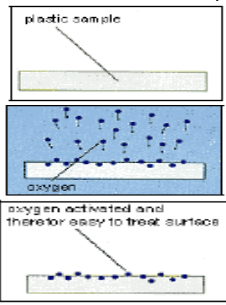
Рассмотрим применение плазм низкого давления в операциях формирования элементов микроструктур на подложках, пленках и кристаллах. Это операции обработки и удаления органических резистов, плазменного травления, плазменного получения резистивных пленок и слоев, плазменного осаждения покрытий (включая плазменное напыление и стимулированное плазмой осаждение из газовой фазы); плазменной модификации поверхности, очистки поверхности.

Благодаря содержащимся в газовой струе радикалам с помощью атмосферной плазмы можно проводить тонкую очистку поверхностей и их активацию. Атмосферную плазму применяют при работе с пластмассами, керамикой, стеклом, металлами и комбинированными материалами для выполнения склеивания, микросварки, сварки, пайки, печати и при нанесении покрытий.

На основе проведенной систематизации операций плазменной обработки выполнена их классификация по виду выполняемых операций, приведенная в табл. 1. Картинки, приведенные в столбце «Вид плазменной обработки», поясняются функциональными задачами, указанными в соседнем столбце.

Наиболее эффективные и перспективные виды плазменных операций представлены в табл. 2.

Таблица 1

Вид плазменной обработки	Функциональные задачи	Характеристики операций
<p>Очистка поверхности</p> 	<ul style="list-style-type: none"> <li>– Удаление жиров, масел, окислов и силиконов;</li> <li>– предварительная обработка перед микросваркой, пайкой или склеиванием;</li> <li>– предварительная обработка металла перед нанесением лакового покрытия</li> </ul>	<p>За счет ионной бомбардировки поверхность очищается физически, кроме того, возможно протекание химической реакции между газом и загрязняющим веществом. Частицы и молекулы загрязнителя переходят в газовую фазу и отсасываются вытяжным устройством</p>
<p>Плазменная активация</p> 	<ul style="list-style-type: none"> <li>– Предварительная обработка пластмассовых деталей перед склеиванием;</li> <li>– обработка пластика перед нанесением лакового покрытия;</li> <li>– предварительная обработка перед нанесением печати</li> </ul>	<p>Образование свободных радикалов на обрабатываемой поверхности за счет обработки ее с помощью кислорода приводит к формированию химически активного поверхностного слоя. Время обработки составляет порядка 1-5 мин. На такой активированной поверхности легче удерживаются лаковые и клеящие покрытия</p>

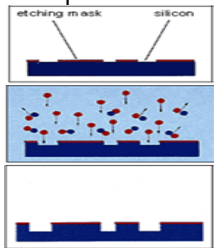
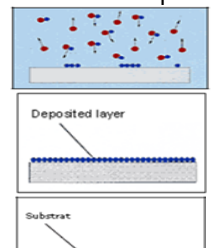
<p><b>Травление</b></p> 	<ul style="list-style-type: none"> <li>– Обработка синтетических материалов, эластомеров, полупроводников;</li> <li>– обработка высокотемпературных пластмасс (PFTE, PFA, FEP) перед склеиванием;</li> <li>– структурирование кремния</li> </ul>	<p>При длительной обработке поверхности (более 5 мин) поверхностный слой удаляется, что приводит к структурированию и увеличению поверхности, а это, в свою очередь, способствует более надежному сцеплению покрытия с поверхностью. Материал поверхности в местах обработки переходит в газовую фазу, и отсасывается насосом</p>
<p><b>Нанесение покрытий</b></p> 	<ul style="list-style-type: none"> <li>– Осаждение гидрофобного слоя;</li> <li>– осаждение гидрофильного слоя;</li> <li>– предотвращение диффузии</li> </ul>	<p>С помощью подбора необходимого газа (гексаметил-ди-силоксан, гексаметил-ди-силазан, диметиловый эфир тетраэтиленгликоля и т.п.) можно полимеризовать поверхность, осадив этот газ тонким слоем на поверхности</p>

Таблица 2

Вид операции	Описание операции
Предварительная обработка атмосферной плазмой	Является одной из самых эффективных плазменных технологий для очистки, активации или нанесения покрытия
Нанесения нано-покрытий для функционализации поверхностей	Функциональные покрытия наделяют поверхность абсолютно новыми эксплуатационными свойствами – антипригарными, сверхлипкими, огнеупорными, антибактериальными, с повышенной микротвердостью, высокостабильными и др. Возникают новые продукты, появляется возможность производить старые продукты на базе более экологичных и дешевых материалов
Плазменная модификация	Новый метод нанесения тонкослойных покрытий с высокой производительностью. Цель плазменной модификации - изготовление изделий со специальными свойствами поверхности - износостойкостью, антифрикционностью, коррозионностойкостью и др.
Плазменная закалка деталей и инструмента	Прогрессивный метод локального поверхностного упрочнения, многократно повышающий надежность и долговечность изделий
Плазменное напыление порошковых покрытий	Прогрессивный метод защиты деталей и конструкций от изнашивания и коррозии, восстановления размеров и декоративной обработки
Плазменная сварка деталей и изделий	Универсальный процесс получения надежных неразъемных соединений практически из любых металлов и сплавов, применяющихся в промышленности для сварных конструкций
Плазменно-дуговая наплавка порошковыми и проволочными материалами	Эффективный метод получения специальных свойств поверхностей и восстановления изношенных размеров деталей и изделий

Плазменные технологии позволяют электронной промышленности реагировать на постоянно растущие требования к размерной и функциональной точности выполняемых операций. Освоены технологии плазменной обработки в нанодиапазоне. При помощи нанотехнологии в зависимости от поставленных задач создается специфическое функциональное покрытие, проникающее в микроструктуру поверхности материала. В результате образуется высокоэффективный слой, наделяющий материал принципиально новыми свойствами. Создание целенаправленно функционализированных поверхностей означает абсолютно новые возможности для продук-

тов будущего, например в получении защитных покрытий для солнечных элементов, в разработке топливных элементов и облегченных конструкций из углеродного волокна.

Возможности плазменных технологий безграничны, это решение проблем избавления от опасных отходов, получение невероятно эффективного вид топлива, а также новых строительных материалов и много другого.

На основе выполненного анализа и классификации операций с использованием плазмы низкого давления предстоит проведение исследований по формированию элементов микро-структур с особыми функциональными задачами на полиимидных пленках и кристаллах.

### **Библиографический список**

1. *Ефремов А. М., Светцов В. И., Рыбкин В. В.* Вакуумно-плазменные процессы и технологии: учеб. пособие; ГОУВПО Иван. гос. хим.-технол. ун-т. Иваново, 2006. 260 с.
2. *Галперин В. А., Данилкин Е. В., Мочалов А.И.* Процессы плазменного травления в микро- и нанотехнологиях: учеб. пособие; под ред. С.П. Тимошенко. 3-е изд. (эл.). М.: БИНОМ. Лаборатория знаний, 2013. 238 с.: ил. (Нанотехнологии).
3. Конструктивно-технологические варианты проектирования электронных модулей на основе внутреннего монтажа / Ларин В.П., Шелест Д.К., Васильев С.А. и др.: Известия ГУАП. Аэрокосмическое приборостроение: науч. журнал /-СПб.: ГУАП, 2011, №1, с.112-116

**УДК 620.199**

**В. П. Кузьменко**

магистрант кафедры конструирования и технологий электронных и лазерных средств

**В. П. Пашков**

канд. техн. наук, доц. – научный руководитель

## **АНАЛИЗ ПРИЧИН ОТКАЗОВ ЭЛЕКТРОННЫХ УЗЛОВ**

Целью анализа причин отказов электронных узлов является прогнозирование безотказности по конструктивно-технологическим параметрам. Для достижения поставленной цели необходима разработка методики, которая должна будет позволять проводить оценку надежности объекта достаточно быстро и не требовать глубоких технических знаний.

Практически все современные многофункциональные радиоэлектронные средства построены на основе интегральных микросхем, которые выполняют не только управляющую роль, т. е. в них запрограммированы алгоритмы работы со служебными данными, но и отвечают за хранение и преобразование данных.

От безотказности таких функциональных единиц напрямую зависит работоспособность, сохранность и правильность служебной информации, потеря которой может быть критическим критерием эксплуатации всей системы.

Имеющиеся на данный момент аппаратно-программные средства контроля и анализа отказов интегральных микросхем (ИМС) не обеспечивают возможности полноценного автоматизированного управления, функционального контроля и диагностирования отказов и сбоев ИМС в реальном времени и непосредственно после испытаний.

В связи с этим актуальной является задача сокращения времени подготовки испытаний и функционального контроля ИМС без потери достоверности обнаружения функциональных отказов, а также оценка безотказности по конструктивно-технологическим параметрам.

Решение этой задачи основывается на исследовании закономерностей поведения ИМС в различных климатических условиях, анализе архитектурных особенностей, методик и аппаратно-программных средств при проведении испытаний.

Объект исследования – программируемые логические интегральные схемы (ПЛИС) относятся к классу сверхбольших интегральных схем, производятся по современным субмикронным технологиям (комплементарная структура металл-оксид-полупроводник –технология).

Плис благодаря своим достоинствам получили широкое применение в различных областях. Эта элементная база высокой степени интеграции позволяет разработчику за короткое время создать на основе стандартных компонент устройства обработки сигналов с высоким быстродействием и низким потреблением. На базе ПЛИС могут быть изготовлены логические блоки и устройства, например, преобразователи кодов, контроллеры, умножители, сигнальные процессоры с не очень сложными алгоритмами (вычислитель быстрого преобразования Фурье, различные фильтры и др.). ПЛИС относятся к полузаказным интегральным схемам, поскольку внутрисхемная топология частично формируется при производстве самих ПЛИС, а частично программируется в соответствии с требованиями потребителя. Программирование пользователем ПЛИС производится путем изменения внутренней структуры, т. е. соединения имеющихся на кристалле элементов в соответствии с требуемым алгоритмом. Настройка ПЛИС на заданный алгоритм и заданные спецификации производится программатором или с использованием САПР, что существенно уменьшает время проектирования.

Логика работы ПЛИС определяется путем дополнительного программирования с помощью специальных программаторов и программного обеспечения.

Микросхема ПЛИС состоит из:

- конфигурируемых логических блоков, реализующих требуемую логическую функцию;
- программируемых электронных связей между конфигурируемыми в схеме логическими блоками;
- программируемых блоков ввода/вывода, обеспечивающих связь внешнего вывода микросхемы с внутренней логикой.

Отказы таких ИМС в основном обусловлены наличием дефектов в исходных материалах, ошибками в проектировании, нарушениями технологических режимов при изготовлении, деградационными процессами вследствие физического старения конструктивных элементов, а также несоблюдением предельно допустимых режимов эксплуатации.

На надежность в значительной степени оказывает влияние качество исходных материалов. Такие факторы, как плотность и распределение структурных дефектов в полупроводниках, загрязнения исходных материалов, неправильное процентное соотношение химических компонентов, могут являться причиной изменения и ухудшения электрических и механических свойств готовых изделий.

Анализ отказов ПЛИС, обусловленных нарушением технологии изготовления, показывает, что основной причиной потери работоспособности, являются нарушения в межсоединениях. Различают два вида внутренних межсоединений в изделиях микроэлектроники: соединения внутренних компонентов между собой и с выводами корпуса, а также соединения между структурными элементами схемы, выполненными на кристаллах. Характерными нарушениями соединений внутренних компонентов между собой и с внешними выводами являются обрывы проводочных выводов, отрыв сварного или паяного соединения, касание проводочным выводом других компонентов конструкции изделия. Обрывы и короткие замыкания межсоединений на кристалле, возникают в большинстве случаев вследствие разрыва металлизации на ступеньках окисла, коррозионных разрушений, процессов электромиграции, механических повреждений токоведущих дорожек и дефектов фотолитографии.

Кроме того, достаточно часто к отказу ИМС вследствие нарушения соединения приводят дефекты внешних выводов, не позволяющие создавать надежные контакты выводов ИМС с элементами платы. Внешние выводы могут быть соединены с элементами печатной платы различными способами (сварные соединения, паяные соединения).

Часть отказов, связанных с разрушением внутренних соединений между элементами электронного компонента в процессе эксплуатации также можно отнести к отказам вследствие

физического старения изделия. Основные механизмы этих видов отказов связаны с деградацией механических свойств сварных и паянных соединений, а также самих проводников.

Наиболее часто встречающиеся ошибки при проектировании новых электронных компонентов связаны с пренебрежением такими дестабилизирующими факторами как деформация, термические и электрические перегрузки отдельных элементов, износ соединений.

Для повышения надежности разрабатываемого устройства большое внимание следует уделить предотвращению износа паяных соединений. Существует несколько причин износа паяных соединений. Некоторые из них очевидны, включая отсутствие надежного моделирования и расширенных испытаний. Помимо этого, тремя наиболее частыми причинами являются слишком близкое расположение источников тепла, избыточное рассеивание мощности и использование переходных колодок и соединителей.

Уменьшение размеров полупроводниковых приборов выдвигает все более высокие требования к долговечности диэлектрических и металлических пленок, служащих конструктивными элементами ПЛИС.

Среди деградационных процессов, оказывающих наибольшее влияние на интенсивность отказов ПЛИС вследствие физического старения можно выделить: диэлектрический пробой, электромиграцию, электролитические процессы, адсорбцию в оксиде, деградацию механических свойств, ионизацию, диффузию в металлизации, окисление металлизации, диффузию в объеме полупроводникового кристалла.

На всех технологических операциях вносятся внутренние механические напряжения как результат нагрева, легирования, нанесения и травления различных функциональных слоев. В поле механических напряжений резко увеличивается подвижность точечных дефектов, изменяется их равновесная концентрация, и дефектная структура эволюционирует в направлении формирования макроскопических дефектов, наличие которых резко ухудшает надежность ПЛИС.

Растягивающие напряжения могут приводить к образованию и росту пор в проводящей металлической пленке. Нарушение сплошности проводника в том месте, где перенос атомов вызывает истощение материала, приводит к разрыву электрической цепи, что является основным видом отказа ИМС. С другой стороны, сжимающие напряжения вызывают противоположное явление накопления материала в виде бугорков или наплывов с последующей экструзией металла. Накопление массы может вызывать разрушение изолирующего или защитного слоя, обуславливая дальнейшую коррозию металлической пленки.

Наиболее слабым звеном микросхем являются выводы кристаллов. Основной причиной разрушения внутренних выводов ИМС является возникновение напряжений при упругом смещении различных частей корпуса, например, при смещении металлического основания кристаллодержателя относительно стенок корпуса с прикрепленными к ней алюминиевыми или золотыми выводами. Разрушение внешних выводов в основном происходит в результате механических нагрузок и коррозии.

Так же осуществление замены компонентов неизбежно влечет за собой снижение надежности модуля.

Для схем большой степени интеграции определяющими могут оказаться отказы элементов и соединений внутри самого кристалла. Значительное число отказов происходит также из-за дефектов металлизации и контактов, дефектов посадки кристаллов и разварки внутренних выводов.

В Приложении 17 РД 110755-90 приведены весовые коэффициенты доминирующих видов отказов для интегральных микросхем при нормальных условиях эксплуатации и при повышенных температурах. При номинальных условиях эксплуатации (температура окружающей среды 25–65°C) самые большие доли в общем потоке отказов у следующих механизмов отказов: утечки по поверхности между диффузионными шинами, разрушение металлизации, пробой окисла.

Для температур окружающей среды 65 и 125°C по результатам ускоренных испытаний на безотказность доминируют механизмы отказов – зарядовая нестабильность в окисле и разруше-



ние металлизации. По результатам испытаний на долговечность – пробой окисла, дефекты кристалла и сварных соединений.

Для современных субмикронных топологических норм распределение весовых коэффициентов механизмов отказов в общем потоке отказов может измениться, то есть доминирующими являются другие отказы. При расчете энергии активации и показателей надежности следует учитывать весовые коэффициенты различных механизмов отказов.

Так же анализ радиационных испытаний плис ПЛИС показывает, что во многих случаях именно функциональные отказы являются доминирующими и определяют уровень радиационной стойкости ПЛИС. Это связано с функциональной сложностью микросхем и разнообразием блоков, входящих в её состав. Поэтому рекомендации по проведению испытаний и их методики должны включать в себя и радиационные испытания ПЛИС.

Анализируя рассмотренные характерные отказы ПЛИС и подводя итог, можно выделить следующую классификацию основных отказов по характерным признакам:

- по характеру изменения параметров объекта (постепенный, внезапный);
- по связи с отказами других объектов (зависимый, независимый);
- по стадии возникновения (конструкционный, производственный, эксплуатационный, деградационный);
- по устойчивости неработоспособности (устойчивые, неустойчивые, периодические);
- по способам обнаружения (явные, скрытые).

В процессе проведения ускоренных климатических испытаний ПЛИС возникает необходимость в случае появления отказа провести анализ причин возникновения отказа. Знание причин отказа помогает уточнять статистические данные о надежности ПЛИС и их составных конструкторско-технологических частей, более точно оценивать показатели отказов (подбирать энергии активации) а также вносить коррективы и рекомендации в методиках испытаний. Выявление причины возникновения отказа поможет разработать рекомендации по повышению надежности испытываемой ИМС, что в дальнейшем снизит интенсивность отказов, а так же позволит создать методику расчета показателей надежности ИМС.

### **Библиографический список**

1. *Богданов В. М., Ржевский В. Г., Слыхов А. А.* Испытания промышленной продукции. Ч. IV. Ускоренные испытания на надежность. М., 1980. 97 с.
2. *Ганиев Р. А.* Ускоренные испытания на надежность с использованием сильнодействующих факторов. Казань, 1985. 59 с.
3. *Строганов А. В.* Оценка долговечности БИС по результатам ускоренных испытаний // Технологии в электронной промышленности. 2007. № 3. С. 90–96.

**УДК 621.396.96**

**А. В. Леппик, Р. Р. Булякулов, Д. Д. Поляков, А. П. Суворов**

студенты кафедры бортовой радиоэлектронной аппаратуры

**А. Г. Охонский**

д-р техн. наук, профессор – научный руководитель

## **ИСПОЛЬЗОВАНИЕ ОБРАТНОГО СИНТЕЗИРОВАНИЯ АПЕРТУРЫ АНТЕННЫ ДЛЯ ДОСТИЖЕНИЯ ВЫСОКОЙ РАЗРЕШАЮЩЕЙ СПОСОБНОСТИ ПО УГЛУ**

Введение

Задача увеличения разрешающей способности радиолокационных систем актуальна уже не одно десятилетие. Как правило, этот параметр определяется отношением длины волны излучаемого колебания к диаметру раскрыва этой излучающей антенны. Ввиду этого факта, на мето-

ды улучшения разрешающей способности, связанные с уменьшением длины волны и увеличении размеров антенны, накладывает ряд ограничений. В связи с этим использование синтеза апертуры на данный момент – единственный способ увеличения разрешающей способности по углу при неизменных размерах антенны.

#### Описание метода

В соответствии с теорией пространственной селекции РЛС для обеспечения разрешения по угловой координате необходимо наблюдать цели при различных ракурсах, т.е. апертура антенны радиолокационной станции должна иметь определенный угловой размер относительно цели. Таким образом, синтезированная апертура в виде дуги радиусом  $R_n$  обеспечивает наблюдение цели последовательно во времени под различными ракурсами в пределах углового размера  $\beta_0 = VT_c / R_n$ . В этом случае реальная апертура увеличивает угловой сектор наблюдения (синтезирует апертуру) за счет облета цели с радиусом разворота  $R_n$  и постоянной линейной скоростью  $V$ . При малых угловых размерах синтезированной апертуры  $\beta_0$  разрешение по угловой координате  $l$  при работе антенны только на прием равно  $\delta l = \lambda / \beta_0$ . При использовании единой антенны на передачу и прием у синтезированной апертуры угловое разрешение увеличивается в два раза:  $\delta l = \lambda / 2\beta_0$  [1].

Очевидно, что изменение ракурса наблюдения цели возможно не только за счет перемещения антенны РЛС, но и за счет перемещения самой цели. При неподвижной РЛС угловой размер синтезированной апертуры в этом случае определяется скоростью перемещения цели  $V_c$  и временем синтеза  $T_c$ :  $\beta_0 = V_c T_c / R_n$ .

При равном угловом размере  $\beta_0 = V_c T_c / R_n = VT_c / R_n$  обеспечивается такое же разрешение, как и при синтезировании апертуры за счет движения приемо-передающей антенны РЛС:

$$\delta l = \frac{\lambda}{2\beta_0} = \frac{\lambda R_n}{2V_c T_c}.$$

Также очевидно, что изменение угла наблюдения цели возможно при неподвижных РЛС и цели за счет поворота цели. Угол поворота цели за время синтеза определяется угловой скоростью вращения цели  $\Omega_c$ :  $\beta_0 = \Omega_c T_c$ , что обеспечивает угловое разрешение

$$\delta l = \frac{\lambda}{2\beta_0} = \frac{\lambda}{2\Omega_c T_c}.$$

В отличие от других методов в случае синтеза апертуры за счет вращения цели разрешающая способность не зависит от дальности до цели при постоянном времени синтеза. Так, например, при угле поворота цели всего на  $3^\circ$  и длине волны 3 см достигается разрешение  $\delta l = 30$  см на любой дальности, на которой обеспечивается достаточное для обнаружения отношение сигнал/шум.

Высокое разрешение по азимуту при использовании классических алгоритмов обеспечить весьма сложно, поскольку для получения узкой диаграммы направленности (ДН) необходимы очень большие (практически нереализуемые) антенны. Решить эту проблему можно с помощью антенн с синтезированной апертурой, которые реализуются при относительном движении объектов и РЛС, имеющей реальную антенну с малым раскрытием. Синтез апертуры антенны возможен при любом виде движения объекта, источника облучающего сигнала и приёмного устройства. Важно лишь, чтобы при таком перемещении происходило изменение ракурса наблюдаемой цели. Синтезирование, основанное на использовании поступательного или вращательного движения объекта относительно неподвижной РЛС, получило название инверсного (Inverse Synthetic Aperture). Характерным примером использования обратного синтеза является получение радиолокационных портретов морских целей (кораблей) за счет их качки и

рыскания по курсу. В основе методов получения таких портретов лежит отражение сигналов от надстроек судов, определяемых их архитектурно–конструктивными типами [2].

Рассмотрим траекторный сигнал РЛС при обратном синтезировании апертуры. Фаза и задержка траекторного сигнала как основные источники информации о цели определяются изменением расстояния до элементов цели в процессе синтезирования апертуры. В общем случае расстояние изменяется вследствие перемещения цели относительно РЛС и вращения цели. При этом цель может одновременно вращаться в различных плоскостях с различной угловой скоростью.

Для частного случая получения изображения цели в плоскости «дальность  $r$  – азимут  $l$ », когда ось вращения цели перпендикулярна направлению на РЛС, текущее расстояние  $R_i$  от РЛС до  $i$ -го элемента цели при  $RH \gg \sqrt{r_i^2 + l_i^2}$  можно представить в виде

$$R_i = R_H - l_i \sin \Omega_{ц} t + r_i \cos \Omega_{ц} t - V_{цр} t,$$

где  $\Omega_{ц}$  – угловая скорость вращения цели;  $V_{цр}$  – радиальная скорость цели в направлении РЛС.

Соответственно фаза траекторного сигнала  $\varphi_i = \frac{4\pi * R_i}{\lambda}$ , а доплеровская частота

$$f_{дi} = \frac{2dR_i}{\lambda dt} = \frac{2l_i \Omega_{ц}}{\lambda} \sin \Omega_{ц} t - \frac{2r_i \Omega_{ц}}{\lambda} \cos \Omega_{ц} t - 2V_{цр} / \lambda,$$

В случае малых угловых размеров синтезированной апертуры, когда  $\beta_0 = \Omega_{ц} T_c \ll 1$ , без учета начальной фазы

$$\varphi_i(t) = \frac{4\pi}{\lambda} \left( l_i \Omega_{ц} t - \frac{r_i * \Omega_{ц}^2 * t^2}{2} \cos \Omega_{ц} t - V_{цр} t \right),$$

$$f_{дi}(t) = \frac{2l_i \Omega_{ц}}{\lambda} - \frac{2r_i \Omega_{ц}^2}{\lambda} t - 2V_{цр} / \lambda,$$

Таким образом, вращение цели формирует траекторный сигнал с линейной частотной модуляцией (ЛЧМ). Постоянная доплеровская частота  $2V_{цр} / \lambda$  образуется вследствие радиального перемещения одновременно всех элементов цели относительно РЛС. Обычно производится оценка и компенсация этой частоты в сигнале.

Доплеровская частота, равная  $\frac{2l_i \Omega_{ц}}{\lambda}$ , образуется в результате линейной скорости движения  $i$ -го элемента в направлении РЛС при вращении цели. Разрешение сигналов по этой доплеровской частоте обеспечивает разрешение элементов цели по азимуту  $S$ . Линейное изменение частоты сигнала  $i$ -го элемента цели определяется координатой дальности этого элемента  $g_i$  относительно центра вращения цели. Разрешение элементов цели по частотной модуляции траекторных сигналов при малом размере синтезирования апертуры невелико. Поэтому разрешение по дальности обеспечивается модуляцией зондирующего сигнала. При этом в алгоритме обработки траекторного сигнала необходимо учитывать изменение как частоты сигнала, так и его задержки:

$$\Delta \tau_j = \frac{2l_i \Omega_{ц} t}{c},$$

Эти изменения можно не учитывать, если допустить, что изменение частоты и задержки за время синтезирования меньше, чем разрешение по частоте и задержке. В этом случае размер зоны обзора  $\Delta R \times \Delta L$  и разрешение  $\delta r = \delta l$  будут связаны следующим условием:

$$\Delta R = \Delta L \leq 4 * \delta * l_{12} / \lambda,$$

При таких ограничениях алгоритм обработки траекторного сигнала при обратном синтезировании сводится к доплеровской фильтрации в каждом канале, разрешаемом по дальности. Координата элемента цели по дальности определяется задержкой сигнала  $l_i = ct_i/2$ , а координата азимута – доплеровской частотой  $f_{di} = \lambda f_{\Omega_{\text{ц}}}$ . Масштаб изображения цели по азимуту определяется скоростью вращения цели  $\Omega_{\text{ц}}$  [3].

Полоса доплеровских частот траекторного сигнала и, следовательно, требуемая частота повторения зондирующих импульсов определяются размером цели (зоны обзора) по азимуту:

$$\Delta f_{\text{д}} = \frac{2\Omega_{\text{ц}}\Delta l}{\lambda},$$

В общем случае результирующий вектор вращения цели  $\Omega_{\text{р}}$ , образованный одновременным перемещением цели в различных плоскостях, не перпендикулярен направлению наблюдения. Тогда вектор скорости вращения цели  $\Omega_{\text{ц}}$ , определяющий масштаб и положение плоскости изображения цели ( $l, r$ ) в пространстве, является проекцией результирующего вектора  $\Omega_{\text{р}}$  на плоскость, нормальную к вектору направления наблюдения  $R_{\text{н}}$ . В этом случае плоскость изображения ( $l, r$ ) нормальна плоскости, в которой расположены векторы вращения  $\Omega_{\text{р}}$ ,  $\Omega_{\text{ц}}$  и направление наблюдения  $R_{\text{н}}$ . Масштаб изображения цели по азимуту, как и ранее, определяется угловой скоростью вращения цели [4]:  $\Omega_{\text{ц}} = \Omega_{\text{р}} \sin \phi$ , где  $\phi$  – угол между векторами  $\Omega_{\text{р}}$  и  $R_{\text{н}}$ .

Применение метода для достижения высокой разрешающей способности по углу

Одновременно с перемещением по курсу при волнении при волнении моря корабль испытывает также колебания корпуса вокруг центра масс. Для задач обратного синтезирования обычно использует рыскание по курсу, килевую и бортовую качку корабля. Рыскание по курсу соответствуют вращению корабля относительно вертикальной оси. Килевая качка (попеременный дифферент на нос и на корму) соответствует вращению корабля относительно поперечной горизонтальной оси. Попеременный крен (бортовая качка) на левый и правый борт соответствует вращению корабля относительно продольной горизонтальной оси.

Вращение корабля (качка, рыскание) носят периодический характер, т.е. угол отклонения корабля от равновесного положения изменяется в соответствии с гармоническим законом:

$$\beta = \beta_{\text{макс}} \sin\left(\frac{2\pi}{T_k} t\right),$$

где  $\beta$  – соответственно угол крена, рыскания или угол килевой качки;  $\beta_{\text{макс}}$  – максимальное отклонение корабля по соответствующему углу,  $T_k$  – период колебаний.

Угловая скорость вращения изменяется по гармоническому закону:

$$\Omega = \frac{d\beta}{dt} = \frac{2\pi}{T_k} \beta_{\text{макс}} \cos\left(\frac{2\pi}{T_k} t\right).$$

У большинства океанских кораблей период качки  $T_k$  практически не зависит от степени волнения моря и определяется конструкцией корабля. Амплитуда угловых колебаний  $\beta_{\text{макс}}$  определяется многими факторами: высотой волн, направлением бега волн относительно курса корабля, скоростью движения и конструкцией корабля. Большой корабль как колебательная система эквивалентен узкополосному фильтру, и параметры его колебаний можно считать постоянными за время синтезирования порядка долей секунд. С уменьшением водоизмещения корабля (менее 1000т) амплитуда и период отклонения уже зависят от характеристик волнения моря и носят случайный характер. Обычно считают, что угол  $\beta$  является узкополосным нормальным процессом [5].

Оценочные значения характеристик колебаний кораблей различного типа при волнении моря 5–6 баллов приведены в работе [6].

Если линия визирования корабля перпендикулярна оси угловых колебаний, возможно получение изображения корабля в различных плоскостях. Килевая качка обеспечивает получение изображения вдоль корабля и его надстроек, качка по крену – изображение в боковом направлении корабля надстроек, рыскание по курсу – изображение корабля в горизонтальной плоскости. Движение корабля с постоянной скоростью и курсом эквивалентно движению РЛС при неподвижном корабле и обеспечивает изображение в горизонтальной плоскости. В реальной ситуации одновременно существуют все виды колебаний корабля, что затрудняет определение положения плоскости изображения относительно корабля. В то же время, наблюдая «динамическое» изображение корабля, т.е. изображение, изменяющееся в зависимости от фазы колебаний корабля во время синтезирования, можно эффективно распознавать его класс.

Рассмотрим характеристики сигнала при обратном синтезировании в случае бортовой и продольной качки корабля. При бортовой качке изменение  $r_h$  от надстройки на высоте  $h$  до РЛС определяется выражением:

$$r_h(t) = h \sin(\beta t) = h \sin\left(\beta_{\text{макс}} \sin\left(\frac{2\pi}{T_k} t\right)\right),$$

где  $h$  – высота элемента надстройки, отсчитываемая от оси вращения корабля. Учитывая, что угловое отклонение  $\beta_{\text{макс}} \ll 1$  (в радианах), измерение расстояния до РЛС и, следовательно, фазы отраженного сигнала можно представить в виде

$$r_h(t) = h\beta_{\text{макс}} \sin\left(\frac{2\pi}{T_k} t\right),$$

$$\phi(t) = \frac{4\pi}{\lambda} r_h(t) = \frac{4\pi}{\lambda} h\beta_{\text{макс}} \sin\left(\frac{2\pi}{T_k} t\right).$$

Соответственно доплеровская частота равна:

$$f_d(t) = \frac{4\pi h \beta_{\text{макс}}}{\lambda T_k} \cos\left(\frac{2\pi}{T_k} t\right).$$

Как видно из двух предыдущих выражений, характеристики отраженного сигнала зависят от момента наблюдения  $t=t_0 \pm T_c/2$ . Так при  $t_0=0$  корабль наблюдается в момент прехода угла  $\beta(t)$  через ноль, когда угловая скорость отклонения максимальна:

$$\Omega_{\text{макс}} = \frac{2\pi}{T_k} \beta_{\text{макс}} h.$$

При малом времени синтезирования  $T_c \ll T_k$  доплеровская частота сигнала при  $t_0=0$  изменяется незначительно и каждому элементу надстройки по высоте соответствует своя доплеровская частота

$$f_d = \frac{4\pi}{\lambda T_k} \beta_{\text{макс}} h.$$

В случае  $t_0=0$  разрешение по высоте надстройки будет достаточно большим. Учитывая, что изменение угла  $\beta$  за время синтезирования

$$\beta_0 = \Omega_{\text{макс}} T_c = \frac{2\pi}{T_k} \beta_{\text{макс}} T_c,$$

Получим разрешение по высоте надстройки корабля

$$\delta h = \frac{\lambda}{2\beta_0} = \frac{\lambda T_k}{4\pi \beta_{\text{макс}} T_c}.$$

Алгоритм обработки сигналов сводится к доплеровской фильтрации в каждом элементе разрешения по наклонной дальности. Полученные зависимости справедливы и при килевой кач-

ке корабля и радиолокационном наблюдении в передней зоне обзора РЛС на встречных курсах самолета – носителя РЛС и корабля. Разрешение по доплеровской частоте в этом случае соответствует разрешению по высоте надстройки  $h$ , а разрешение по задержке зондирующего импульса соответствует разрешению вдоль корабля. При наблюдении крейсера разрешение по надстройке  $\delta h = 90 \text{ см}$  при  $T_k = 6 \text{ с}$ ;  $\beta_{\text{макс}} = 2^0$ ,  $T_c = 0,5 \text{ с}$ ,  $\lambda = 3 \text{ см}$ .

При увеличении времени синтезирования начинают сказываться изменения доплеровской частоты и задержки сигнала, что необходимо учитывать в алгоритме обработки. Предельное разрешение без учета изменений задержки и частоты ограничено величиной  $(\delta h)^2 \geq \frac{\lambda h_{\text{макс}}}{4}$ . Для других моментов наблюдения ( $t_0 \neq 0$ ) при постоянном времени синтезирования разрешение будет ухудшаться.

Перемещение цели, так же как и движение носителя РЛС, создает эффект синтезирования апертуры, угловой размер которой определяется взаимным угловым перемещением РЛС и цели. Покажем это на примере разрешения элементов групповой цели, состоящей из двух синхронно движущихся малоразмерных объектов (точечных целей).

Доплеровская частота траекторного сигнала при взаимном перемещении РЛС с постоянной скоростью  $V$  и цели 1 также с постоянной за время синтезирования скоростью  $V_{ц1}$ :

$$f_{ц1} = 2(V \cos \theta_{н1} + V_{ц1} \cos \alpha_{ц1}) / \lambda.$$

Для цели 2, смещенной по углу на величину  $\Delta \theta \ll 1$ , доплеровская частота изменяется так, что разница частот сигналов двух целей, движущихся с одинаковой скоростью, составляет

$$\Delta f_{ц} = 2(V \sin \theta_{н2} - V_{ц2} \sin \alpha_{ц2}) \Delta \theta / \lambda = 2(V_{т+} - V_{ц2}) \Delta \theta,$$

где  $V_{т+}$  и  $V_{ц2}$  – соответственно тангенциальные составляющие скорости РЛС и целей.

Разрешение сигналов двух целей по доплеровской частоте  $\delta f_{д} = 1/T_c$  определяет разрешение целей по углу  $\Delta \theta$  при  $\Delta f_{ц} = \delta f_{д}$ :

$$\delta \theta = \lambda / (2(V_{т+} - V_{ц2}) T_c).$$

Соответственно линейное разрешение по углу определяется формулой:

$$\delta l = \delta \theta R_n = \lambda / (2 \beta_0),$$

определяется, как и ранее, угловым размером синтезированной апертуры  $\beta_0 = \Omega T_c$ , где  $\Omega$  – угловая скорость вращения линии визирования «РЛС – цель».

Выводы

Разрешающая способность по углу при синтезировании апертуры определяется угловым размером апертуры, который формируется путем относительного перемещения РЛС и цели. Формирование синтезированной апертуры при перемещении (вращении) цели и неподвижной РЛС называется обратным, или инверсным, синтезированием.

При обратном синтезировании полагают, что цель движется как единый объект и параметры отраженного сигнала определяются характеристиками движения цели. Фаза отраженного сигнала РСА при вращении цели с постоянной скоростью изменяется по гармоническому закону. При малых угловых размерах синтезирования формируется траекторный сигнал с линейной частотной модуляцией. Средняя доплеровская частота сигнала разрешаемого элемента цели пропорциональна его угловой координате, а индекс частотной модуляции – координате дальности. Разрешающая способность по углу определяется временем синтезирования и угловой скоростью вращения цели в плоскости наблюдения «РЛС – цель», а получение высокого разрешения по дальности обеспечивается модуляцией зондирующего сигнала. При обратном синтезировании по морским целям используется вращение (колебания) корабля: бортовая и килевая качка и рыскание по курсу. Угол отклонения корабля от нормального положения изменяется по гармоническому закону. Период колебаний определяется в основном типом корабля, а амплитуда колебаний – степенью волнения моря. При использовании колебаний корабля для обратного синтезирования возможно получение трехмерного изображения корабля с высоким разрешением по азимуту, углу места и дальности. Так как параметры колебаний корабля (векторы угловых скоро-

стей) неизвестны, требуется адаптивная обработка траекторного сигнала РСА. Обратное синтезирование возможно также при использовании линейного перемещения цели (без вращения). Разрешение по азимуту в этом случае определяется временем синтезирования и угловой скоростью вращения линии визирования «РЛС – цель». При построении радиолокационного изображения целей в алгоритмах компенсации движения с точки зрения помехоустойчивости целесообразно использовать метод преобразования Радона для выравнивания по дальности и метод минимума энтропии для фазовой настройки

### Библиографический список:

1. Кондратенков Г. С., Фролов А. Ю. Радиовидение. Радиолокационные системы дистанционного зондирования земли. М.: Радиотехника. 2005.
2. Орлов В.М., Шустиков В.Ю., Нефедов С.И. Моделирование инверсно синтезированных радиолокационных изображений движущихся объектов на основе фацетных моделей // Антенны. 2005. №10 (101). С. 69–75.
3. Орлов В.М. Методы компенсации движения в РЛС с инверсным синтезированием апертуры (ИСА) // Радиолокация, навигация, связь: Труды XVII международной научно–технической конф. Воронеж. 2011. Т. 3. С. 2331–2342.
4. Пасмуров А.Я. Получение радиолокационных изображений летательных аппаратов // Радиотехника и электроника. 1987. Т. 32, №12. С. 3–30.
5. Нефедов С.И. Вопросы перспективной радиолокации. Коллективная монография / Под ред. А.В. Соколова // Перспективы повышения эффективности метровых РЛС при использовании длительного когерентного накопления (по результатам экспериментальных исследований) М.: Радиотехника, 2003. С. 49–63.
6. Детков Н.И., Макаров И.А. Модель движения космического объекта и формирование его радиолокационного портрета / ОАО «НПК НИИДАР», г. Москва
7. Методы обратного (инверсного) синтезирования апертуры <http://www.tehnoinfra.ru/radiovidenie/43.html>

УДК 535.015

**А. А. Орлов**

студент кафедры конструирования и технологий электронных и лазерных средств

**О. Д. Москалец**

доц., канд. техн. наук – научный руководитель

## КОРРЕЛЯЦИОННЫЙ АНАЛИЗ В РАДИО И ОПТИЧЕСКОМ ДИАПАЗОНЕ

Измерение спектров оптических сигналов является одним из важнейших видов оптических измерений, а оценка энергетического спектра стационарного случайного процесса представляет собой важнейший раздел теории и практики статистических измерений.

Существует несколько способов получения оценки энергетического спектра: с помощью предварительно определенной оценки корреляционной функции; на базе обработки мгновенного спектра; с помощью метода фильтрации. Данная работа посвящена реализации последнего метода с помощью многоканальной узкополосной системы в форме многоканального спектрометра оптического диапазона (рис. 1). Такой метод получения оценки представляется идеальным в классе получения оценок методом фильтрации [1], [2].

Этот коррелятор может рассматриваться как система, включающая следующие блоки:

– блок формирования монохроматического волнового пучка с плоским фронтом. Этот блок содержит лазер PS и формирующую оптику – линзы и три слоя свободного пространства;

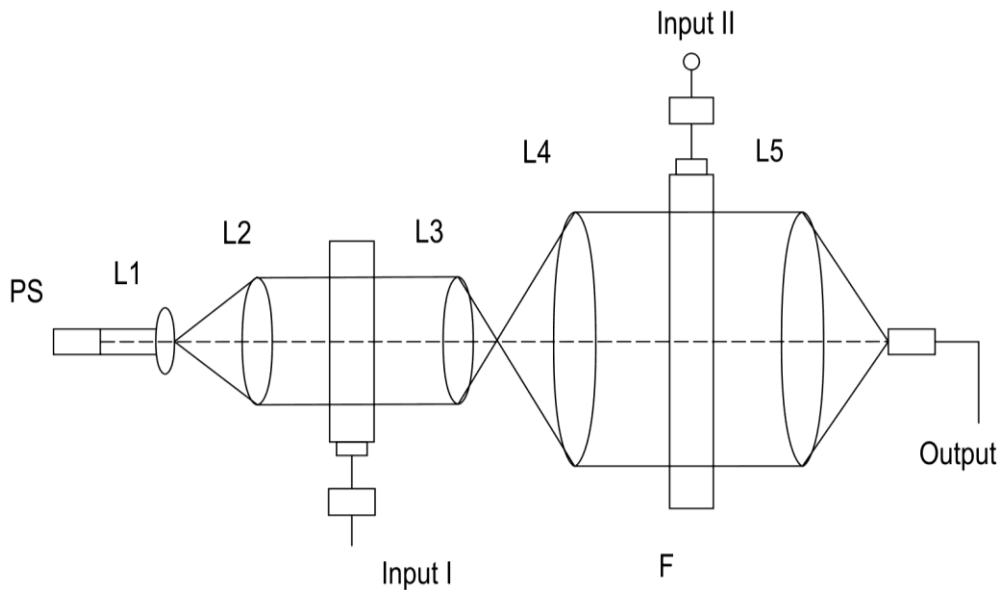


Рис. 1. Структурная схема многоканального резонаторного спектрометра оптического диапазона

– первый акустооптический анализатор спектра радиосигналов [3], этот блок содержит акустооптический модулятор, на вход которого (input I) подается один из коррелируемых сигналов и оптический когерентный Фурье – процессор, включающий линзу и два слоя свободного пространства;

– второй акустооптический анализатор спектра радиосигналов, этот блок содержит акустооптический модулятор, на вход которого (input II) подается второй коррелируемый сигнал и оптический когерентный Фурье – процессор, включающий линзу, и два слоя свободного пространства с дальнейшей системой увеличения с помощью линзы;

– фотодетектор с интегрированием в области пространственных частот;

– полосовой фильтр, на выходе которого формируется функция взаимной корреляции;

$$R(\tau) = \int_{-\infty}^{\infty} s_1(t) s_2(t - \tau) dt. \quad (1)$$

Предложенное разбиение акустооптического коррелятора на блоки позволяет сделать заключение, что вычисление функций взаимной корреляции сигналов, подаваемых на входы input I и input II, осуществляется на основе вычисления мгновенных спектров этих сигналов и дальнейшей их обработки. При этом понятие мгновенного спектра требует уточнения.

Значение акустооптического анализатора спектра радиосигналов при выполнении корреляционных измерений методами акустооптики требует отдельного рассмотрения основных моментов действия этого спектрального прибора.

Показано, что действие коррелятора может быть описано как соответствующее преобразование мгновенных комплексных спектров коррелируемых импульсных радиосигналов, т.е. в форме

$$I = \int_{-\infty}^{\infty} S_1(\omega, t) S_2(\omega, t) d\omega, \quad (2)$$

где  $S_1(\omega, t)$ ,  $S_2(\omega, t)$  – мгновенные спектры коррелируемых сигналов в общем виде. В свою очередь мгновенный комплексный спектр определяется выражением

$$\tilde{S}_r(\omega, t) = \int_{-\infty}^{\infty} r(\tau) s(\tau - t) \exp(-i\omega\tau) d\tau, \quad (3)$$



где  $r(t)$  – неподвижное временное окно, относительно которого перемещается анализируемое колебание  $s(t)$ , в частности  $r(t)$  – характеристическая функция промежутка  $[-T/2, T/2]$ .

С учетом выражения (3) соотношение (2) приводится к форме

$$A^2 \int_{-\infty}^{\infty} S_1(\omega') S_2^*(\omega') \exp(i\omega'\tau) d\omega' = R_{12}(\tau), \quad (4)$$

где  $R_{12}(\tau)$  – взаимная корреляционная функция  $\tau = 2t$ .

Соотношение (4) для функции взаимной корреляции является основой обобщенной эквивалентной схемы вычисления функции взаимной корреляции, которая позволяет выполнить дальнейшие исследования акустооптического коррелятора (рис. 2).

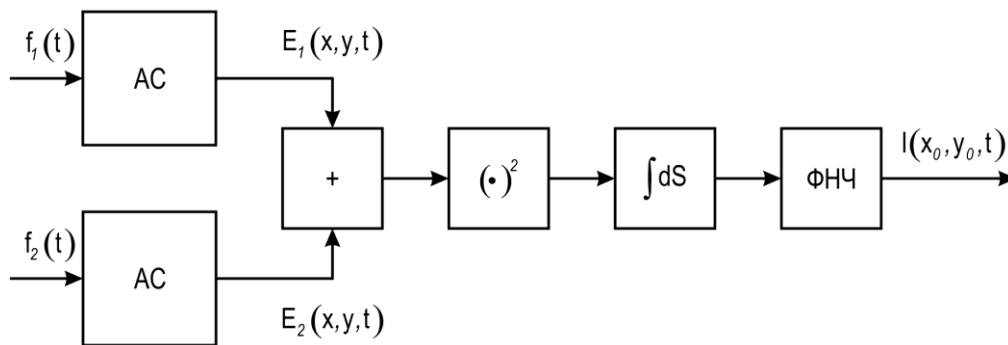


Рис. 2. Эквивалентная схема акустооптического коррелятора

Акустооптические корреляторы с пространственным интегрированием применяются для обработки импульсных сигналов и представляют собой согласованные фильтры. Опорный сигнал формируется с помощью кодирующего транспаранта. Входной сигнал вводится с помощью акустооптического модулятора. Фотоприемник осуществляет пространственное интегрирование мгновенных спектров, формируемых на его светочувствительной площадке. На входы анализаторов спектра (АС) подаются входной  $f_1(t)$  и опорный  $f_2(t)$  сигналы, соответственно. Выходное распределение светового потока после анализаторов можно записать как  $E_1(x, y, t)$  и  $E_2(x, y, t)$ . Эти световые потоки попадают на фотоприемник, где складываются. Фотоприемник мгновенного действия осуществляет пространственное интегрирование по своей светочувствительной площадке интенсивности светового поля, которая пропорциональна квадрату светового поля. Фильтр низких частот (ФНЧ) отражает частотные свойства фотоприемника и может иметь частоту среза порядка сотен МГц.

В общем случае частоты световых полей  $E_1(x, y, t)$  и  $E_2(x, y, t)$  не совпадают, так как опорный сигнал может представлять собой стационарную дифракционную решетку и его частота будет соответствовать частоте входного светового поля. Входной информационный сигнал вводится в коррелятор с помощью акустооптического модулятора и после дифракции частота светового поля отличается от входного на величину, соответствующую средней частоте полосы пропускания акустооптического модулятора.

Проведенные исследования позволили установить на основе мгновенных спектров свертываемых и коррелируемых сигналов общий алгоритм вычисления корреляционных функций и предложить эквивалентную схему акустооптического коррелятора и выполнить его дополнительные исследования. Этот алгоритм ведет свое начало из характерной особенности акустооптических устройств обработки информации, в частности корреляторов и анализаторов спектров радиосигналов. Эта особенность заключается в том, что коррелируемые и анализи-

руемые динамические сигналы в форме колебательных явлений в процессе обработки в результате акустооптического взаимодействия переносятся из радиодиапазона в оптический диапазон и далее обрабатываются как волновые процессы оптическими системами.

### Библиографический список

1. А. с. 374633 СССР, МКИ G 06 G 9/00. Оптико–акустическое устройство.
2. Lambert L.B. Wide-band instantaneous spectrum analyzers employing delay – line light modulators: IRE International Conv., 1962. Часть 6. 69–78 с.
3. Кулаков С.В., Москалец О.Д., Разживин Б.П. Обобщенная структурная схема акустооптического устройства корреляционной обработки// Акустооптические методы обработки информации: Сб. науч. тр./ Ленинград: Наука Ленинградское отделение, 1978. С. 4–12.

УДК 535-45.53.06

**Б. В. Осколков**

студент кафедры конструирования и технологий лазерных и электронных средств

**О. М. Филонов**

доцент, канд. техн. наук – научный руководитель

## ФОТОВОЛЬТАИЧЕСКИЕ ЭФФЕКТЫ КАК ФИЗИЧЕСКАЯ ОСНОВА ДЛЯ СОЗДАНИЯ НОВОГО ПОКОЛЕНИЯ ДАТЧИКОВ

В настоящее время основные требования к электронным компонентам – это их минимизация, простота и эффективность. Постоянно повышающиеся требования к этим параметрам могут удовлетворить датчики, построенные на основе фотовольтаических эффектов.

В статье будут рассмотрены различные фотовольтаические эффекты, допускающие техническую реализацию датчиков без внесения принципиальных изменений в существующие технологии. Показано, что датчики на фотовольтаических эффектах являются высокотехнологичными изделиями, что обеспечивается, предельной простотой конструкции технологического маршрута их изготовления.

Основные проблемы, которые потребуют значительных усилий со стороны разработчиков и конструкторов этих изделий, скорее всего, будут связаны с обработкой выходного сигнала и повышением чувствительности датчика к измеряемым физическим величинам.

**ЭФФЕКТ ДЕМБЕРА.** Если полупроводник осветить светом с длиной волны, соответствующей области собственного поглощения:  $\hbar\omega = E_n$  ( $E_n$  – ширина запрещенной зоны полупроводника), то вблизи поверхности образца, где происходит поглощение света, возникнут электронно-дырочные пары. Эти носители заряда диффундируют (рис. 1) из освещенной области в глубь полупроводника. Поскольку электроны обладают большей подвижностью, чем дырки, то электроны продвигаются дальше в глубь кристалла, чем дырки. Эта разница в диффузии носителей заряда приведет к тому, что поверхность полупроводника окажется положительно заряженной по отношению к его объему. Возникшее электрическое поле направлено так, что оно ускоряет дырки, имеющие меньшую подвижность, и замедляет подвижные электроны, поэтому суммарный ток равен нулю. Объемная ЭДС, возникающая в освещенном полупроводнике вследствие различия в коэффициентах диффузии электронов и дырок, получила название ЭДС Дембера.

Величина ЭДС Дембера в отсутствие ловушек и без учета поверхностной рекомбинации определяется формулой

$$U = -\frac{D_n - D_p}{\mu_n + \mu_p} \int_0^l \frac{d\sigma}{\sigma}, \quad (1)$$

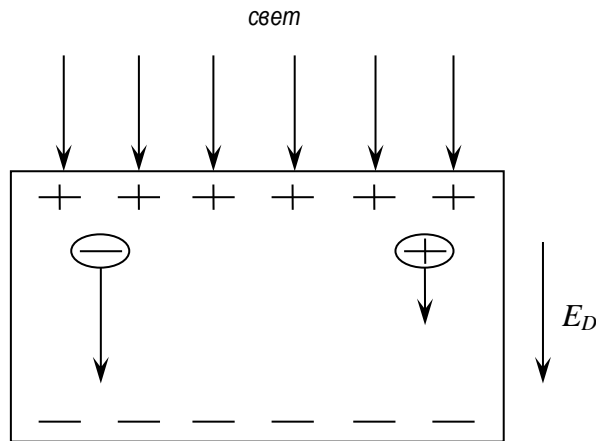


Рис.1. Возникновение ЭДС Дембера

где  $D_n$  – коэффициент диффузии электронов;  $D_p$  – коэффициент диффузии дырок;  $\mu_n$  – подвижность электронов;  $\mu_p$  – подвижность дырок;  $l$  – расстояние от освещаемой поверхности до места, где уже нет неравновесных носителей;  $\sigma$  – проводимость системы.

Вводя обозначение  $b = \frac{\mu_n}{\mu_p}$ , учитывая соотношение Эйнштейна  $eD_{n,p} = \mu_{n,p}k_B T$  и беря интеграл, получим

$$U = \frac{kT}{e} \frac{b-1}{b+1} \ln \frac{\sigma(0)}{\sigma(l)} \quad (2)$$

Из формулы (2) следует, что ЭДС Дембера тем больше, чем сильнее различаются подвижности электронов и дырок. ЭДС Дембера обычно очень мала, она немного больше  $kT/e$  (порядка 26 мВ) [1].

**ЭФФЕКТ КИКОИНА-НОСКОВА.** Возникновение электрического поля (ЭДС) в полупроводнике, помещенном в магнитное поле, при освещении его сильно поглощаемым светом получило название фотомагнитоэлектрического эффекта.

Когда на полупроводник падает свет, частота которого соответствует собственному поглощению:  $\hbar\omega = E_n$  ( $E_n$  – ширина запрещенной зоны полупроводника), то в тонком поверхностном слое образуется высокая концентрация электронов и дырок. Возникающий при этом градиент их концентрации приводит к появлению диффузионного потока носителей в направлении падающего излучения. Если магнитное поле  $H$  приложено вдоль оси  $Oz$  (рис. 2), световой пучок и диффузионный поток – вдоль оси  $Oy$ , то магнитное поле отклоняет электроны и дырки в разные стороны, вызывая в направлении  $Ox$  пространственное разделение зарядов. Если концы образца замкнуты, то в цепи возникает ток  $i_x$ , если разомкнуты, то – фото-ЭДС (рис. 3).

В слабых магнитных полях ЭДС эффекта Кикоина-Носкова пропорциональна магнитному полю  $H$  и меняет знак при изменении направления  $H$  на противоположное (нечетный фотоэффект.).

В слабых магнитных полях ( $uH/c \ll 1$ , где  $u$  – подвижность носителей заряда) плотность тока  $i_x = -eD \frac{H}{c} (u_n + u_p) \frac{dn}{dy}$ ,

где  $D$  – коэффициент амбиполярной диффузии носителей заряда;  $n$  – концентрация неравновесных носителей заряда. Вид распределения  $n$  в общем случае сложен, поскольку зависит от

диффузионно-рекомбинационных параметров полупроводника, от коэффициента поглощения света и квантового выхода фотогенерации носителей.

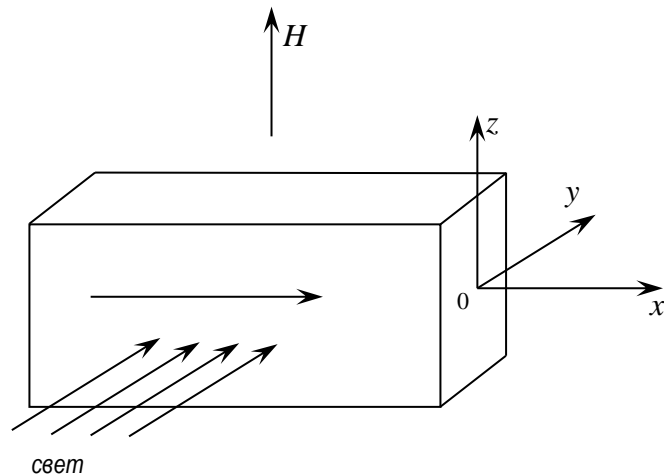


Рис.2. Образец для измерения фотомагнитоэлектрического эффекта

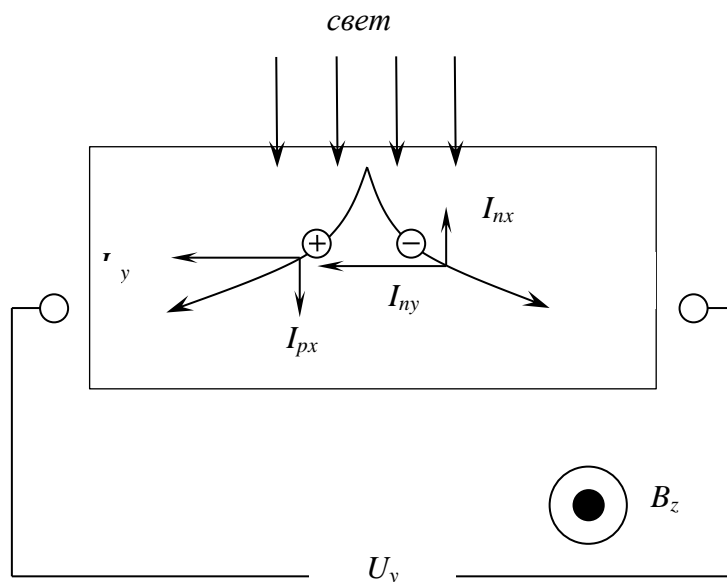


Рис. 3. Возникновение напряжения  $U_x$  при фотомагнитоэлектрическом эффекте

В монокристаллических полупроводниках эффект Кикоина-Носкова анизотропен - величина и знак ЭДС зависят от взаимной ориентации кристаллографических осей и  $H$ . Поскольку анизотропия связана с анизотропией коэффициента диффузии  $D$  носителей в присутствии магнитного поля, то ее исследование позволяет определить эффективные массы электронов и дырок вдоль различных кристаллографических осей образца.

На основе фотомагнитоэлектрического эффекта созданы простые и надежные методы определения таких параметров полупроводников, как время жизни неравновесных носителей заряда, диффузионная длина, скорость поверхностной рекомбинации, а также детекторы излучения и магнитометры.

Уровень напряжения этого эффекта [2] достигает 10 мВ.

**ФОТОПЬЕЗОЭЛЕКТРИЧЕСКИЙ ЭФФЕКТ.** Суть фотопьезоэлектрического эффекта состоит в возникновении ЭДС из-за разницы ширины запрещенной зоны в объеме освещенного полупроводника, которая меняется под воздействием деформации.

Если ширина запрещенной зоны  $E_G$  различна по всей длине полупроводника и в самом начале освещенной зоны, это значение будет равно  $E_{Gb}$ , а в конце –  $E_{Gc}$  величина ЭДС будет определяться следующим равенством:

$$U = -\frac{1}{e} \Delta t_1 (E_{Gc} - E_{Gb}),$$

где  $e$  – заряд электрона;  $\Delta t_1 = t_1 - t_{10}$ ;  $t_1 = \frac{\sigma_1}{\sigma}$ ;  $t_{10} = \frac{\sigma_{10}}{\sigma_0}$ ;  $\sigma$  – полная проводимость во время освещения;  $\sigma_0$  – полная проводимость в темноте;  $\sigma_1$  и  $\sigma_{10}$  соответственно проводимости электронов в зонах проводимости.

Исследования[3] на образцах германия, кремния и арсенида галлия показывают, что импульсное оптическое облучение вызывает переменную фото-ЭДС в объеме кристалла и через обратный пьезоэлектрический эффект порождает упругие колебания на частоте продольного механического резонанса. Переменная деформация растяжения сжатия пластины приводит к возникновению импульсов ЭДС на ее противоположных поверхностях за счет поперечного пьезоэффекта.

Описанный эффект может быть использован для селективного приема оптических сигналов с использованием как объемных, так и поверхностных акустических волн.

Описанные выше эффекты объединяет одно условие: наличие освещенного участка полупроводника. Датчики, созданные на основе данных эффектов, обладают необычайно высокой технологичностью, так как заготовки из полупроводникового материала требуют минимальной обработки для использования фотовольтаических эффектов. Единственная проблема – малый уровень напряжения, получаемый на обкладках таких датчиков. Эту проблему можно решить глубокими исследованиями фотовольтаических эффектов и совершенно новыми конструктивными решениями.

### Библиографический список

1. Шалимова К.В. Физика полупроводников. М.: Энергоатомиздат. 1985. С. 366-368
2. Кикоин И.К., Лазарев С.Д. Фотоэлектромагнитный эффект. Успехи физических наук. Т. 124, вып. 4, 1979. С. 588-616.
3. Jan Tauc, Milena Zavetova. Photo-piezoelectric effect in semiconductors. Czechoslovak Journal of Physics, 9, 1959. P. 572-577

УДК 535.243.25

**А. С. Параскун**

студент кафедры конструирования и технологий электронных и лазерных средств

**О.Д. Москалец**

доц., канд. техн. наук – научный руководитель

**М. А. Ваганов**

доц., канд. техн. наук – научный руководитель

### ПРОЦЕСС ФОТОДЕТЕКТИРОВАНИЯ И ОЦЕНКА ЭНЕРГЕТИЧЕСКОГО СПЕКТРА ОПТИЧЕСКОГО ИЗЛУЧЕНИЯ МНОГОКАНАЛЬНЫМ РЕЗОНАНСНЫМ СПЕКТРАЛЬНЫМ ПРИБОРОМ

Измерение спектров оптических сигналов является одним из важнейших видов оптических измерений, а оценка энергетического спектра стационарного случайного процесса представляет собой важнейший раздел теории и практики статистических измерений.

Существует несколько способов получения оценки энергетического спектра: с помощью предварительно определенной оценки корреляционной функции; на базе обработки мгновенного спектра; с помощью метода фильтрации. Данная работа посвящена реализации последнего метода с помощью многоканальной узкополосной системы в форме многоканального спектрометра оптического диапазона (рис. 1) [1]. Такой метод получения оценки представляется идеальным в классе получения оценок методом фильтрации.

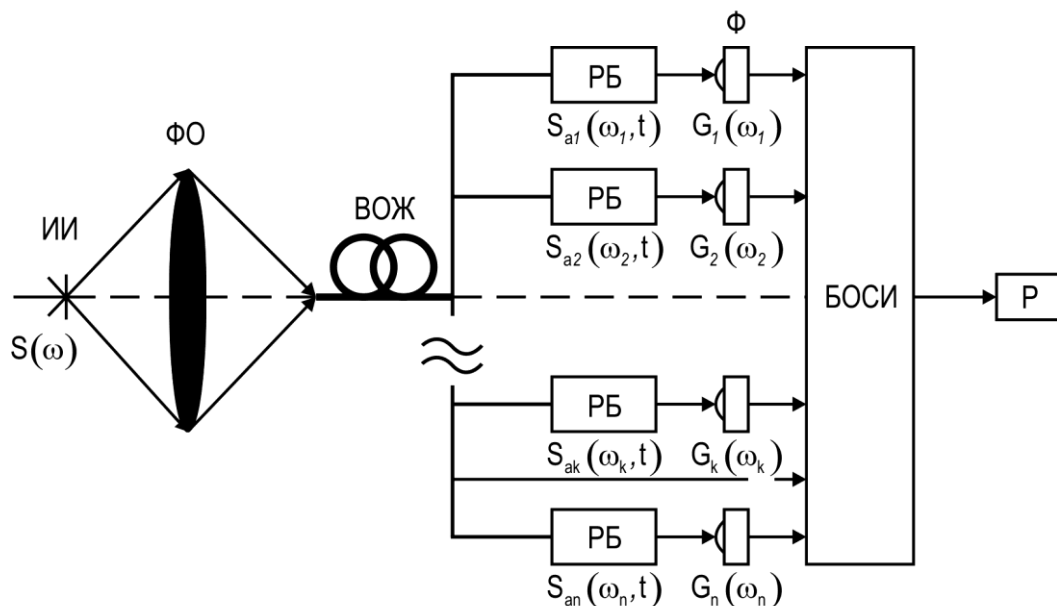


Рис. 1. Структурная схема многоканального резонаторного спектрометра оптического диапазона

Где ИИ – источник оптического излучения; ФО – формирующая оптика; ВОЖ – волоконно–оптический жгут; РБ – резонаторные блоки; Ф – фотоприемники; БОСИ – блок обработки спектроскопической информации; P – регистратор;  $S(\omega)$  – комплексный спектр излучения на входе спектрометра;  $S_a(\omega, t)$  – комплексный аппаратный спектр;  $G(\omega)$  – энергетический спектр оптического излучения, получаемый спектрометром.

Функциональная схема одного канала анализа спектра многоканального спектрометра приведена на рис. 2.

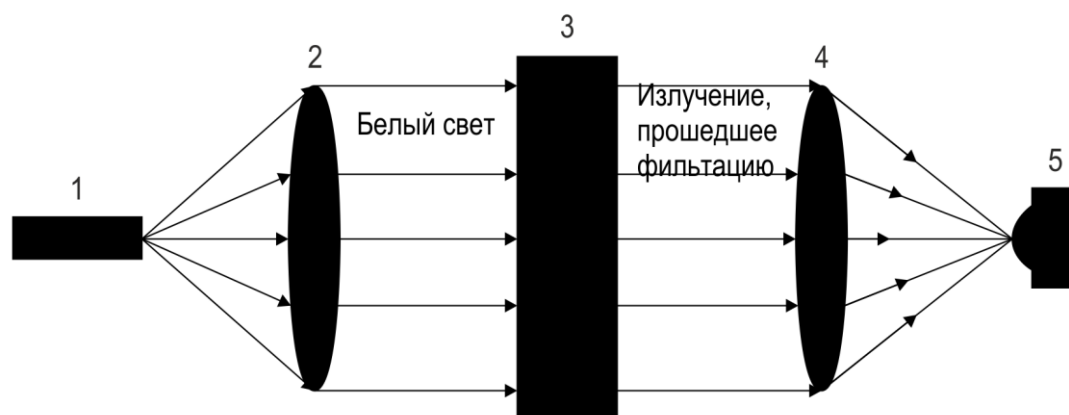


Рис. 2 – Функциональная схема одного канала анализа спектра многоканального спектрометра: 1 – выходной торец волоконно–оптического жгута; 2 – первая коллимирующая линза; 3 – резонатор; 4 – вторая коллимирующая линза; 5 – фотоприемник.

Спектральная характеристика резонатора приведена на рис. 3.

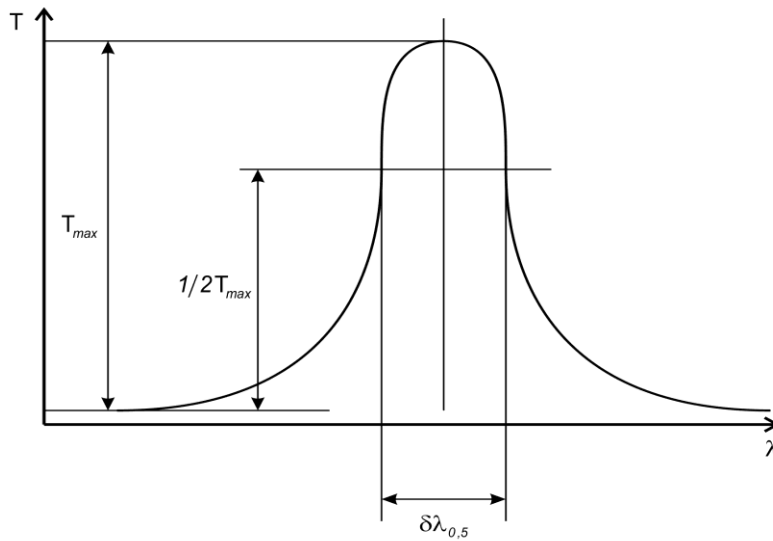


Рис. 3. Спектральная характеристика резонатора оптического диапазона  
Функциональная схема фотодетектирования приведена на рис. 4.

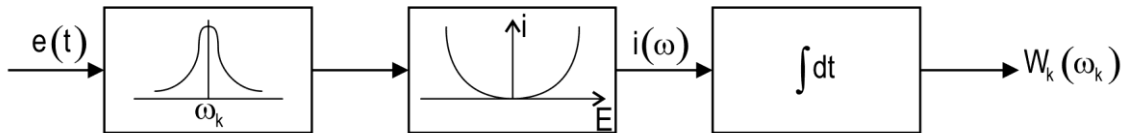


Рис. 4. Функциональная схема вычисления энергетического спектра оптического сигнала

$$\text{Ток на выходе фотодетектора } i = \gamma P = \gamma \iint_{\Delta S} |\vec{\Pi}| ds = \frac{c\gamma}{4\pi} \sqrt{\frac{\varepsilon}{\mu}} \iint_{\Delta S} E^2 ds, \quad (1)$$

где  $\gamma$  – коэффициент пропорциональности;  $P$  – мощность падающего излучения;  $\vec{\Pi}$  – вектор Пойнтинга падающего излучения;  $\vec{e}$  – единичный вектор;  $c$  – скорость света;  $\varepsilon$  – диэлектрическая проницаемость;  $\mu$  – магнитная проницаемость;  $\Delta S$  – площадь чувствительной поверхности фотоприемника.

Колебания напряженности электрической компоненты поля в плоскости фотоприемника даются выражением:

$$\vec{E}_k = \vec{E}_k = \frac{\vec{e}_1}{2\pi} \left( \int_{-\omega_k - \Delta\omega}^{-\omega_k + \Delta\omega} S(\omega) K_k(\omega) B_k(\omega) e^{i\omega t} d\omega + \frac{\vec{e}_1}{2\pi} \int_{\omega_k - \Delta\omega}^{\omega_k + \Delta\omega} S(\omega) K_k(\omega) B_k(\omega) e^{i\omega t} d\omega \right) \quad (2)$$

Вектор Пойнтинга:

$$\vec{\Pi} = \frac{c}{4\pi^2} \sqrt{\frac{\varepsilon}{\mu}} \vec{e} \int_{\omega_k - \Delta\omega}^{\omega_k + \Delta\omega} S(\omega) K_k(\omega) B_k(\omega) e^{i\omega t} d\omega \int_{-\omega_k - \Delta\omega}^{-\omega_k + \Delta\omega} S^*(\omega') K_k^*(\omega') B_k^*(\omega') e^{-i\omega' t} d\omega'. \quad (3)$$

Наличие интегратора является принципиально важным и говорит о том, что результат оценки определяется накопленной энергией. Это означает необходимость интегрирования вектора Пойнтинга во времени:

$$\int_{-T}^T \bar{\Pi}(t) dt = \int_{-T}^T dt \int_{\omega_k - \Delta\omega}^{\omega_k + \Delta\omega} S(\omega) K_k(\omega) B_k(\omega) e^{i\omega t} d\omega \int_{\omega_k - \Delta\omega}^{\omega_k + \Delta\omega} S^*(\omega') K_k^*(\omega') B_k^*(\omega') e^{-i\omega' t} d\omega'. \quad (4)$$

Применение теории вытянутых волновых сфероидальных дает:

$$G_k(\omega) = (\Delta\omega_k)^2 P_k(\omega_k) \int_{-\Delta\omega_k}^{\Delta\omega_k} |K_{kk}(\omega_k, \omega')|^2 |B_{kk}(\omega')|^2 |S_0(\omega')|^2 d\omega', \quad (5)$$

где  $P_k(\omega_k)$  – коэффициент, учитывающий спектральную чувствительность фотоприемника для  $k$  – го канала.

Это выражение определяет плотность энергии, запасенной в процессе оценивания энергетического спектра оптического излучения.

Энергия, запасенная во время этого процесса,  $W_k = \iint_{\Delta S} G(\omega) ds$  определяет оценку энергетического спектра оптического излучения как стационарного случайного процесса. Если постоянная времени интегрирующей цепочки значительно превышает время обработки, то имеет место процесс, близкий к идеальному интегрированию, когда это условие перестает выполняться, появляются искажения выполняемой оценки энергетического спектра.

Выполненный анализ показал явную зависимость свойств оценки от характеристики передаточной функции узкополосного фильтра и времени идеального интегрирования. В зависимости от передаточной функции могут быть определены искажения оценки энергетического спектра. Установлено, что применение простейшего интегратора в виде интегрирующей цепочки позволяет выполнить интегрирование, близкое к идеальному, и установить искажения при длительном времени наблюдения.

Выполненный анализ показал явную зависимость свойств оценки от характеристики передаточной функции узкополосного фильтра и времени идеального интегрирования. В зависимости от передаточной функции могут быть определены искажения оценки энергетического спектра. Установлено, что применение простейшего интегратора в виде интегрирующей цепочки позволяет выполнить интегрирование, близкое к идеальному, и установить искажения при длительном времени наблюдения.

Полученная оценка энергетического спектра оптического излучения является состоятельной и асимптотически не смещенной [3].

### Библиографический список

1. Пат. 86734 РФ, МПК G 01 J 3/26. Параллельный анализатор спектра сигналов оптического диапазона.
2. Ваганов М.А., Москалец О.Д. Параллельный анализ спектра динамических сигналов// Информационно-управляющие системы. 2011. № 5. С. 15–22.
3. Bendat J.S., Piersol A.G. Measurement and analysis of random data. John Wiley & Sons, Inc. New York–London–Sydney. 1967.

УДК 621.791.3

**А.А Рудых**

магистрант кафедры конструирования и технологий электронных и лазерных средств

**В.П. Ларин**

д-р техн. наук, профессор – научный руководитель

## ИССЛЕДОВАНИЕ ЗАВИСИМОСТИ КАЧЕСТВА ПАЯНЫХ СОЕДИНЕНИЙ ЭЛЕКТРОННЫХ МОДУЛЕЙ ОТ ФАКТОРОВ, ОПРЕДЕЛЯЕМЫХ ПАЯЛЬНОЙ ПАСТОЙ

Технология поверхностного монтажа весьма сложна, на качество пайки электронных модулей влияет множество различных факторов. Так, надежность конечного изделия сильно зависит не только от качества выполнения операций очистки и нанесения защитного покрытия [1], но и в значительной степени – от проблем, связанных с используемой паяльной пастой. Статистика



показывает, что более половины дефектов возникает еще до поступления печатных плат в установщик компонентов. Наиболее характерны такие дефекты, как: перемычки между контактными площадками, приводящие к дефекту «Надгробный камень»; расслоение флюса; вызывающее неполное нанесение паяльной пасты на контактные площадки (КП) и др. Поэтому необходимо уделять большое внимание характеристикам паяльной пасты и способу нанесения.

Решая задачу поверхностного монтажа конкретной печатной платы (ПП), в соответствии с комплексом требований необходимо выбрать марку паяльной пасты, обладающую необходимыми свойствами и удовлетворяющую параметрам выбранного способа нанесения. Понятно, что при каждом запуске в производство нового изделия менять паяльную пасту не целесообразно, но и производить монтаж компонентов на плату вручную не допустимо, учитывая высокий уровень развития автоматизированных технологий.

В процессе выполнения исследований по проблеме, связанной с паяльной пастой, был разработан перечень факторов, оказывающих влияние на качество паяных соединений и эти факторы были подвергнуты анализу (рис. 1) с целью определения эффективных технологических решений по проблеме получения качественного паяного соединения.

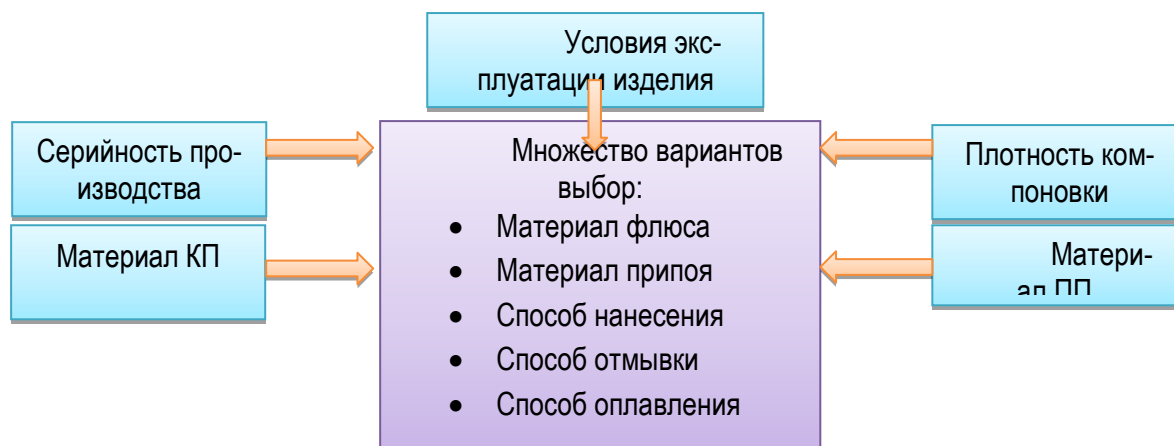


Рис. 1. Основные факторы, определяющие выбор технологических решений

Паяльную пасту необходимо рассматривать не только как смесь припоя, флюса и добавок, формирующих реологические свойства пасты, но и как взаимодействие химических и физических характеристик материалов: небольшие различия в составе пасты и условия технологического процесса могут существенно повлиять на «поведение» паяльной пасты, что в конечном итоге будет причиной дефектов. В таблице представлены выявленные дефекты паяных соединений в процессе производства, вызванные паяльной пастой, дополняющие атлас дефектов [2].

Дефекты паяных соединений, вызванных паяльной пастой

Фактор, влияющий на паяльную пасту	Дефект
Несоответствие припоя и флюса: – большая разница температуры плавления припоя и флюса; – удельный вес флюса больше удельного веса припоя	Включения флюса в паяном шве Отсутствие или плохая смачиваемость контактных площадок
Припой не взаимодействует с паяемым металлом	Отсутствие или плохая смачиваемость контактных площадок
Низкая активность флюса	Отсутствие или плохая смачиваемость контактных площадок
Неправильное хранение паяльной пасты: – расслоение припоя и флюса;	Не оплавленная паяльная паста Неравномерное нанесение паяльной пасты

Фактор, влияющий на паяльную пасту	Дефект
– просроченность годности	
Неправильный выбор размера частиц припоя	Закупоривание апертур, как следствие, недостаток паяльной пасты на контактной площадке. Избыток нанесения паяльной пасты на контактную площадку, как следствие – перемычки и мостики припоя между выводами компонентов
Истощение флюса во время пайки и повторное окисление частиц припоя и паяемых поверхностей	Шарики припоя на контактных площадках – паяное соединение отсутствует или имеет низкую механическую надежность
Истощение и потеря защитных свойств флюса на этапе предварительного нагрева	Отсутствие или плохая смачиваемость контактных площадок и выводов компонентов

Полученные перечни факторов и исследованные виды дефектов позволили построить диаграмму причинно-следственных зависимостей качества паяного соединения. На рис.2. показана детализация сложной многофакторной зависимости технологических операций получения качественного паяного соединения. В зависимости от степени влияния рассмотренных факторов на качество паяного соединения, их можно расположить в следующей последовательности по убыванию значимости: печатный узел, паяльная паста, выполнение технологических операций, инструменты и оснастка, контроль операций, технологическая среда.

Представленная диаграмма позволяет построить начальный план мероприятий и детального анализа при наличии каких-либо дефектов паяных соединений. Если поставлена задача повышения выпуска годной продукции, то необходимо следить за приведенными в диаграмме характеристиками и их параметрами. Например, если предположительно дефекты связаны с паяльной пастой, то необходим более детальный анализ состава, вязкости, временных и температурных характеристик операций нанесения и перемещения в установку пайки, условий и продолжительности хранения и т.п.

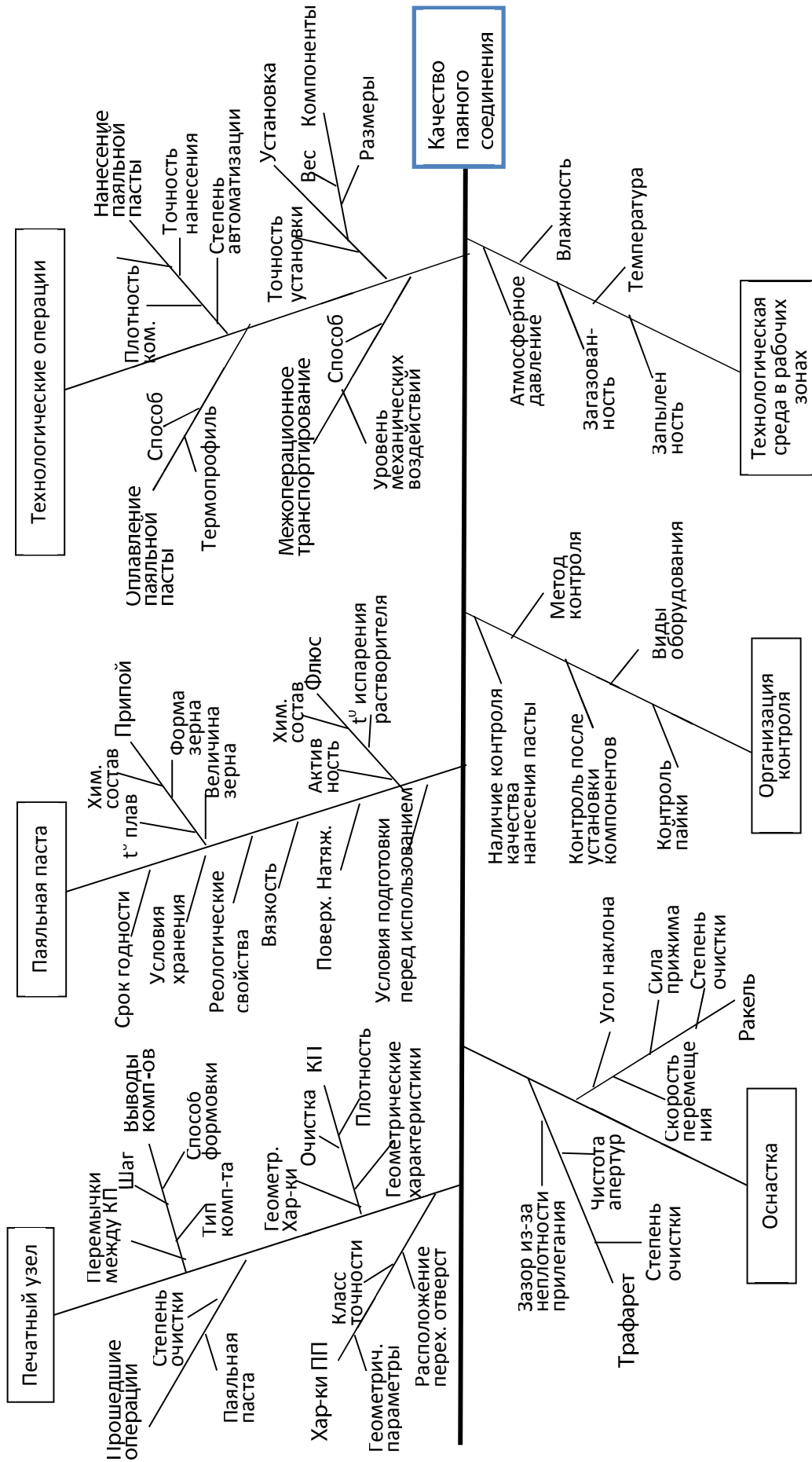


Рис.2. Причинно-следственная диаграмма качества паяного соединения

**Библиографический список**

1. Ларин В.П., Рудых А.А. Обеспечение качества многофункционального покрытия электронных модулей // Научная сессия ГУАП: сб. докл. ч. II. Технические науки. СПб.: ГУАП, 2015. С. 115–120.
2. Калининченко Н.П. Атлас дефектов паяных соединений. М.: ТПУ, 2012. 83 с.

УДК 621.735.33

**С. Н. Федоров**

магистрант кафедры конструирования и технологий электронных и лазерных средств

**О.М. Филонов**

к-т техн. наук, доц. – научный руководитель

**СОВРЕМЕННОЕ ИЗМЕРИТЕЛЬНОЕ УСТРОЙСТВО - НАНОМЕХАНИЧЕСКИЙ РЕЗОНАТОР**

Развитие нанотехнологий привело к возможности создания и исследования наноразмерных структур. Так, наноэлектромеханические системы (НЭМС) являются следующим шагом миниатюризации по отношению к микроэлектромеханическим системам (МЭМС) и увеличений ее эффективности для исследования процессов на атомарном уровне. Термин МЭМС описывает устройства, где механические функции работают в сочетании с электрическим сигналом (рис. 1). НЭМС устройства объединяют в себе электронные и механические компоненты размером до 100 нм. Микро- и наноэлектромеханические системы широко применяются в фундаментальных исследованиях.

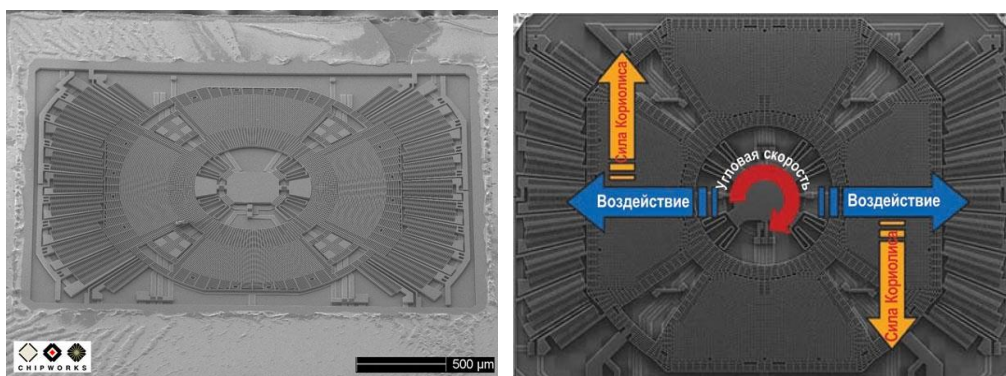


Рис. 1. Изображение МЭМС-гироскопа, полученное с помощью микроскопа

Их пороговая чувствительность по сравнению с механическими устройствами является беспрецедентно высокой: по силе – на уровне нескольких цептоньютонов ( $1 \text{ цН} = 10^{-21} \text{ Н}$ ) и по массе – на уровне нескольких цептограммов ( $1 \text{ цг} = 10^{-21} \text{ г}$ ). С помощью этих миниатюрных сенсоров можно исследовать такие процессы: измерение силы между индивидуальными биомолекулами, исследование магнитного поля индивидуальных ядерных спинов, измерение массы отдельных молекул и атомов и т.д. Но в основном МЭМС- и НЭМС-системы находят свое применение в коммерческих устройствах как всевозможные датчики (рис. 2): акселерометры, гироскопы, датчики давления и т.д.

Датчик – это устройство, воспринимающее внешние воздействия и реагирующее на них изменением электрических сигналов. Под внешним воздействием подразумевается измеряемая физическая величина (температура, давление, ускорение и т.д.). Любой датчик является преобразователем энергии (механической в электрическую – все зависит от типа измеряемой величины). Отличие датчика от преобразователя в том, что преобразователь конвертирует один тип

энергии в другой, тогда как датчик любой тип энергии внешнего воздействия – в электрический сигнал. Датчик может состоять из нескольких преобразователей (рис. 3).

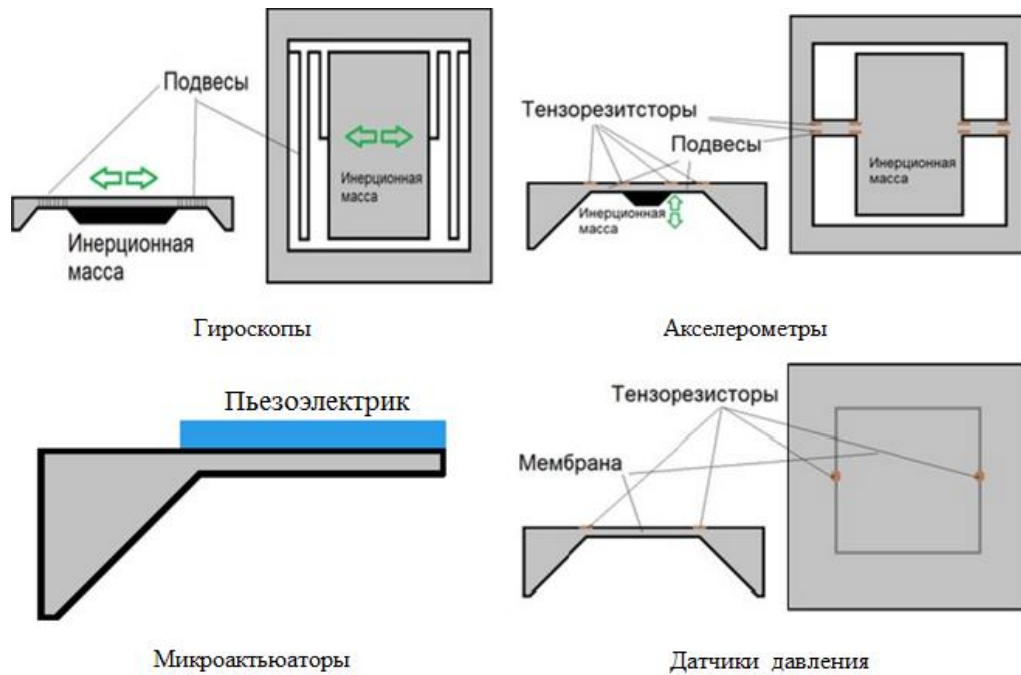


Рис. 2. Пример МЭМС устройств с кратким описанием конструкции

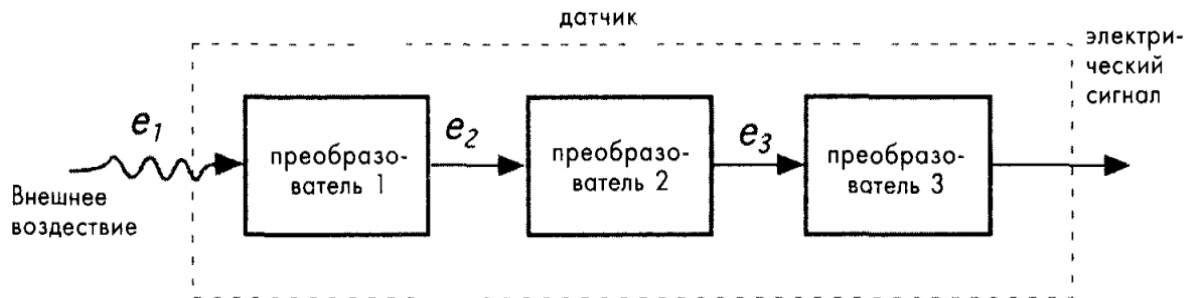


Рис. 3. Структурная схема датчика:  $e_1$ ,  $e_2$  – различные виды энергии

Наномеханические резонаторы, размеры которых в плане, как правило, находятся в диапазоне от нескольких десятков до нескольких сотен нанометров, являются уникальными по характеристикам: их фундаментальные частоты лежат в полосе от нескольких мегагерц до нескольких десятков гигагерц, механическая добротность составляет несколько десятков тысяч, активная масса порядка нескольких фемтограммов ( $10^{-15}$  г), поглощаемая мощность – порядка нескольких аттоватт ( $10^{-17}$  Вт). В настоящее время по некоторым параметрам достигнуты следующие пороговые чувствительности НЭМС: по перемещению –  $2 \times 10^{-15}$  м Гц $^{-1/2}$ ; по массе – ( $10^{-18}$ – $10^{-19}$ ) г Гц $^{-1/2}$ ; по силе –  $10^{-16}$  Н Гц $^{-1/2}$ ; по электрическому заряду –  $10^{-1}$  е Гц $^{-1/2}$  ( $e = 1,6 \times 10^{-19}$  Кл – заряд электрона); по пространственному разрешению – менее 100 нм.

Как следует из приведенных выше данных, уникальные свойства НЭМС позволяют создавать сверхчувствительные измерительные приборы в таких областях, как квантовая метрология, сканирующая силовая микроскопия, ядерный магнитный резонанс (ЯМР) и др. Кроме того, из ведущихся сейчас разработок в этой области следует, что возможные применения НЭМС включают в себя также ряд методик по определению электронной структуры отдельных молекул и управлению их перемещением, в частности, для анализа атомной структуры наноматериалов и биомолекул.

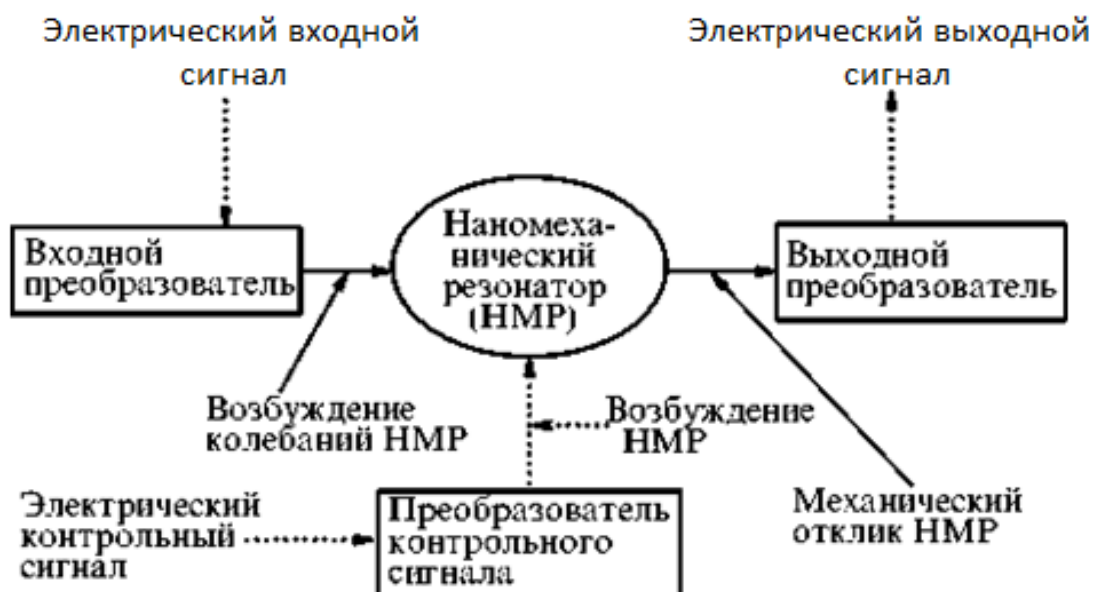


Рис. 4. Структурная схема НЭМС

На рис. 4 приведена типичная структурная схема НЭМС. Основным элементом НЭМС является наномеханический резонатор (НМР), возбуждаемый электрическими сигналами через входной преобразователь, который преобразует электрическую энергию в энергию механических колебаний резонатора (на его основной гармонике или субгармониках). Одновременно на НМР через отдельный преобразователь подается контрольный сигнал, который связан с изучаемым объектом. Этот сигнал может содержать информацию об измеряемом параметре (например, о смещении объекта или его магнитном моменте). Результирующий сигнал механических колебаний НМР, несущий информацию об измеряемой величине, на выходе вновь преобразуется в электрический сигнал, который после надлежащего усиления поступает в регистрирующее устройство.

Наномеханические резонаторы привлекают особое внимание, поскольку они более чувствительны и менее энергозатратны (один миллион наномеханических элементов рассеивают лишь одну миллионную ватта) по сравнению с их микромеханическими аналогами. Следует также отметить, что добротность механических резонаторов намного превышает добротность электрических (несверхпроводящих) резонаторов. Встраивание НМР в одноэлектронные транзисторы, сверхпроводниковые интерферометры, сверхпроводниковые зарядовые и потоковые кубиты, полосковые сверхвысокочастотные (СВЧ) резонаторы открывают новые возможности изучения богатой динамики интегрированных электромеханических систем, имеющих как механические, так и электромагнитные степени свободы. Связь между этими степенями свободы может контролироваться внешним сигналом. Такие эксперименты позволяют изучать квантовые явления в подобных гибридных электромеханических системах.

В большинстве случаев НМР представляет собой кантилевер (балка, закрепленная на одном конце (рис. 5)) или мостик (балка, закрепленная на обоих концах (рис. 6)), изготовленный с помощью субмикронной (электронной) литографии из монокристаллических материалов, таких как, например, Si или SiC. Однако добротность всех кремниевых НМР зависит от качества их поверхности. В частности, было замечено, что при уменьшении отношения поверхности к объему (для образцов, изготовленных в одном технологическом цикле) добротность возрастает.



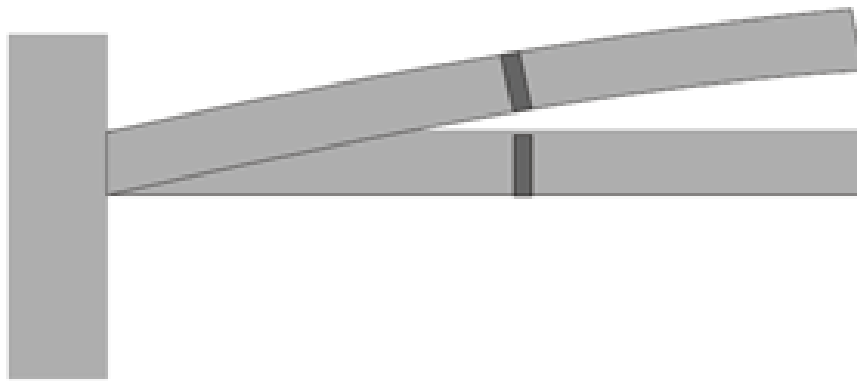


Рис. 5. Колеблющийся НМР типа кантилевера

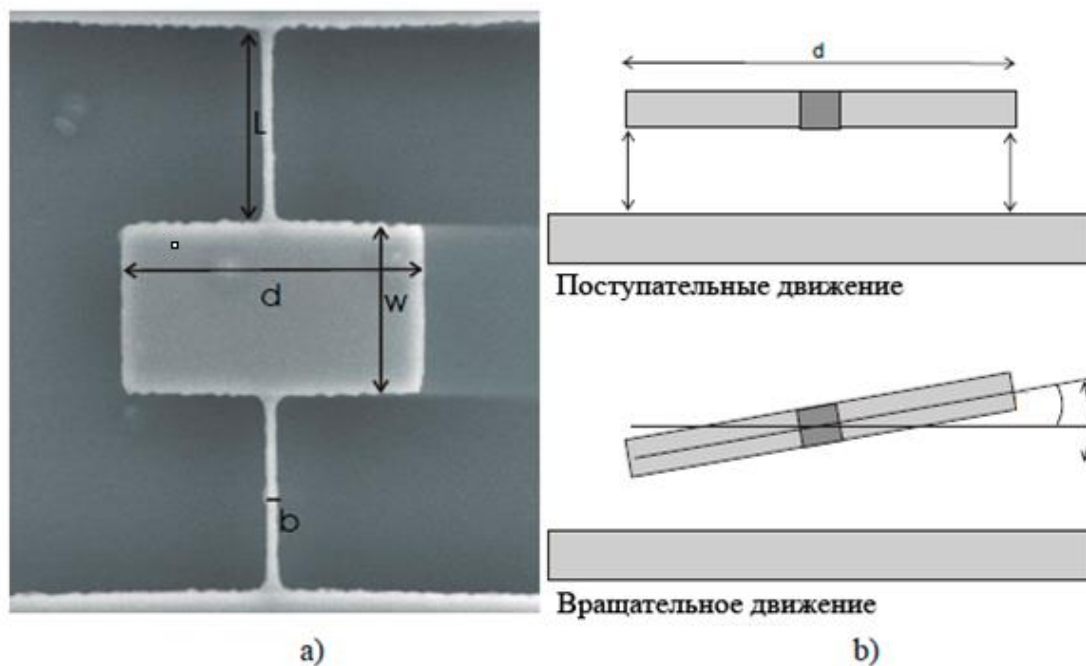


Рис. 6: НМР типа мостик: а – геометрия НМР типа мостик; б – типы движений, совершаемых НМР

Над рассмотренным устройством в настоящее время ведутся исследования во многих лабораториях мира. Интерес к НМР объясняется, прежде всего, фундаментальными проблемами квантового поведения макроскопических объектов и вопросами квантовых измерений. Высокие резонансные частоты и высокие добротности НМР, удобство их встраивания в электрические и оптические схемы открывают широкие перспективы использования этих устройств в целях создания сверхчувствительных детекторов для измерения ряда физических величин на уровне их квантовых флуктуаций. Нет сомнения, что исследования квантовых свойств наномеханических резонаторов приведут к созданию новых типов детекторов и устройств на их основе.

### Библиографический список

1. Гринберг Я. С., Пашкин Ю. А., Ильичев Е. В. Наномеханические резонаторы // Успехи физических наук. 2012. Том 182, № 4. С. 407–436.
2. Фрайден Дж., Современные датчики. Справочник. М.: Техносфера 2005. 592 с.
3. Martin Duemling Modeling and characterization of nanoelectromechanical systems: thesis Masters of Science in Materials Science and Engineering/ Virginia Polytechnic Institute, 2002. 109 с.

УДК 629.735.33

**Д. Л. Хабаров**

магистрант кафедры конструирования и технологий электронных и лазерных средств

**В. Г. Федченко**

доцент, канд. техн. наук. – научный руководитель

### АНАЛИЗ ТЕХНОЛОГИЙ ИЗГОТОВЛЕНИЯ ЧУВСТВИТЕЛЬНЫХ ЭЛЕМЕНТОВ НА ОСНОВЕ МИКРОМЕХАНИЧЕСКИХ СИСТЕМ

С развитием авиации увеличивалось время пребывания самолета в воздухе, удлинялись расстояния, которые самолет пролетал без посадки и, следовательно, без отдыха пилота. В настоящее время существуют системы стабилизации легких летательных аппаратов, использующие в качестве датчиков движения современные микромеханические системы, для которых характерны следующие преимущества: малый разброс параметров в пределах изделия, высокая технологичность и повторяемость, микроминиатюрность, высокая функциональность, улучшенные характеристики функционирования, низкая стоимость. Данные критерии являются главными в авиастроении, чтобы выбрать более эффективный датчик, нужно разобраться в анализе технологии изготовления сенсора, понимая под этим чувствительный элемент.

Традиционные технологии изготовления интегральных схем (ИС) легли в основу технологических процессов изготовления микроэлектромеханических систем (МЭМС), таких как микродатчики (сенсоры). Ниже на (рис.1) приводится блок-схема ТП [1].

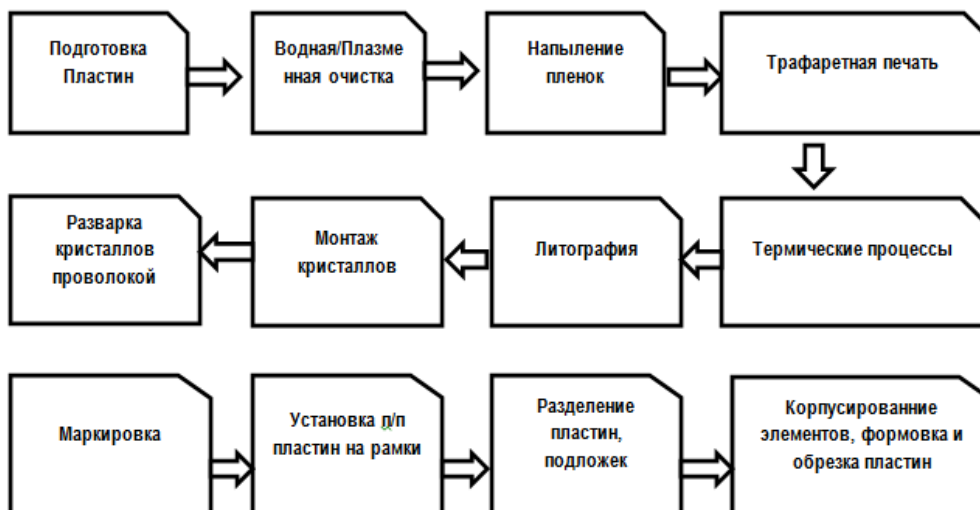


Рис.1. Технологический процесс МЭМС

Наиболее значимые части технологии были приняты с ИС. Например, почти все устройства базируются на пластинах кремния и их структуры реализуются в тонких пленках, используются фотолитографические методы, подобно ИС. Однако существуют технологические процессы, которые не являются технологиями ИС, и поскольку технология продолжает развиваться, отрыв от технологии ИС также растет.

Выделяют основные три стандартных блока технологии МЭМС:

- способность депонировать тончайшие пленки материала на подложку;
- точно воспроизводить образы на вершины пленок литографическим отображением;
- травить пленки выборочно к образам.

Следует понимать основные факторы, влияющие на выбор различных слоистых структур материалов при создании чувствительных элементов: совместимость кристаллических решёток,



учёт коэффициента температурного линейного расширения, согласование максимальной тепловой нагрузки, а для полупроводниковых материалов температурный переход в состояние, при котором концентрация носителей собственных зарядов близка к примесной. При выборе материалов положено придерживаться традиционными требованиями: электрические и механические прочности, высокие тепло и электропроводности, стойкость к внешним воздействиям. Основным конструктивным материалом среди изготовления чувствительных элементов акселерометров, гироскопов и датчиков давления является кремний, обладающий таким главным свойством, как получение в окислительной среде утончённого, крепкого и влагонепроницаемого слоя аморфной двуокиси кремния ( $\text{SiO}_2$ ) на поверхности кристалла. Данная пленка может быть использована как для защитных покрытий поверхности кремния, так и для диэлектрических изоляционных схемных элементов.

Операция окисления кремниевых пластин проходит около 6 ч в зависимости от толщин окисла ( $\text{SiO}_2$ ) и способов окислений. Слой двуокиси кремния образуется обычно на подложке за счёт химических соединений в полупроводнике атомных единиц кремния с кислородом, потом подается к поверхности кремниевой подложки, техническая печь осуществляет нагрев до высоких температур (рис. 2). В кварцевой трубке диаметром 110–129 мм размещаются пластины.



Рис. 2. Термическое окисление

Наиболее часто толщина окисла составляет десятые доли микрона, а верхний потолок термического окисления составляет 1 мкм.

Фотолитография является главным технологическим процессом в микроэлектронике, обычно она применяется для получения линий шириной до 2–3 мкм. В настоящее время для изготовления шаблонов используют электронно-лучевую литографию, строящуюся по эффекту обратного рассеивания отраженных электронов в микроструктурах с высокой плотностью, минимальная ширина линии ограничивается величиной порядка 0,5 мкм, однако ионно-лучевая литография позволяет проводить локальное легирование примесей с разрешающей способностью меньше 0,01 мкм.

Например, в фотолитографии, прежде всего, следует сделать маску (шаблон), с назначенным рисунком. После завершения разработчик схемы должен определить места расположения всех компонентов на поверхности кристалла. Несмотря на то, что поверхность кристалла обычно имеет размер от 0,4 до 4 мм, оригинал должен быть изготовлен во много раз больше кристалла, чтобы при воспроизведении общей картины избежать ошибок или неточностей линейных размеров [2].

Следующим этапом в процессе фотолитографии заключается в запечатлении оригинала большой фотокамерой. В больших случаях оригиналы в 500 раз больше ключевых рисунков микросхем на кристалле. Последующие фотографии делаются с последовательным уменьшением первичного рисунка в 100 раз, а затем в 10 раз, чтоб на выходе получался высокоточный размер рисунка. Потом фотошаблон используется для последующего копирования. Полученные таким образом фотошаблоны используются в фотолитографии в качестве маски для трансляции рисунка на поверхность подложки, покрытой слоем фоторезистора.

Обычно литье фоторезистора или распыление равномерно распределяется на вращающуюся поверхность пластины под действием центробежных сил. Толщина фоторезистора составляет 0,6–6 мкм (толщина зависит от типа фоторезистора).

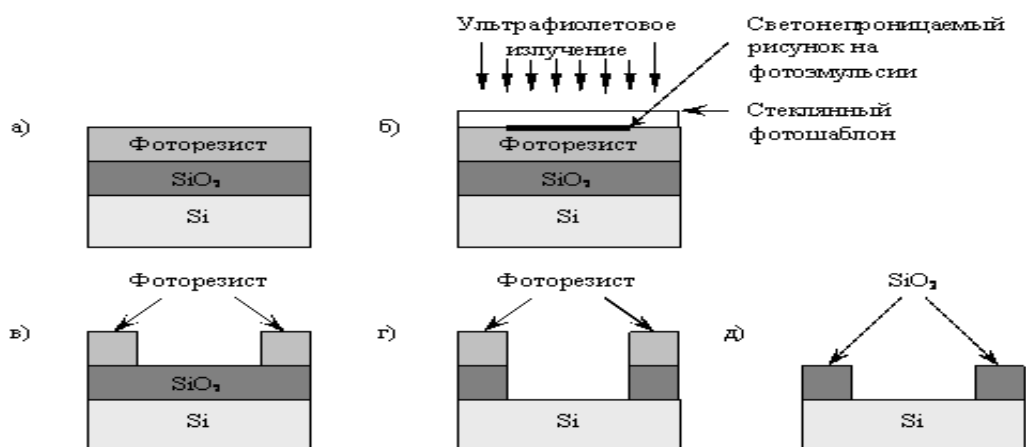


Рис. 3. Этапы фотолитографического процесса: а) первичное покрытие; б) контактная печать; в) после проявления; г) после удаления окисла; д) после удаления фоторезистора

Чтобы принять изображение на пластине высокого качества, следует одновременно и взаимосвязано изменять время и проявления экспонирования рисунка.

Наиболее употребляемым методом создания рисунка на пластине является метод контактной печати, для наглядности которого рассмотрим случай, когда в слое двуокиси кремния, расположенном на подложке, необходимо сделать миниатюрные отверстия. Сначала на окисный слой наносится фоторезистор (см. рис. 3, а), а затем к фоторезистору прикладывается стеклянный фотошаблон (маска) с рисунком, соответствующим той части окисла, которая должна быть удалена. После этого осуществляется экспонирование структуры в ультрафиолетовых лучах (см. рис. 3, б). В процессе проявления неэкспонированные участки фоторезистора пропадают, оставляя при этом окна, покрывающие остальную часть окисла, химически стойкие к буферному кислотному раствору, обычно используемому для травления окисного слоя с целью получения окон в окисле (см. рис. 3, в, г). Затем оставшийся слой фоторезистора удаляется с подложки, и подложка становится готова (см. рис. 3, д).[3]

Требуемая конфигурация чувствительного элемента с глухими и сквозными отверстиями, щелями и пазами обеспечивается преимущественно локальным химическим травлением. Доступ к кремнию травителей обеспечен вскрытием защитного слоя.

При химическом травлении окислительно-восстановительные реакции протекают на поверхности кремния, связанные с чисто химическим взаимодействием молекул травителей с поверхностными атомами. Состав травителей, как правило, подбирается опытным путем. Изотропное травление одно из самых известных и распространенных способов микропрофилирования (рис. 4). Характерной особенностью изотропного травления является одинаковое воздействие во всех кристаллографических направлениях, такое травление называют полирующим [4].

В качестве изотропных травителей используются травители на основе плавиковой и азотной кислот. Скорость травления полупроводника зависит от многих показателей: тип травителя, его температура, скорость отвода продуктов реакции и подвода реагентов к поверхности полупроводника, наличие или отсутствие дефектов, как в самом полупроводнике, так и в защитной маске, обеспечивающей локальность травления; испарение травителя, приводящее к изменению его концентрации и т.д. Большинство этих факторов плохо поддается контролю. Это приводит к тому, что неурегулирование скорости травления изменяет геометрию упругого элемента.

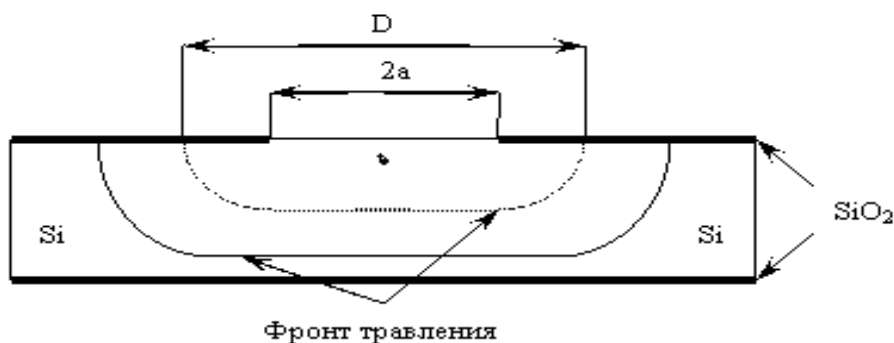


Рис. 4. Форма микропрофиля при изотропном травлении

Особенностью анизотропного травления является то, что в разных кристаллографических направлениях скорость удаления атомных слоев с поверхности травления имеет разное влияние. Анизотропные травители состоят из окислителя кремния до гидратированной двуокиси растворителя и замедлителя или ускорителя процесса травления. Процесс анизотропного травления состоит из поэтапного удаления атомных слоев (слой за слоем) с поверхности кристалла.

Скорость травления двуокисью кремния во всех применяемых травителях ниже скорости травления кремния из этого следует то, что при локальном травлении защита поверхности кремниевой пластины от травления может быть осуществлена с помощью оксидной пленки. Оксидные маски локализуют процесс травления профилированных углублений, сквозных отверстий и пазов.

Проведенный анализ существующих технологий изготовления МЭМС показал, что изготовление чувствительных элементов МЭМС-приборов является довольно сложным процессом и требует интеграции знаний большинства фундаментальных наук, таких как физика, химия, математика, материаловедение и т.д. В подавляющем большинстве вид (форма и геометрические размеры) и качественные характеристики ЧЭ МЭМС-приборов зависят от имеющихся технологических приспособлений и инструментов, необходимых для их изготовления. Выход годных, соответствующих по качественным и количественным характеристикам, чувствительных упругих элементов МЭМС-приборов составляет не более 30%, что в первую очередь определяется сложностью осуществления технологии изготовления в целом и отдельных её стадий в частности, поэтому следует находить новые пути решения, отклоняясь от традиционных технологий изготовления интегральных схем.

#### Библиографический список

1. Иванов С.В., Карелин Е.Ю., Анализ технологических процессов изготовления интегральных схем и микроэлектромеханических систем//Воронеж. С.1-2
2. Броудай И., Мерай Д. Физические основы микротехнологии. М.: Мир, 1985.315 с.
3. Распопов В.Я. Микромеханические приборы: учеб. пособие. Тульский Государственный Университет. Тула, 2002. 111 с.
4. Вавилов В.Д., Поздьяев В.И. Конструирование интегральных датчиков. М.: Изд-во МАИ, 1993. 28 с.

УДК 621.791.3

**П.М. Черноусова**

студентка кафедры конструирования и технологий электронных и лазерных средств

**В.П. Ларин**

д-р техн. наук, проф. – научный руководитель

### ФАКТОРЫ, ОПРЕДЕЛЯЮЩИЕ НАДЕЖНОСТЬ ПАЯНЫХ СОЕДИНЕНИЙ ПРИ ПОВЕРХНОСТНОМ МОНТАЖЕ

Целью данной работы является выполнение обзорно-аналитического исследования по проблеме формирования и обеспечения надежности паяных соединений электронных модулей приборов для разработки технологической системы, обеспечивающей стабильное качество паяных соединений электронных модулей бортовой аппаратуры.

В настоящее время реализуется первый этап работы с задачей анализа направлений и методов обеспечения надежности паяных соединений при выполнении процессов поверхностного монтажа компонентов на печатные платы (ПП). Объектом исследования являются паяные соединения, получаемые по технологии поверхностного монтажа, сокращенно обозначаемой ТМП (технология монтажа на поверхность). Компоненты, монтируемые на поверхность, принято обозначать ПМИЭТ (поверхностно монтируемые изделия электронной техники). Этим обозначениям соответствуют английские аббревиатуры SMT (Surface Mount Technology) и SMD – (Surface Mounted Device).

При исследовании по данной теме использован системный подход в связи с множеством факторов, влияющих на надежность паяных соединений. При применении системного подхода необходимо соблюдать определенные принципы, главные из которых относятся к структурному построению исследуемой системы.

Структура технологической системы, обеспечивающей выполнение условий формирования надежных паяных соединений в процессе поверхностного монтажа компонентов по свойствам безотказности и долговечности, представлена на рис. 1.

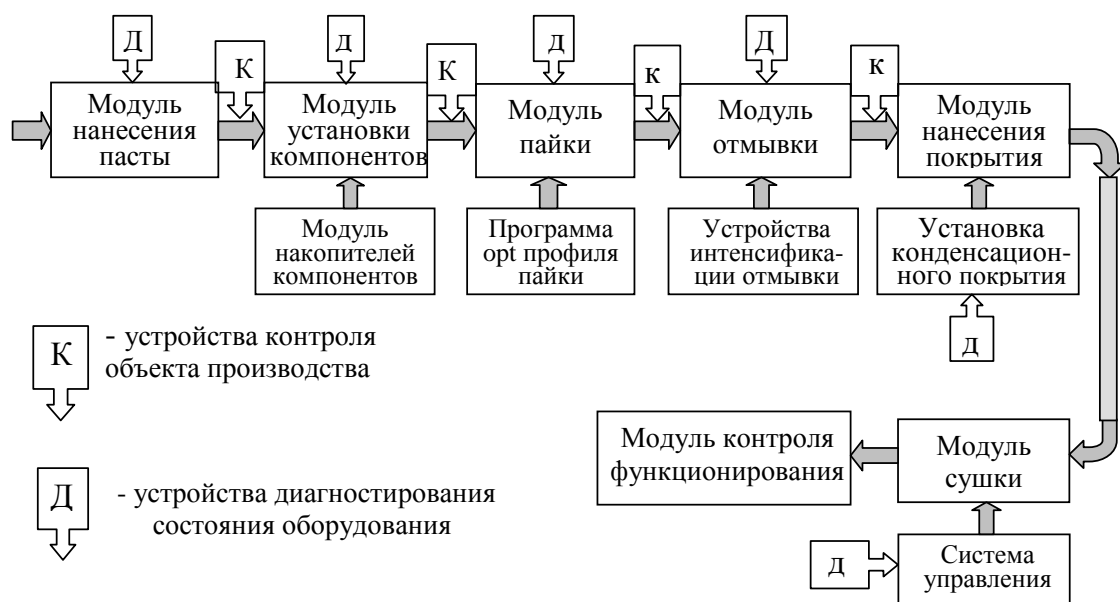


Рис. 1. Структура ТС с функцией обеспечения надежности паяных соединений

Надежность аппаратуры с поверхностным монтажом в значительной степени зависит от надежности паяных электрических соединений. В соответствии с ГОСТ 27.004 - 85 [1] появление

дефекта изделия, производимого в ТС, определяется как ее неработоспособное состояние по параметрам продукции, а событие классифицируется как параметрический отказ ТС. Таким образом, любой из структурных элементов ТС, изображенной на рис. 1, ставший причиной дефекта изделия, переводит ТС в состояние параметрического отказа, который формулируется как отказ ТС, при котором сохраняется ее функционирование, но происходит выход значений одного или нескольких параметров технологического процесса за установленные пределы.

По ГОСТ 53429-2009 [2] предусмотрено 7 классов точности изготовления печатных плат. Класс точности конструкции конкретной ПП определяется размерными параметрами элементов печатного рисунка установленного данным стандартом. Для рассматриваемой задачи обеспечения надежности паяных соединений такими элементами являются формы и размеры контактных площадок посадочных мест компонентов монтажа. Стандартизованная форма и размеры конкретного паяного соединения определяются стандартизованной формой и размерами контактной площадки.

Рассмотрим задачу определения размеров контактных площадок на примере формирования посадочного места простейшего компонента типа чип-резистор. Номинальные размеры сторон контактных площадок для установки ПМИЭТ с двумя точками подсоединения M и N (мм) рассчитывают по формулам [1]:

$$M = B(D)_{\max} \left( T_s^2 + T_D^2 + \Delta t_{H.O.}^2 \right)^{1/2} \quad (1);$$

$$N = Q_n + \left( \Delta l + T_D^2 + \Delta t_{H.O.}^2 \right)^{1/2} \quad (2),$$

где  $B(D)_{\max}$  – максимальная ширина контактирующей части ПМИЭТ, мм;  $T_s$  – позиционный допуск расположения геометрического центра корпуса устанавливаемого ПМИЭТ относительно номинального положения, определяемый используемым оборудованием, мм;  $T_D$  – позиционный допуск расположения контактной площадки, мм;  $\Delta t_{H.O.}$  – нижнее предельное отклонение размеров элементов конструкции (ширины контактной площадки), мм;  $\Delta l$  – верхнее предельное отклонение длины контактирующей части устанавливаемого ПМИЭТ по техническим условиям на ПМИЭТ конкретного типа (далее – ТУ на конкретный тип), мм;  $P$  – минимальное расстояние от края контактной площадки до контактирующей части ПМИЭТ, необходимое для обеспечения пайки надлежащего качества, устанавливаемое равным 0,3 мм при пайке паяльной пастой и 0,6 мм – при пайке припоем, мм;  $Q_n$  – гарантированный размер контактной площадки, необходимый для совмещения с контактирующей частью ПМИЭТ, устанавливаемый равным от 0,1 до 0,3 мм.

Эскиз положения компонента на посадочном месте и его основные параметры представлены на рис. 2 и рис. 3 соответственно [3]. На рисунках обозначено:

- C – расстояние между центрами контактных площадок,
- X – ширина контактной площадки (поперек C);
- Y – длина контактной площадки (вдоль C);
- L – длина контура компонента (вдоль C);
- W – ширина контура компонента (поперек C);
- H – высота корпуса (максимальная).

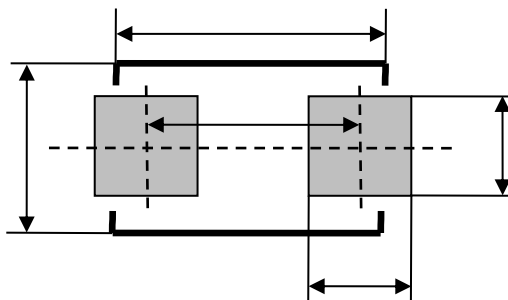


Рис. 2. Положение чип-компонента

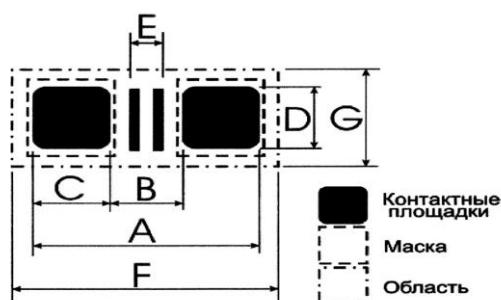


Рис. 3. Размерные параметры компонента и посадочной площадки

Рассмотрим для примера параметры чип-компонента типоразмера 0201 (таблица):

Параметры корпуса компонента типоразмера 0201

Типоразмер	Размер площадки, мм							Точность размещения
	A	B	C	D	E	F	G	
0201	0,75	0,3	0,3	0,3	0,2	1,1	0,5	0,05

Максимальный объем стандартизованного паяного соединения корпуса 0201 составляет ( $C \times D \times H$ )  $0,015 \text{ мм}^3$ . В действительности он меньше  $\sim 0,007 - 0,008 \text{ мм}^3$ .

Такой малый объем паяного соединения заставляет задуматься о его прочностных возможностях в изделиях с повышенными механическими и тепловыми нагрузками, что является задачей последующих исследований. Задача технологического этапа состоит в необходимости обеспечить качество соединения с указанными размерными параметрами. В связи с этим рассмотрим две группы факторов, влияющих на надежность:

- По показателям безотказности:
  - размерные параметры;
  - наличие и вид дефектов;
  - материалы;
  - состояние контактируемых поверхностей;
  - способ пайки;
  - режим пайки;
  - параметры технологической среды.
- По показателям долговечности:
  - степень защиты соединения;
  - рабочий диапазон температур соединения;
  - уровень механических нагрузок на соединение;
  - старение материалов.

Как видим, показатели безотказности непосредственно связаны с технологическими факторами, в то время как показатели долговечности в значительной степени определяются эксплуатационными факторами. Для решения задач обеспечения надежности соединения необходимы исследования с применением различных моделей, проведения испытаний, обработка и анализ результатов контрольных операций. На начальном этапе исследований большой информативностью и наглядным представлением объектов исследований обладают структурно-функциональные модели и модели на основе причинно-следственных диаграмм.

Рассмотрим разработанную структурно-функциональную модель модуля ТС, непосредственно реализующего операцию образования паяных соединений компонентов на ПП (рис. 4).

При оценке надежности паяных электрических соединений необходимо учитывать возможные причины возникновения отказов, среди которых различают отказы, обнаруженные сразу

же после окончания монтажа, и отказы, происшедшие спустя несколько недель, месяцев после ввода аппаратуры в эксплуатацию.

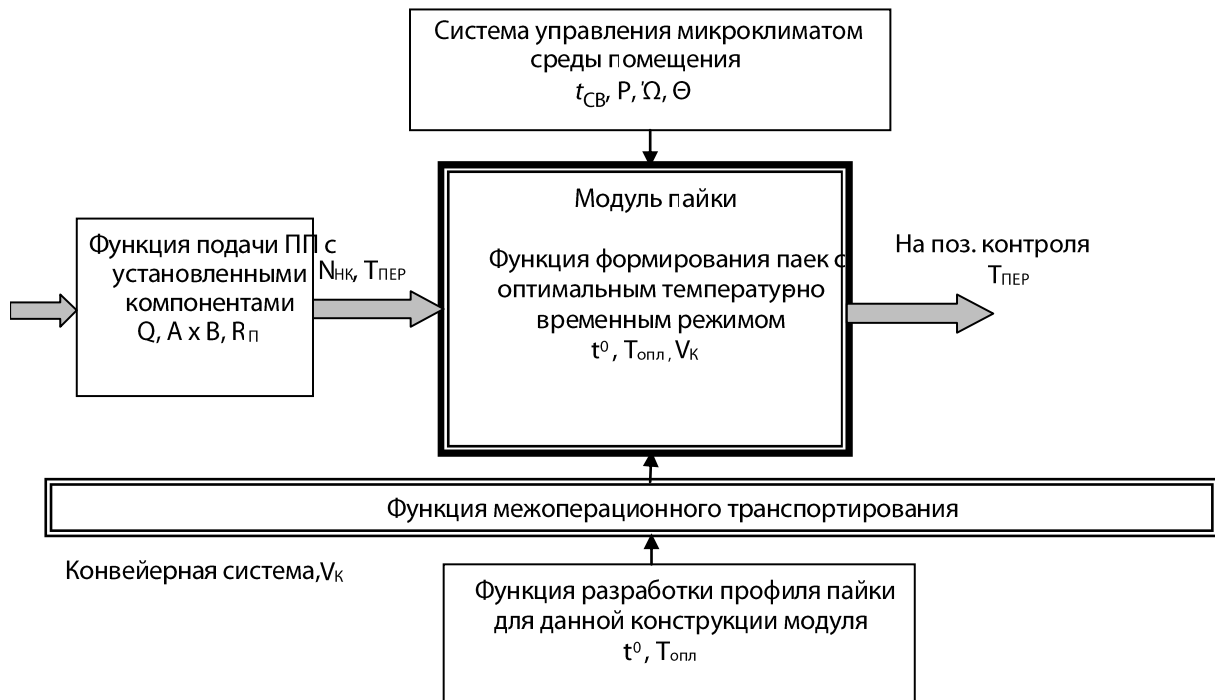


Рис. 4. Структурно-функциональная модель модуля пайки:  $Q$  – объем модуля;  $A \times B$  линейные размеры печатной платы;  $R_{п}$  – характеристики пасты;  $N_{нк}$  – емкость накопителя ПП;  $T_{пер}$  – периодичность подачи объекта в модуль;  $t_{ср}^0$  – температура в помещении;  $P$  – атмосферное давление;  $\Omega$  – влажность воздуха в помещении;  $\Theta$  – характеристики воздушной среды помещения;  $t^0$  – температурный параметр профиля пайки;  $T_{опл}$  – временной параметр профиля пайки;  $V_{к}$  – скорость конвейера в модуле пайки

Причинами первого вида отказов в большинстве случаев являются дефекты материалов или метода пайки, причинами вторых – ошибки в конструировании модуля и преждевременное ухудшение физико-химических свойств припоя, вызванное неудовлетворительным выбором режима пайки.

Оценивать уровень надежности паяных электрических соединений наиболее целесообразно по результатам специализированных испытаний на надежность, руководствуясь стандартизованными методиками (ГОСТ 16504, ГОСТ 27.402, ГОСТ 27.405, РД 50-690, IEC-PAS 62137-3, MIL STD 883). Анализ отказов позволяет установить причины их возникновения и внести соответствующие изменения в конструкцию паяного узла или технологический процесс пайки

### Библиографический список

1. ГОСТ 27.004 – 85 Надежность в технике. Системы технологические Термины и определения. М., Госстандарт, 1986. 9 с.
2. ГОСТ Р 53429 – 2009 Платы печатные. Основные параметры конструкции. М., Стандартинформ, 2010. 11 с.
3. Ларин В.П., Шелест Д.К. Конструирование и производство типовых приборов и устройств: учеб. пособие для вузов. СПб: СПбГУАП. 2005. 378 с.

УДК 004.27

**А.А. Чугунов**

студент кафедры радиотехнических и оптоэлектронных комплексов

**В.К. Прилипко**

доц., канд. техн. наук – научный руководитель

### SIMULINK-МОДЕЛЬ БИСТАБИЛЬНОЙ ПОТЕНЦИАЛЬНОЙ ЯМЫ В КУРСЕ КВАНТОВЫХ ВЫЧИСЛЕНИЙ

Работа посвящена виртуальной демонстрации одного из принципиальных положений квантовой теории информации – принципа Р. Ландауэра. Теория квантовых вычислений возникла в конце XX в.. Одной из причин, приведших к ее созданию, была попытка снять одно из принципиальных ограничений, наложенных Природой на вычислительные возможности современных компьютеров (с архитектурой фон Неймана). Речь идет о так называемом тепловом пределе, оцениваемом величиной

$$k_B T \cdot \ln 2,$$

где  $k_B$  – постоянная Больцмана, равная  $1.38 \cdot 10^{-23}$  Дж/К,  $T$  – абсолютная температура компьютера.

Эта величина означает минимальное количество теплоты, выделяемое компьютером при стирании одного бита информации. При проведении сложных вычислений возникает большое количество «мусорных» битов и потому выделяемое количество теплоты может оказаться очень большим. Так, например, при моделировании молекулы метана количество «мусорных» битов оценивается величиной порядка  $10^{43}$ , что приводит к огромному количеству выделенной теплоты – порядка  $10^{22}$  Дж (!) для температуры порядка 300 К.

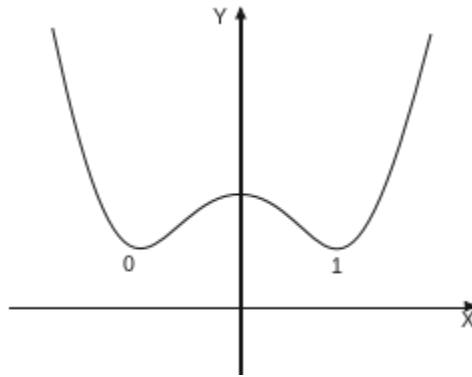
Указанная проблема происходит из того обстоятельства, что основные логические элементы (И, исключающее ИЛИ, НЕ-И, НЕ-ИЛИ, исключающее ИЛИ), участвующие в вычислительном процессе, являются логически необратимыми, т. е. по состоянию элемента на его выходе нельзя однозначно сказать, что было на его входе.

Оказывается, что указанная логическая необратимость обусловлена физической необратимостью логического устройства, а последняя всегда приводит к выделению теплоты. Такое, например происходит при работе необратимого устройства – переключателя, переводящего все биты вычислительного устройства, находящиеся в состояниях 0 и 1 в одно состояние, например, 0. Существующие устройства обработки данных реализуются на туннельных диодах, на магнитных ядрах и тонких магнитных пленках. Последние характеризуются наличием двух устойчивых состояний.

Следуя принципу Р. Ландауэра, можно предложить похожую на магнитное устройство механическую модель устройства обработки данных, которое способно хранить информацию, возникающую в ходе вычислений и в которой логическая необратимость сопутствует физической. Эта механическая модель – частица в потенциальной яме с двумя минимумами – так называемая бистабильная потенциальная яма (рис. 1).

Процесс вычисления в рассматриваемом примере – это перемещение частиц (каждая движется в своей потенциальной яме) из одного положения равновесия в другое, в соответствии с алгоритмом решаемой задачи. Роль переключателя состоит в том, что после завершения некоторого вычисления, когда оба состояния 0 и 1 будут заняты примерно одинаково, следует перевести компьютер в исходное начальное состояние.



Рис. 1. Потенциальная энергия  $U(x)$  одномерной механической системы.

В рассматриваемой механической модели для осуществления переключения необходимо приложить к частицам одну, общую для всех частиц специально силу, которая переведет каждую частицу системы в одно из двух ее состояний (но одинаковое для всех частиц, например, в 1) с вероятностью равной единице. Последнее означает, что над частицами системы совершается операция, которая с позиции логики является необратимой – получившееся состояние не позволяет определить состояние каждой из частиц до переключения. Физическая необратимость, обусловленная логической в данном примере возникает в результате действия на частицу силы трения, возникающей в момент переключения. Таким образом, операция «переключение» состоит из двух действий:

- сообщить частице скорость или, что то же – приложить к ней импульс силы;
- «включить» силу трения.

Предлагаемая на рис. 2 виртуальная демонстрация позволяет наблюдать динамику частицы, находящейся в одномерном поле сил, описываемом кривой потенциальной энергии с двумя минимумами. Можно задавать начальное положение частицы и ее массу.

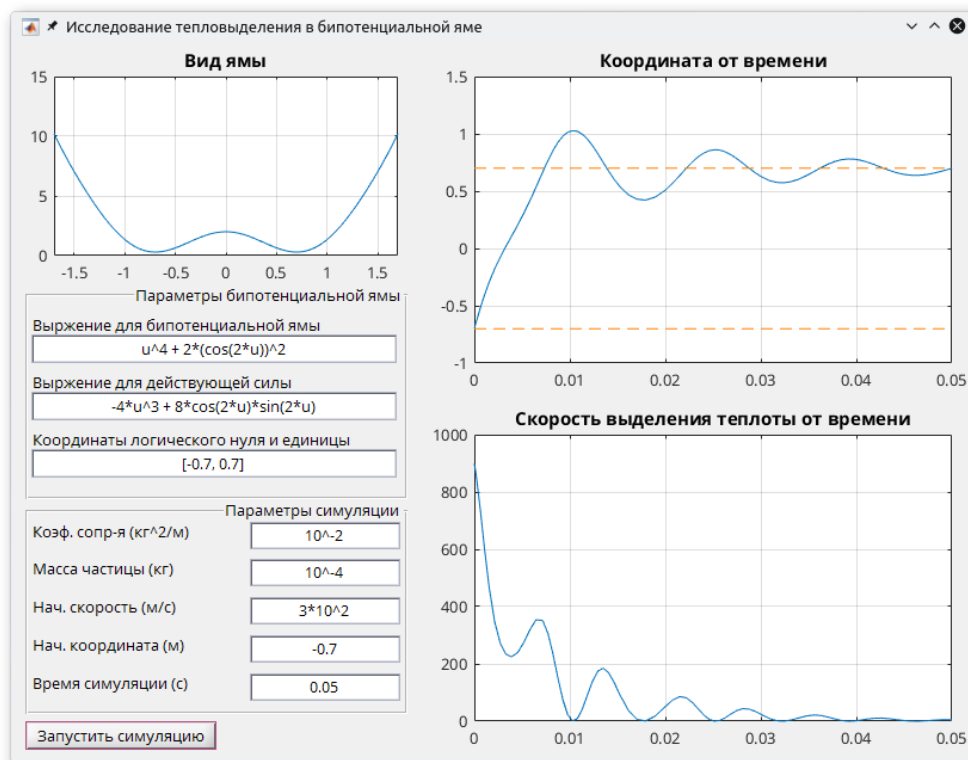


Рис.2. Внешний вид программы

Переключение состоит в том, что частице сообщается импульс силы в произвольном направлении (задавая скорость частицы в начальный момент времени) и также к частице прикладывается необходимая для ее остановки сила трения. Последняя задается с помощью коэффициента вязкого трения  $F_{тр} = kv$ . При желании можно менять вид потенциальной кривой.

Удобный интерфейс позволяет проводить поиск необходимого сочетания параметров системы для перевода частицы в общее состояние (например 0) независимо от ее начального расположения. В программе предусмотрено вычисление скорости выделения теплоты.

Теперь рассмотрим построение модели в Simulink. Пусть потенциальная энергия задана выражением

$$U(x) = x^4 + 2[\cos(2x)]^2.$$

В таком случае сила, действующая на частицу, будет иметь вид

$$F(x) = -\text{grad } U = -\frac{\partial U}{\partial x} = -4x^3 + 8 \cos(2x) \cdot \sin(2x)$$

Исходя из вида потенциальной кривой, можно сделать выводы о характере движения частицы [1, стр. 147]: а именно, после преодоления барьера посередине частица будет двигаться по гармоническому колебательному закону. Таким образом, для консервативного случая можно записать уравнение движения в виде дифференциального уравнения [2, стр. 276]:

$$\ddot{x} + \frac{1}{m} \cdot F(x) = 0, \quad (1)$$

где  $m$  – масса частицы. Для затухающих колебаний:

$$\ddot{x} - \frac{\mu}{m} \cdot \dot{x} + \frac{1}{m} \cdot F(x) = 0, \quad (2)$$

где  $\mu$  – коэффициент сопротивления.

Характер выделения тепла и его зависимость от времени можно выразить с помощью диссипативной функции [3, стр. 562]:

$$\frac{dE}{dt} = -2 \frac{\mu \cdot \dot{x}^2}{2} = \mu \cdot \dot{x}^2, \quad (3)$$

где  $E$  – полная энергия частицы.

Для построения модели на основе в формулы (1) (без силы трения) потребуется два последовательно подключенных блока «Integrator» [4, стр. 209] – выходной сигнал первого из них будет соответствовать мгновенной скорости, а выходной сигнал второго – мгновенной координате.

Для симуляции квазиупругой силы можно использовать блок «Fcn». Он в связке с блоком «Constant» и блоком «Divide» соответствует члену  $\frac{1}{m}F(x)$  в формуле (1).

Чтобы получить график зависимости координаты от времени, подключим вход блока «Scope» к выходу блока «Координата». Вид модели представлен на рис. 3. Для удобства начальные условия в блока «Скорость» и «Координата» вынесены в отдельные входы на этих блоках.

Для того, чтобы расширить модель до неконсервативного случая, в формуле (2) необходимо добавить два блока «Constant» («Масса» и «Коэффициент сопротивления»), блок «Divide» с тремя входными портами и блок «Sum». Внешний вид модели с вязким сопротивлением показан на рис. 3.

Финальный штрих – диссипативная функция из формулы (3). Для нее добавим два блока: «Multiply» и «Math Operation». Окончательный вид модели показан на рис.4.

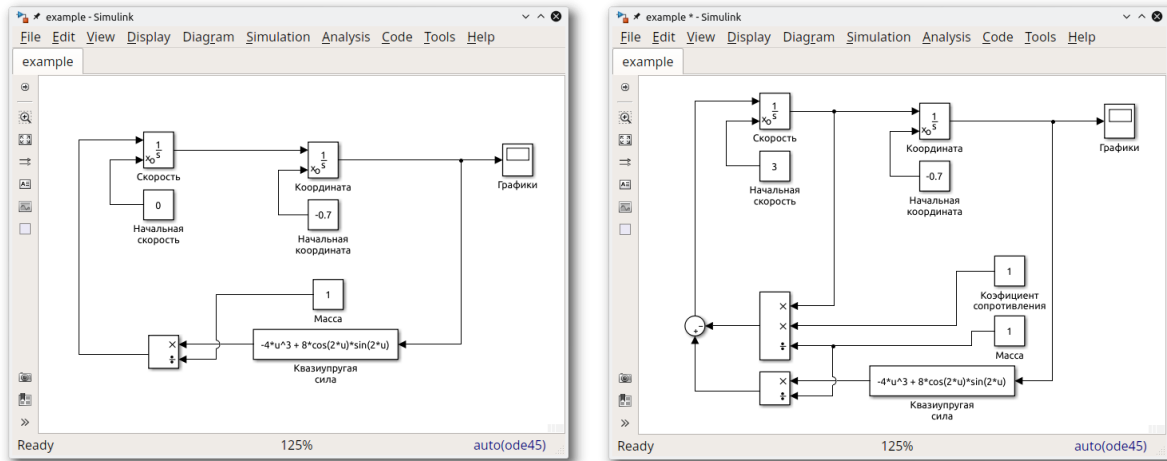


Рис.3. Модель для бистабильной ямы без силы трения (слева) и с силой трения (справа)

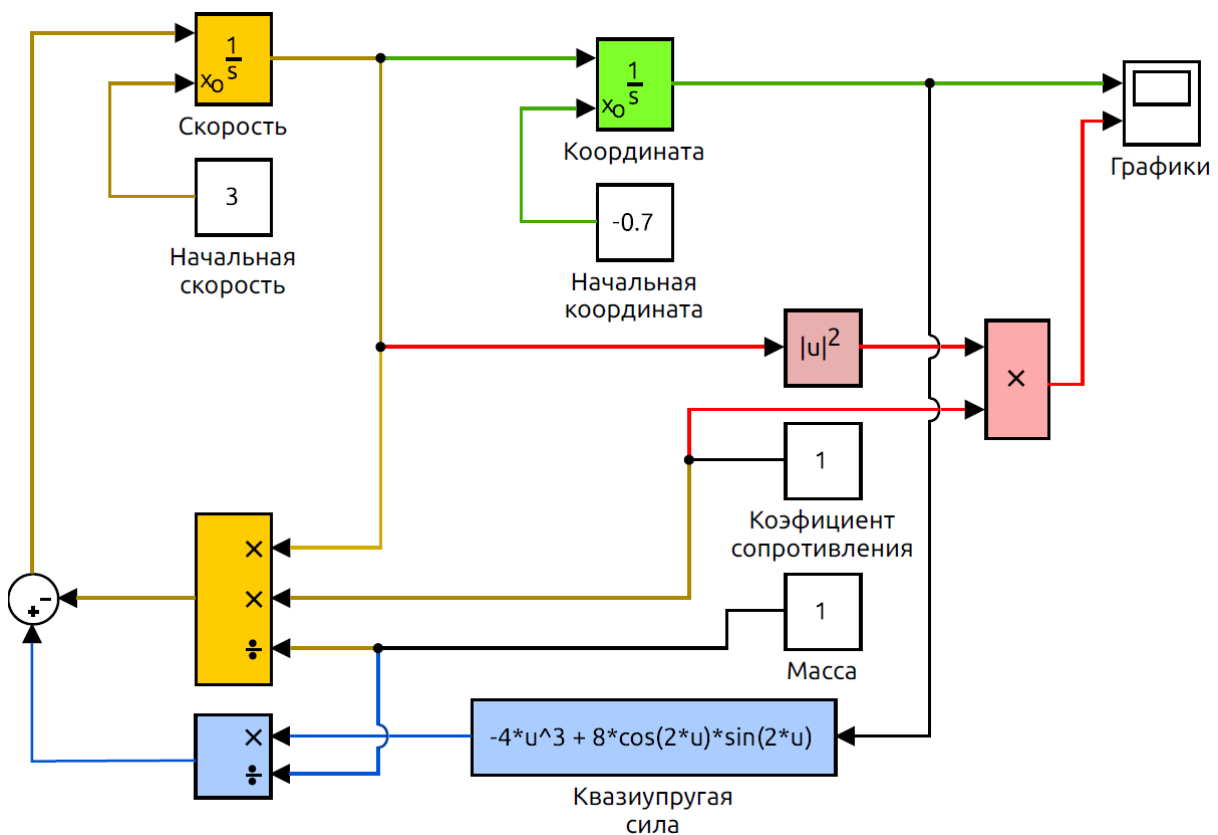


Рис.4. Финальный вид модели

**Библиографический список**

1. Шпольский Э.В. Атомная физика. Том 1. Введение в атомную физику. М.: Издание восьмое, Лань, 2010.
2. Савельев И.В. Курс общей физики. Книга 1: Механика. М.: Третье издание, Наука, Физматлит, 1998.
3. Никитин Н.Н. Курс теоретической механики. М.: Издание восьмое, Лань, 2011.
4. Colgren R. Basic MATLAB, Simulink and Stateflow. USA, Virginia: American Institute of Aeronautics and Astronautics, 2007.

# Вычислительные системы и программирование

УДК 004.42

**О. Д. Александров**

студент

**П. А. Степанов**

научный руководитель

## ПРОГРАММНОЕ СРЕДСТВО ПОИСКА АНТИПАТТЕРНОВ В ИСХОДНОМ КОДЕ НА ЯЗЫКЕ JAVA

В настоящее время достаточно сложно найти область деятельности человека, в которой не используется программное обеспечение, а любая незначительная ошибка в коде может привести к огромным финансовым потерям или даже к гибели людей. Поэтому автоматическое обнаружение ошибок в программном коде - весьма актуальная задача.

Примером ошибок, которые могут быть обнаружены автоматически, являются антипаттерны. Антипаттерны программирования представляет собой формализованные типовые ошибки, которые совершают программистами в процессе написания кода.

Современные компиляторы и интерпретаторы во многом решают задачу обнаружения ошибок. Однако это не касается ситуаций, когда исходный код с точки зрения языка корректен, однако с точки зрения логики и опыта ошибка все же присутствует (под опытом подразумевается набор правил и принципов, выработанных на практике при использовании определенной технологии, которые среди большинства программистов принято соблюдать).

Как правило, чем больше в исходных кодах подобных ошибок, тем сложнее такой код понимать, сопровождать и расширять, в результате это может привести к нежелательным результатам, начиная со срыва сроков выполнения работы и заканчивая прекращением разработки.

Большинство программных продуктов разрабатываются с учетом следующих ограничений:

- время, которое можно потратить на разработку и отладку, сильно ограничено;
- над проектом работают люди, которым, как известно, свойственно ошибаться, а значит любой программный продукт неизбежно содержит разного рода ошибки, своевременное обнаружение которых, способно повысить вероятность успешного выхода программного продукта на рынок.

Конечной целью является разработка механизмов обнаружения антипаттернов в языке программирования Java. Для достижения данной цели, необходимо решить ряд задач:

- описание антипаттернов, выбранных для рассмотрения в рамках настоящей работы;
- проектирование и разработка ПС (программное средство) способного обнаруживать описанные антипаттерны в исходных кодах на языке Java;
- тестирование разрабатываемого ПС.

Задача описания антипаттернов программирования наиболее фундаментальна для настоящей работы, на основе результатов которой, можно:

- перейти к решению остальных задач;
- использовать полученные данные для предоставления пользователю более информативной справки об ошибках, которые обнаружило программное средство;

– использовать полученные данные, как отправную точку для разрабатываемых алгоритмов;

– использовать часть полученных данных как входные данные тестов, которые будут использоваться при тестировании ПС.

Описание каждого антипаттерна программирования должно содержать ответы на следующие вопросы:

- что представляет собой антипаттерн программирования;
- какие проблемы присутствие данного антипаттерна может вызвать.

Разработанное программное обеспечение выполняет поиск антипаттернов посредством анализа абстрактного синтаксического дерева (AST). На вход программы поступает программный код, для которого строится AST. Затем выполняется анализ дерева, при котором собирается различная информация исходном программном коде (объявленные классы, функции, переменные). Поле этого модуль принятия решений, если это необходимо, выполняет поиск в дереве дополнительной информации и на основе полученных данных принимает решение о наличии или отсутствии искомого антипаттерна программирования. Краткая схема работы разработанного программного обеспечения представлена на рис. 1.

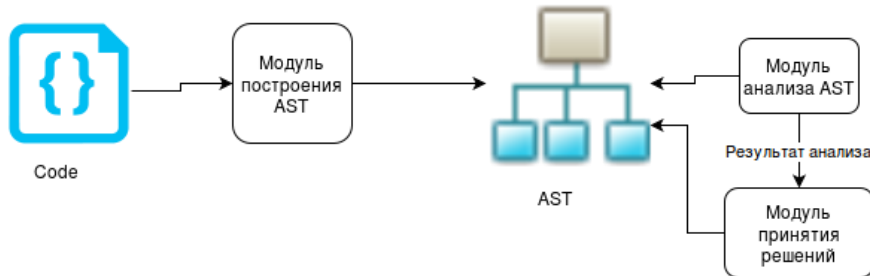


Рис. 1. Схема работы разработанной программы

В результате выполнения поставленной цели были достигнуты следующие результаты:

- составлено описание для девяти антипаттернов программирования;
- разработано и протестировано программное средство, которое обнаруживает описанные антипаттерны программирования.

Разработанное программное обеспечение обнаруживает следующие антипаттерны программирования:

- переопределение функции equals без переопределения функции hashCode;
- программирование с помощью “копировать-вставить(soru-paste)”;
- использование оператора “==” для сравнения сложных типов;
- использование в классе открытых полей;
- создание конструкторов класса, аргументы которого отличаются лишь порядком следования;
- создание множества конструкторов для реализации необязательных полей класса;
- некорректное переопределение метода equals;
- некорректное использование класса String;
- использование оператора “return” в try-catch-finally блоке.

В настоящее время, разработанное программное обеспечение не способно на конкуренцию, с такими продуктами как Findbug или Parasoft JTest, однако при дальнейшем развитии программы это может стать возможным. В первую очередь необходимо увеличивать количество обнаруживаемых антипаттернов, во вторых весьма перспективна идея добавление составляющей позволяющей использовать разработанный программный продукт через браузер.

### Библиографический список

1. Christopher Alexander, Sara Ishikawa, Murray Silverstein and etc. A Pattern Language. Oxford University Press, New York, 1977. 1218 pp.

УДК 629.735.33

**Я. В. Баранов**

магистрант кафедры компьютерных технологий и программной инженерии

**А. В. Фомин**

канд. техн. наук., доц. – научный руководитель

### СБОР, ОБРАБОТКА И ХРАНЕНИЕ ЛИЧНОЙ СТАТИСТИКИ. ПРОЕКТ TRACKALLMYLIFE

При самостоятельном освоении учебных дисциплин возникает необходимость получения высокоточной информации о том, когда человек занимался, сколько часов было потрачено, какие факторы способствовали улучшению или ухудшению обучения. На момент написания статьи не существовало готовых решений в этой области, удовлетворяющих перечисленным требованиям. Не было единой программы, которая позволяет в удобной форме (через мобильное приложение) добавлять дела и анализировать их в глобальном масштабе.

Представленная информационная система состоит из двух частей: мобильного приложения и веб-сайта, соединённых через базу данных на сервере. Мобильное приложение позволяет добавлять в систему всё, что вы делали в течение дня и выводит историю дела в виде списка (рис. 1). Для некоторых дел вы можете указывать детали (например, какую главу книги вы читали или какую лекцию смотрели). Эту информацию можно использовать для того, чтобы получить детализацию обучения по определённому курсу (рис. 2). Программа позволяет добавить структуру вашего учебного курса (или книги) в базу и удобно отслеживать её. Вы можете увидеть, сколько времени длилась некоторая лекция, сколько времени вы потратили на её разбор, было ли вам интересно, а также какие факторы вам мешали. На основе этой информации можно сделать много полезных выводов. Например, вы можете определить, какие разделы давались вам трудно, чтобы вернуться к ним в конце обучения.



Рис. 1. Мобильное приложение

	Name	Duration	Spend	Rate	AllEnj	StudyEnj
✓	1) Introduction to Chemistry	00:47	02:32	3.2	0.26	0.32
✓	2) Atomic Theory and Measurement	01:02	02:35	2.5	0	0
✓	3) The Chemists Atom	00:22	00:40	1.8	-0.10	0
✓	4) Significant Figures and the Mole	00:51	02:15	2.7	0.13	0.13
✓	5) Balancing Equations	00:58	04:10	4.3	0.25	0.25
✓	6) Empirical and Molecular Formulas	01:01	02:26	2.4	-0.34	0.06
✓	7) Chemical Reactions - Classifying Chemical Reactions	00:42	01:49	2.6	-0.11	0.08
✓	8) Chemical Reactions - Energy in Chemical Reactions	01:03	02:47	2.7	0.01	0.07
✓	9) Electromagnetic Radiation	01:04	02:39	2.5	-0.30	-0.02
✓	10) Periodic Table and Bonding	01:14	03:30	2.8	0.22	0.28
✓	11) Lewis Structures	01:17	03:01	2.4	-0.47	0.02
✓	12) Shapes of Molecules and Hybridization	01:11	03:41	3.1	0.07	0.12
✓	13) States of Matter	01:00	02:24	2.4	-0.25	0.05
✓	14) Partial Pressure	01:01	02:40	2.6	-0.06	0.11
✓	15) Vapor Pressure,	00:56	02:46	3.0	-0.19	0.11

Рис. 2. Детализация обучения

Программа также позволяет вам отразить некоторый учебный курс на календаре (рис. 3). Вы можете наглядно увидеть, сколько времени каждый день было потрачено на обучение или повторение материала. Также вы можете посмотреть список всех учебных курсов, которые вы проходили за всю вашу жизнь, и количество часов, которые были потрачены на них. Это может быть полезно с психологической точки зрения. Подводя итоги года, вы будете точно знать, что время потрачено не зря, а вложено в ваше будущее.

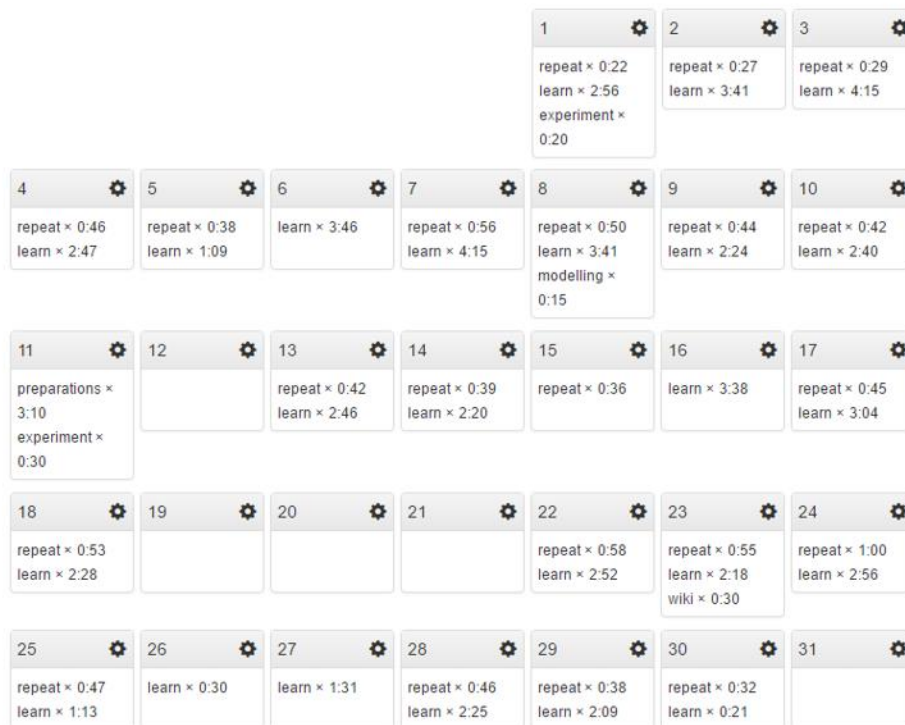


Рис. 3. Календарь обучения

Система TrackAllMyLife позволяет вам наглядно увидеть, куда уходит ваше свободное время. Это очень полезно для общего развития. Вы тратите всего 5 мин в день на работу с программой, получая огромную и высокоточную базу всех ваших дел. Ценность этой базы растёт прямо пропорционально её размеру. Одна из успешно решаемых задач – мониторинг обучения.

УДК 681.587.72

**А. О. Беляевский**

магистрант кафедры компьютерных технологий и программной инженерии

**А. А. Ключарев**

доц., к.т.н., доц. – научный руководитель

## УПРАВЛЕНИЕ НЕСКОЛЬКИМИ АНАЛОГОВЫМИ СЕРВОПРИВОДАМИ С ПОМОЩЬЮ ОДНОГО ТАЙМЕРА НА ПРИМЕРЕ ОТЛАДОЧНОЙ ПЛАТЫ STM32F429I-DISCO

Существует очень мало отраслей промышленности, в которых не применяются сервоприводы. К сервоприводам, как к категории приводов, относится множество различных регуляторов и усилителей с отрицательной обратной связью, например, гидро/электро/пневмо-усилители ручного привода управляющих элементов, однако термин «сервопривод» чаще всего (и в данной статье) используется для обозначения электрического привода с обратной связью по положению, применяемого в автоматических системах для привода управляющих элементов и рабочих органов. Существует очень мало автоматических систем, основанных на сервоприводах, включающих в себя только один сервопривод, и для каждого сервопривода необходимо формировать управляющее воздействие. При управлении сервоприводами с помощью микроконтроллера, управляющие сигналы (в основном широтно-импульсная модуляция, далее ШИМ) формируются с помощью таймеров, способных генерировать управляющий ШИМ – сигнал независимо от основной программы микроконтроллера и его прерываний. Но в таком случае зачастую встает проблема нехватки таких таймеров для управления всеми сервоприводами. Не смотря на то, что такой подход обеспечивает ряд преимуществ (в частности, практически не занимает процессорное время), он является неприменимым к задаче управления сразу большим количеством сервоприводов, поэтому, задача управления или с использованием малого количества таймеров является актуальной задачей в управлении автоматическими системами.

Сервоприводы управляются с помощью либо широтно – импульсной модуляции, либо частотно – импульсной модуляции. Так же различают аналоговые и цифровые сервоприводы. Цифровые сервоприводы работают на более высокой частоте, в связи с чем обладают рядом преимуществ по сравнению с аналоговыми: более высокая разрешающая способность, минимальные «мертвые зоны», более точное позиционирование, быстрое реагирование на команды, постоянное усилие на валу в процессе его поворота, увеличенная мощность. При этом стоят цифровые сервоприводы значительно больше, в связи с чем, в виду отсутствия необходимости в очень точном и быстром позиционировании, для написания статьи был выбран простой и дешевый аналоговый сервопривод Futabas3003 (рис. 1). Принцип работы управляющей схемы этого сервопривода, заключается в интегрировании входного сигнала, и подаче на мотор тока через усилитель, в случае наличия разницы между входным сигналом и показаниями датчика. Структурная динамическая схема сервопривода представлена на рис. 2.

Управляющая схема сравнивает показания датчика с входным воздействием, и при наличии разницы подает соответствующее напряжение на мотор, который вращает выходной вал. К выходному валу подключен датчик, и с вращением вала изменяются его показания. Как только показания датчика уравниваются с входным воздействием, вращение мотора останавливается.

Для управления сервоприводом требуется генерировать на вход сервопривода ШИМ – сигнал с определенными характеристиками. Характеристики сигнала однозначно задают угол поворота сервопривода.



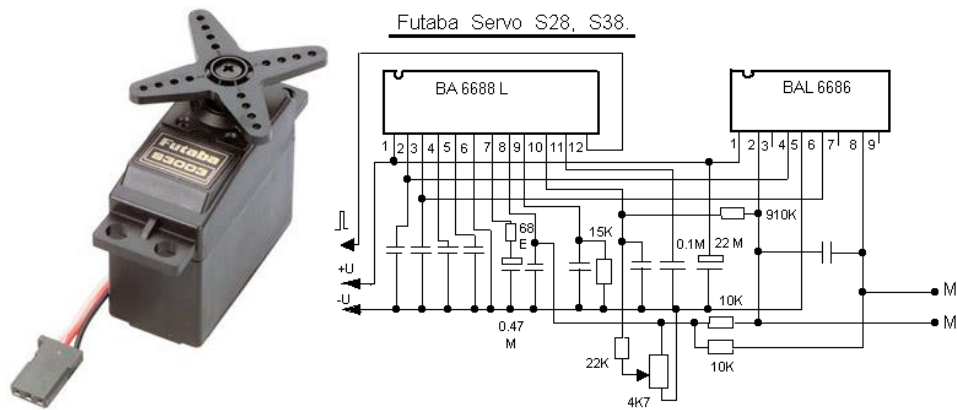


Рис. 1. Сервопривод FutabaS3003 и его управляющая схема



Рис. 2. Структурная динамическая схема сервопривода: 1– 3 провода – питание для мотора + и -, и управляющий сигнал; 2 – управляющая схема; 3 – датчик (потенциометр); 4 – электромотор; 5 – выходной вал.

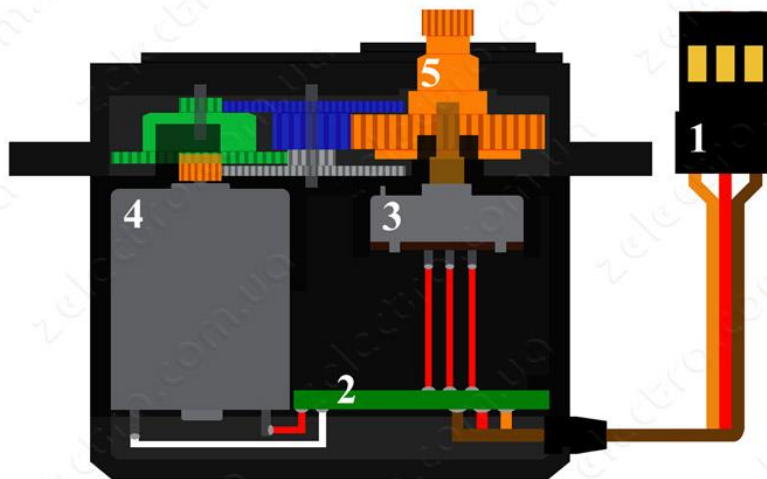


Рисунок 3- Схема сервопривода

Очень важно правильно выбрать выводы процессора для генерации ШИМ – сигнала и управления сервоприводом. Неправильный выбор выводов может привести к полностью нерабочей программе и трудно предсказуемым результатам. При работе с отладочной платой, подобной той, что использовалась при написании данной статьи (STM32F429I-DISCO), это особенно актуально, так как на выводы процессора подключено большое количество периферии, такой, как например, LCD дисплей с touch – screen. Для правильного выбора свободных выводов

необходимо тщательно изучить документацию. Задачу может сильно упростить использование программы STM32CubeMX, но это решение актуально только при работе с процессорами фирмы STMicroelectronics. В данной программе, при создании проекта необходимо в окне выбора MCU на вкладке Boardselector выбрать необходимую отладочную плату, после чего на вкладке Pinout главного окна программы, на изображении микроконтроллера будут отображены все выводы процессора, и из них можно выбрать свободные. В приведенном ниже примере использовались выводы PG2,PG3,PE2,PE3,PE4,PE5.

Следует учитывать, что слишком высокая частота вызова прерываний по таймеру может остановить основную программу почти полностью. Так же следует избегать сложных вычислений. Но если они необходимы, следует помнить о том, что ручная генерация ШИМ сигнала занимает очень много процессорного времени, и при возникновении ситуации, когда основная программа будет выполняться слишком долго, необходимо уменьшить частоту таймера, отвечающего за генерацию ШИМ – сигнала, что приведет к снижению точности позиционирования сервоприводов, или уменьшить время обработки прерывания путем оптимизации программы.

Для генерирования ШИМ – сигнала выбрана частота таймера 100 КГц, что обеспечивает достаточно маленькую ошибку позиционирования сервопривода и позволяет основной программе работать без значительных задержек.

Для расчета ошибки позиционирования сервопривода необходимы следующие параметры:

- период  $P=30$  мс;
- частота таймера  $f=100$  КГц, т. е. таймер будет срабатывать 100 раз в миллисекунду;
- число срабатываний таймера за период  $P$  равно 3000;
- ширина пика для 0 градусов:  $W_0=0,388$  мс ( $\approx 39$  тиков таймера);
- ширина пика для 180 градусов:  $W_{180}=2,14$  мс ( $\approx 214$  тиков таймера).

На интервал  $W_0-W_{180}$  придется количество точек:  $214-39=175$ , следовательно, ошибка будет равна  $180^\circ/175 = 1,029$  град.

Для реализации управления несколькими сервоприводами выполняются действия, описанные ниже. Инициализируется таймер с частотой 100 КГц, разрешается обработка прерываний таймера, таймер запускается в режиме генерирования прерываний. Описываются переменные, ограничивающие ширину пика для каждого ШИМ - сигнала, а так же инициализируется счетчик времени. Необходимо учитывать, что переменные, ограничивающие ширину пика ШИМ – сигнала, заданы в тиках таймера. Таким образом, если требуется получить ширину пика равную 1 мс, необходимо задать соответствующее 1мс значение в тиках таймера, которое равно 100 тикам в данном случае. В обработчике прерываний таймера сначала увеличивается счетчик времени, а также проверяется его значение на предмет выхода за 3000, что равно периоду в 30 мс в данном случае. Если счетчик больше этого значения, счетчик необходимо обнулить, и установить на все используемые для генерации ШИМ – сигнала выводы процессора 1. Далее, используя оператор *if*, счетчик последовательно сравнивается со всеми переменными, которые отвечают за ширину пика для каждого ШИМ-сигнала. В случае если значение счетчика равно соответствующему значению, на вывод процессора устанавливается 0. Эту программу при необходимости можно оптимизировать, однако такая оптимизация не будет рассматриваться в рамках данной статьи.

После выполнения описанных выше действий для поворота любого сервопривода на требуемый угол необходимо присвоить переменной, отвечающей за ширину пика ШИМ – сигнала, подведенного к выбранному сервоприводу, значение, соответствующее требуемому углу поворота. Но следует учитывать, что прерывания не остановлены, и если с переменными будет совершаться множество различных операций, то в прерывании может считаться промежуточное и неверное значение соответствующей переменной. Для решения этой проблемы достаточно остановить все маскируемые прерывания на время выполнения операций,

а после снова запустить. В данном примере это делалось функциями «`__disable_interrupt()`» и «`__enable_interrupt()`».

По описанной выше методике, при написании данной статьи, управлялось шесть разных сервоприводов. Экспериментальным путем было получено, что с помощью используемой в рамках данной работы платы, по описанной методике возможно формировать до 60 управляющих сигналов для сервоприводов одновременно без существенных задержек основной программы и без существенных ошибок в позиционировании сервоприводов при наличии такого количества свободных выводов процессора.

### Библиографический список

1. Широотно-импульсная модуляция [электронный ресурс] URL:<https://ejudge.lksh.ru/archive/2015/08/sc/arduino/theory/pwm.html> (дата посещения 09.06.2016).
2. Сервопривод – электропривод с обратной связью [электронный ресурс] URL:<http://www.servotechnica.spb.ru/servodrive/> (дата посещения 09.06.2016).
3. Типы сервоприводов [электронный ресурс] URL:<http://www.amg-rc.ru/node/50> (дата посещения 09.06.2016).
4. Пат. 018909 Rev 11 США. RM0090 Reference manual // STM32F405/415, STM32F407/417, STM32F427/437 and STM32F429/439 advanced ARM®-based 32-bit MCUs; October 2015.
5. Пат. 024030 Rev 8 США. STM32F427xx/STM32F429xx // Datasheet; January 2016.

УДК 629.735.33

**Д. А. Бобрышов**

магистрант кафедры вычислительных систем и сетей

**А. В. Никитин**

канд. техн. наук, доц. – научный руководитель

## РАЗРАБОТКА ТРЕХМЕРНЫХ ВЕБ-ПРИЛОЖЕНИЙ НА ЯЗЫКЕ WEBGL

В настоящее время набирает популярность новая технология WebGL отображения трехмерной графики в веб-браузерах. Вся работа веб-приложений с использованием WebGL основана на коде JavaScript, а некоторые элементы кода – шейдеры могут выполняться непосредственно на графических процессорах на видеокартах, благодаря чему разработчики могут получить доступ к дополнительным ресурсам компьютера, увеличить быстродействие. Таким образом, для создания трехмерных веб-приложений разработчики могут использовать стандартные для веб-среды технологии HTML, CSS, JavaScript и при этом также применять аппаратное ускорение графики [1].

WebGL – технология, которая поддерживается браузером по умолчанию, т.е. нет необходимости устанавливать дополнительные плагины. Поэтому веб-приложения, построенные с использованием данной технологии, будут доступны в любой точке земного шара при наличии сети Интернет вне зависимости от используемой платформы: будь то десктопы с ОС Windows, Linux, Mac, или смартфоны и планшеты, или игровые консоли [1].

На данный момент создано большое количество трехмерных веб-приложений, которые используют различные устройства ввода-вывода, например, 2D-, 3D-мыши, Leap Motion и т.д. Но из-за аппаратно-программных особенностей одни устройства могут напрямую взаимодействовать с трехмерными объектами в сцене, а другие только опосредованно. Понятия прямого и опосредованного манипулирования раскрыты в диссертации исследователя R.J. Teather [2]. Для решения проблемы необходимо разработать решение, с помощью которого станет возможным взаимодействовать с трехмерными объектами в сцене с использованием различных устройств ввода-вывода, что определяет актуальность темы исследования.

Исходя из установленной проблемы, сформулирована следующая цель работы – разработать и реализовать приемы прямого взаимодействия с трехмерными объектами в сцене с использованием различных устройств ввода-вывода в веб-браузере с использованием технологии WebGL.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Провести анализ аналогичных решений взаимодействия с трехмерными объектами в сцене.
2. Сравнить возможности популярных инструментов для создания трехмерных приложений на языке WebGL для веб-браузеров и выбрать наиболее универсальный, который подходит для множества задач, а также обладает большой функциональностью.
3. Разработать и реализовать приемы прямого взаимодействия с трехмерными объектами в сцене с использованием различных устройств ввода-вывода в веб-браузере с использованием технологии WebGL.

Для исследования взаимодействия с трехмерными объектами в сцене на языке WebGL были взяты следующие устройства ввода-вывода:

1. 2D-мышь, или обычная компьютерная мышь;
2. 3D-мышь;
3. Leap Motion.

В таблице приведены особенности известных технологий отображения трехмерной графики в веб-браузере. Из нее можно выделить следующие достоинства WebGL в сравнении с аналогами:

- высокая скорость работы за счет загрузки всего приложения изначально;
- отсутствие необходимости устанавливать дополнительные плагины (WebGL встроен в браузер по-умолчанию), что делает работу с WebGL удобнее;
- поддержка вершинных и фрагментных шейдеров, которые позволяют использовать ресурсы видеокарты, что повышает производительность работы веб-приложений.

Особенности известных технологий отображения трехмерной графики в веб-браузере [3]

	Особенности
VRML	<ul style="list-style-type: none"> <li>– Формат файла для описания 3D-сцены;</li> <li>– стандарт ISO / IEC 14772-1: 1997;</li> <li>– не поддерживает современные 3D-технологии, такие как NURBS;</li> <li>– необходима установка дополнительного плагина в браузере</li> </ul>
X3D	<ul style="list-style-type: none"> <li>– Является заменой VRML;</li> <li>– стандарт ISO / IEC 19775/19776 / 19777;</li> <li>– имеет более продвинутый API, дополнительные форматы кодирования данных и компонентную архитектуру в сравнении с VRML;</li> <li>– предназначен для разработки как 3D веб-приложений, так и в десктопных 3D-приложениях</li> </ul>
X3DOM	<ul style="list-style-type: none"> <li>– Декларативный язык для описания 3D-сцены;</li> <li>– встроен в браузер по-умолчанию, т.е. не требуется установка дополнительного плагина;</li> <li>– предназначен для тесной интеграции со стандартными веб-технологиями, такими как AJAX;</li> <li>– предназначен для интеграции с DOM;</li> <li>– для визуализации сцены X3DOM определяет модель нейтрализации неисправности (сначала инициализируются предпочтительные рендеры, а в случае необходимости используются альтернативные технологии);</li> <li>– поддерживает импорт X3D (или VRML) сцены</li> </ul>
XML3D	<ul style="list-style-type: none"> <li>– Декларативный язык для описания 3D-сцены;</li> <li>– разрабатывался с целью расширения существующих свойств HTML, а именно вложение 3D-контента в веб-страницу;</li> <li>– предназначен для интеграции с DOM</li> </ul>
WebGL	<ul style="list-style-type: none"> <li>– Кроссплатформенная библиотека;</li> </ul>

- спецификация WebGL 1.0;
- разработана на основе OpenGL ES 2.0;
- для отображения 3D-графики используется тег <Canvas> из стандарта HTML5;
- загрузка веб-приложения изначально;
- поддержка вершинных и фрагментных шейдеров, которые использует графический процессор для рендеринга 3D-графики;
- написана на Javascript;
- встроена в браузер по-умолчанию, т.е. не требуется установки дополнительного плагина

Для упрощения работы с WebGL используются различные Javascript-библиотеки и фреймворки. Наиболее популярные – Javascript-библиотека Three.js и игровой движок Unity3D. Проанализировав оба инструмента, была выбрана Javascript-библиотека Three.js, потому что есть доступ к коду библиотеки, что позволяет повысить производительность за счет меньшего объема программного кода приложения.

На рисунке приведена схема для разработки веб-приложений на WebGL.

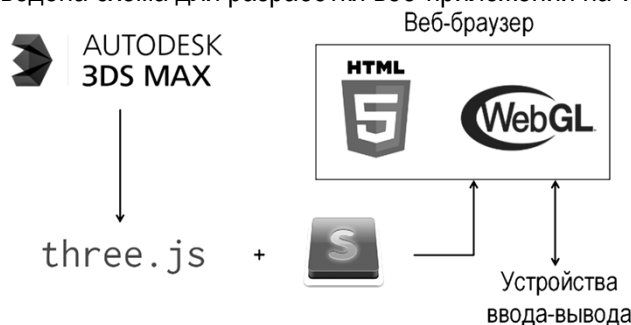


Схема состоит из следующих частей:

- Autodesk 3Ds Max предназначен для моделирования трехмерных объектов;
- Javascript-библиотека Three.js – для разработки веб-приложений на WebGL;
- текстовый редактор sublimtext 3 – для удобного написания кода;
- стандарт HTML5 – для отображения трехмерной графики в браузере;
- WebGL – для создания и рендеринга трехмерных сцен в веб-браузере;
- устройств ввода-вывода, с помощью которых осуществляется взаимодействие с объектами сцены веб-приложения.

Использование 3D-мыши и Leap Motion позволяют напрямую взаимодействовать с 3D-объектами, но из-за аппаратно-программных особенностей алгоритмы прямого взаимодействия с объектами будут отличаться. В случае с 2D-мышью можно управлять объектами только опосредованно, так как в веб-браузере сцена и объекты трехмерные, а взаимодействие с ними происходит с помощью устройства, которое обрабатывает двумерные координаты. Из этого следует, что каждому устройству ввода-вывода должен соответствовать свой алгоритм взаимодействия с трехмерными объектами в сцене.

В процессе исследования рассматриваются следующие операции с объектами:

1. Перемещение;
2. Вращение;
3. Объединение объектов.

Выводы: на данный момент проведен анализ аналогов WebGL, инструментов для разработки веб-приложений на WebGL; разработаны и реализованы алгоритмы перемещения и вращения трехмерных объектов в сцене на WebGL для 2D-мыши.

Планируется разработать и реализовать прием объединения объектов сцены с использованием 2D-мыши и приемы прямого взаимодействия с трехмерными объектами сцены с использованием 3D-мыши и Leap Motion на языке WebGL. Планируемая научная новизна исследования заключается в разработке новых подходов взаимодействия с трехмерными

объектами сцены с использованием различных устройств ввода-вывода в веб-браузере на языке WebGL.

Практическая значимость исследования заключается в возможности использования приемов, разработанных в ходе исследования, для решения задач в таких сферах деятельности, как архитектура, машиностроение и промышленность, например, при реализации операций сборки-разборки для электронной документации сложных объектов и т.д.

### Библиографический список

1. Коичи Мацуда, Роджер Ли WebGL: программирование трехмерной графики: пер. с англ. М.: ДМК Пресс, 2015. 494 с.
2. Teather R. J. Comparing 2D And 3D Direct Manipulation Interfaces : дис. ... магистр: Programme in Computer Science/ York University. Toronto, 2008. 101 с.;
3. Alun Evans 3D graphics on the web: A survey// Computers & Graphics. 2014. № 41. С. 43–61 с.

### УДК 519.8

**В. В. Боженко, А. Н. Игошин**

студенты кафедры проблемно-ориентированных вычислительных комплексов

**Г. С. Евсеев**

канд. техн. наук, доц. – научный руководитель

### СРАВНЕНИЕ МЕТОДОВ ОДНОМЕРНОГО ПОИСКА ЭКСТРЕМУМА

Рассматриваются условия, при которых сложность метода одномерного поиска экстремума может быть существенно уменьшена.

Задача одномерного поиска экстремума сводится к построению последовательности точек  $x_1, x_2, \dots, x_N$ , которые сходятся к точке экстремума функции  $F(x)$ . На итерации  $j$  рассматривается интервал  $[a_j, b_j]$ , в котором выбирается очередная точка  $x_j$ . После этого границы интервала пересчитываются. Процесс завершается при выполнении условия  $b_j - a_j < \varepsilon$ , где  $j$  – номер итерации,  $\varepsilon$  – требуемая точность определения искомой точки [1]. Начальные границы интервала –  $[a_0, b_0]$ .

Количество точек  $x$  не всегда равно числу итераций, поэтому сложность метода оценивается величиной  $N$ , а не числом итераций.

Для известных методов поиска значения  $N$  и  $\varepsilon$  связаны соотношением:

$$N = \alpha \log_2 \frac{1}{\varepsilon}, \quad (1)$$

где  $\alpha$  – коэффициент, зависящий от метода поиска. В частности, для метода дихотомии  $\alpha = 2$ , а число итераций равно  $\frac{N}{2}$ . Для метода золотого сечения  $\alpha \cong 1,45$ , а число итераций равно  $N - 1$ .

Для случая, когда число итераций задано заранее, доказано, что оптимальным является метод чисел Фибоначчи, сложность которого при большом числе итераций близка к сложности метода золотого сечения [2].

Таким образом, для минимизации унимодальных функций наименьшую сложность обеспечивает метод чисел Фибоначчи, для которого коэффициент  $\alpha \cong 1,45$ . Представляет интерес поиск условий, накладываемых на минимизируемую функцию, при которых существует метод поиска экстремума с коэффициентом  $\alpha < 1.45$ .

Если функция симметрична относительно экстремума, то существует метод, с помощью которого можно уменьшить сложность метода дихотомии в 2 раза. При этом на первом шаге вычисляются значения функции на концах отрезка  $[a, b]$ :

$$F_1 = F(a); F_2 = F(b).$$

Затем на каждой итерации отрезок  $[a, b]$  делится пополам точкой:

$$x = \frac{a+b}{2}$$

и в зависимости от значений функции на концах отрезка отбрасывается одна из его частей. Так, если  $F_1 < F_2$ , то  $a = x$ , иначе  $b = x$ . Таким образом, на каждой итерации происходит всего одно вычисление значения функции.

На рис.1 показаны зависимости сложности упомянутых методов от требуемой точности. Для метода улучшенной дихотомии коэффициент  $\alpha = 1$ , т.е. сложность этого метода в 2 раза меньше, чем сложность дихотомии и в 1,45 раза меньше, чем сложность метода золотого сечения.

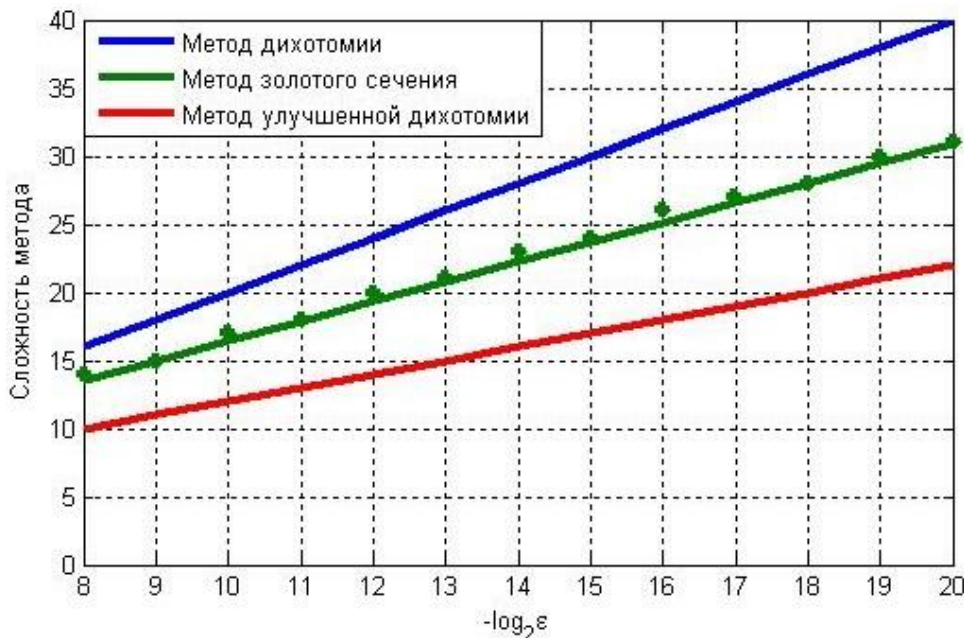


Рис. 1. Графики зависимости сложности метода от требуемой точности

Сформулируем условия применимости алгоритма улучшенной дихотомии. Несмотря на то, что метод улучшенной дихотомии имеет меньшую сложность, чем другие методы, стоит отметить, что он не всегда работает корректно. Например, для функции, которая задана следующей формулой

$$F(x) = \frac{x}{\sigma} e^{\left(-\frac{x^2}{2\sigma^2}\right)}, x \geq 0, \sigma > 0,$$

данный метод не всегда дает верный результат. В частности, при  $x_{max} < \frac{a+b}{2}$  поиск сходится к точке  $\frac{a+b}{2}$  (рис.2).

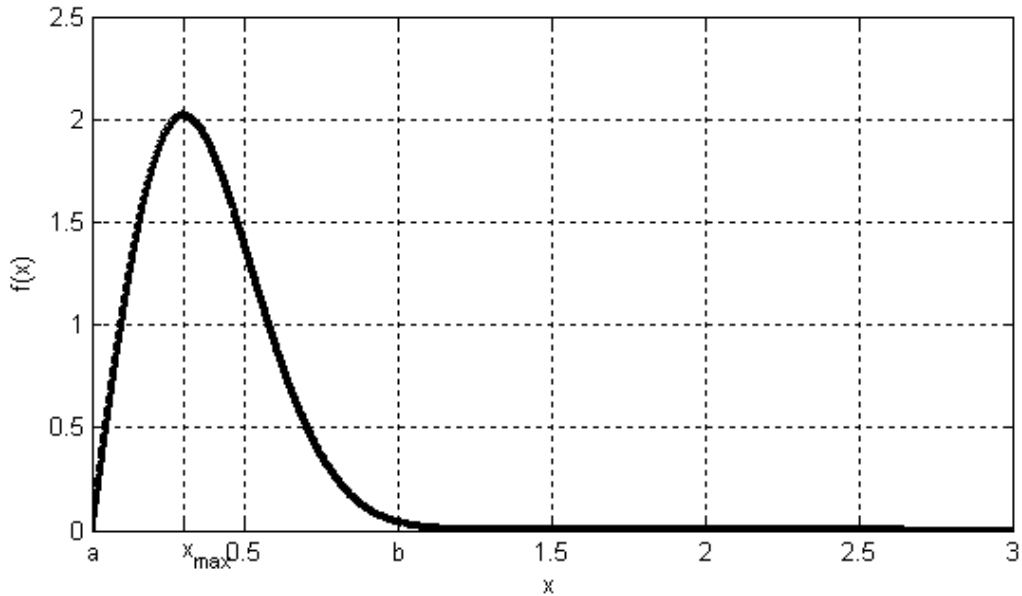


Рис. 2. Пример некорректного применения алгоритма улучшенной дихотомии

Полный набор условий, когда этот метод работает корректно, пока не сформулирован. Вместе с тем, можно доказать, что достаточным условием сходимости метода улучшенной дихотомии является симметричность унимодальной функции относительно точки экстремума. Условие симметричности функции может быть обобщено следующим образом: если  $x_{max} \leq \frac{a+b}{2}$ , то  $F(x_{max} - \varepsilon) \geq F(x_{max} + \varepsilon)$ , а если  $x_{max} \geq \frac{a+b}{2}$ , то  $F(x_{max} - \varepsilon) \leq F(x_{max} + \varepsilon)$ . При моделировании выполнение этих условий всегда обеспечивало сходимость метода улучшенной дихотомии. Математическое доказательство этого факта еще не получено.

В заключение, отметим, что, использование априорной информации о свойствах минимизируемой функции позволяет существенно уменьшать сложность одномерного поиска, поэтому дальнейшие исследования этих свойств являются актуальной задачей.

**Библиографический список**

1. Васильев Ф.П. Методы оптимизации. М.: Факториал Пресс, 2002. 824 с.
2. Пантелеев А.В., Летова Т.А. Методы оптимизации в примерах и задачах: учеб. пособие. М.: Высшая школа, 2005. 544 с.

УДК 621.389

**А. Ф. Виноградов, А. С. Клеченов**

студенты кафедры проблемно-ориентированных вычислительных комплексов

**Е. П. Виноградова**

ст. преподаватель – научный руководитель

**КОМБИНИРОВАННЫЕ УСИЛИТЕЛИ ЗВУКОВОЙ ЧАСТОТЫ**

Человеческий слух способен воспринимать звуковые колебания в диапазоне от 20 Гц до 20 КГц. Комбинированные усилители звуковой частоты – это устройства, обеспечивающие усиление сигнала в диапазоне от 50 Гц до 10 КГц, поскольку их основное применение – усиление звучания музыкальных инструментов и голоса.



Проблема усиления звучания музыкальных инструментов существовала еще до появления электроники и электричества вообще. Особенно это касалось «тихих» инструментов, терявшихся в оркестре или в больших залах, прежде всего – гитары. Для нее изобретали вставки в корпус, дополнительные голосовые отверстия и раструбы, специальные резонаторы... Увы, никакого особого эффекта это не давало, и как только появились электрические средства звукоусиления, их тотчас попытались приспособить к гитаре. Звучание обычной акустической гитары даже при исполнении ритм-партии было очень тихим, потребовались радикальные технические решения.

Современные усилители для музыкальных инструментов делятся на три класса: гитарные, басовые и клавишные. В каждом классе существуют линейки моделей от самых дешевых, маломощных и простых до дорогих, мощных и сложных. Прежде всего, усилитель предназначен для определенного инструмента. Например, частотный диапазон гитары, включая все гармоники, значительно уже слышимого диапазона частот, поэтому для усиления гитары не требуется обеспечивать работу усилителя, в полном диапазоне частот. В силу этого, общий диапазон усиления ограничен: от 80...100 Гц до 8... 10 кГц. Гитарные усилители существуют в двух основных формах. Комбинированный (или комбо) усилитель содержит электронную схему усилителя и акустическую воспроизводящую систему в одном корпусе. В другой форме: усилитель расположен в отдельном корпусе и подключается к акустической системе с помощью соединительного провода. Отдельный усилитель называется голова, а акустическая система – кабинет. Голова располагается сверху кабинета и формирует так называемый гитарный стек (усилительный стек). Различают понятия «полустек» (half-stack), состоящий из усилителя и кабинета, и «полный стек» (full-stack), представляющий собой усилитель с подключенной парой кабинетов, либо комбоусилитель с подключенной дополнительной акустической системой [4]. Современные гитарные усилители разделяются на традиционные ламповые и полупроводниковые; некоторые производители выпускают гибридные модели с ламповой обработкой сигнала и полупроводниковым выходом. Гитаристы, прежде всего в США, отдают предпочтение ламповым усилителям. Стоит заметить, что, хотя ламповые усилители выпускаются в разных странах, лампы для них в настоящее время производятся только в России и Китае. Соотношение ламповых и полупроводниковых усилителей на различных рынках разнятся. Важным функционалом комбинированного усилителя является возможность одновременного подключения нескольких музыкальных инструментов. Комбинированные микшеры-усилители применяются как в качестве установленных стационарно, позволяя экономить место, так и в качестве мобильных и носимых. Идея объединить в одном корпусе усилитель и микшер не нова. Действительно, любой гитарный комбоусилитель, по своей сути, является комбинацией как усилителя мощности звуковой частоты, так и микшера. Несмотря на «одноканальность» гитарного предусилителя, он имеет все атрибуты микшера: регуляторы чувствительности входа, выхода, эквалайзер, возможность посылки сигнала на пружинный ревербератор или процессор эффектов, разрывы цепи прохождения сигнала для возможности подключения внешних приборов динамической обработки. И надо сказать, что попытки включить в один гитарный комбоусилитель кроме гитары еще и микрофон предпринимались всегда. Другое дело, что при этом практически не удавалось добиться хорошего звучания. Комбинированные микшеры-усилители как раз и предназначены для решения такого рода звукоусилительных задач. За рубежом эти устройства называются power mixer. Логически продолжая принцип «все в одном», производители часто встраивают в комбинированные микшеры-усилители эквалайзер, включенный перед усилителем мощности, и процессор эффектов. Для обеспечения гибкости конфигурации и универсальности отдельные функциональные узлы имеют свои индивидуальные входы и выходы. Подключение к ним и, тем самым, разрыв внутренних сигнальных цепей позволяет использовать эквалайзер, процессор эффектов, усилитель мощности и т. д. как отдельные, функционально независимые приборы. Усилитель мощности, встраиваемый в комбинированный прибор, как правило, двухканальный (т. е. стереофонический), мощностью

100...300 Вт, что и определяет область их применения в смысле размера озвучиваемого пространства. Конфигурация микшера предполагает наличие нескольких микрофонных и инструментальных каналов с типовым для небольших микшеров сервисным набором, в который входят: регулятор чувствительности входа; 3...4-полосный эквалайзер; регулятор уровня посылы на эффекты (один или несколько); регулятор уровня посылы на линию мониторов (одну или несколько); регулятор панорамы/баланса; фейдер или вращаемый регулятор уровня. Упрощенная структура комбинированного усилителя звуковой частоты представлена на рис. 1.

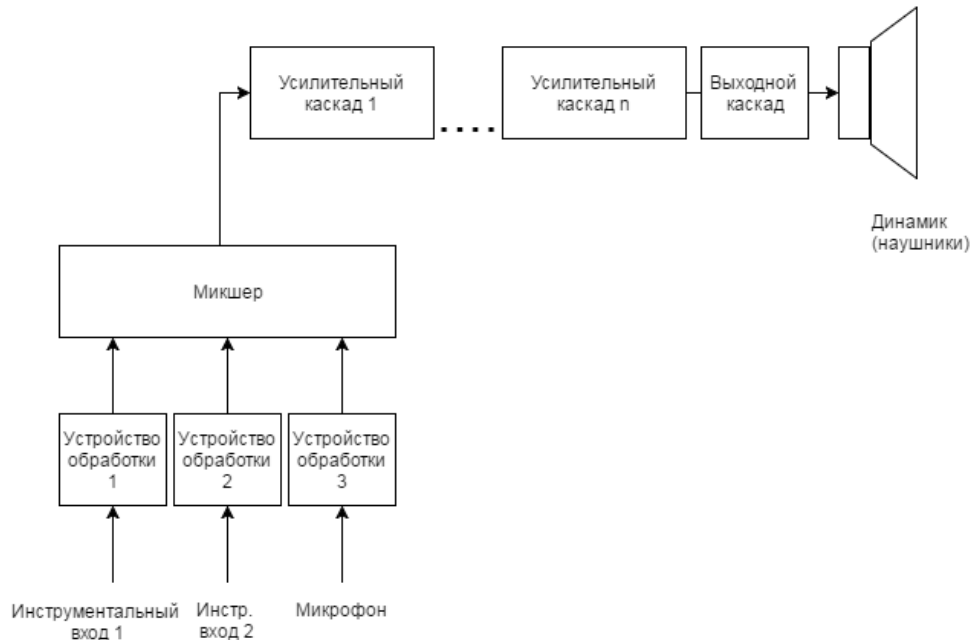


Рис.1. Структурная схема комбинированного усилителя звуковой частоты

Обычно мастер-секция имеет регуляторы уровня главного выхода (он же "мастер", MAIN, MAINS, MIX и т. д.) Схемы, по которым изготавливаются отдельные узлы комбинированных приборов, как правило, уже прошли хорошую "обкатку", и производитель имеет опыт их производства и эксплуатации в других функциональных устройствах. В одном комбинированном приборе часто объединяются узлы и блоки разных торговых марок, т. е. изготовленные разными производителями, каждый из которых имеет свое know-how, и достиг в своей области или в классе устройств, которые он производит, хороших результатов. Размещение в одном корпусе разных узлов и блоков ведет к сокращению количества кабельных межблочных соединений и входных/выходных буферных схем. Также, в интегрированном устройстве легко достигается согласование по уровню сигнала между узлами и блоками, и отпадает необходимость в индикаторах уровня входа/выхода. И, как следствие, повышаются надежность и качество. Наличие общих корпуса и блока питания снижает как цену прибора, так и массогабаритные показатели. Система принудительного охлаждения современного усилителя с успехом решает проблему охлаждения и других устройств, размещенных в одном с ним корпусе. Не все, но большинство комбинированных микшеров-усилителей имеет встроенные процессоры эффектов. Управление ими максимально упрощено – пользователю предлагается набор из нескольких десятков предустановок, содержащих наиболее популярные алгоритмы обработки: реверберацию, задержки, хорус(эхо), спецэффекты. Поскольку микшер-усилитель может быть подключен к акустическим системам самых разных типов, его силовые выходы снабжаются разъемами «на все случаи жизни»: speakon, 1/4" джек, а иногда даже имеются выходы под «бананы» или винты-зажимы для «голых» проводов от акустических систем. На рис. 2 приведен внешний вид комбинированного усилителя фирмы Marshall [2].



Рис.2. Комбинированный басовый усилитель.

#### Полупроводниковые (аналоговые) усилители

Транзисторный усилитель дешевый и, благодаря низкому выходному сопротивлению, легко стыкуется с любыми низкоомными громкоговорителями, т. е. неравномерность полного сопротивления громкоговорителя практически не влияет на коэффициент усиления. Транзисторные усилители не имеют настолько сильный микрофонный эффект, как ламповые, и не требуют регулярной замены активных радиоэлементов, однако, если они выходят из строя, то ремонт в полевых условиях требует много времени и инструментов.

#### Ламповые усилители (аналоговые)

По сравнению с транзисторными УНЧ, выходное сопротивление лампового выходного каскада высоко: до нескольких десятков Ом, поэтому качество звучания зависит от качества ламп выходного каскада и от качества выходного трансформатора, который является согласующим. Также узнаваемость звучания придает подбор головки громкоговорителя и конструкция корпуса (толщина материала, с открытой или закрытой задней стенкой), а также схемные решения и темброблок. Кажущаяся ненадежность и примитивность ламповой конструкции на практике оборачивается простотой ремонта. Практически любой отказ лампового УНЧ (кроме выхода из строя силового и/или выходного трансформаторов) может быть устранен в полевых условиях [1]. [https://ru.wikipedia.org/wiki/%D0%92%D0%B8%D0%BA%D0%B8%D0%BF%D0%B5%D0%B4%D0%B8%D1%8F:%D0%A1%D1%81%D1%8B%D0%BB%D0%BA%D0%B8\\_%D0%BD%D0%BO\\_%D0%B8%D1%81%D1%82%D0%BE%D1%87%D0%BD%D0%B8%D0%BA%D0%B8](https://ru.wikipedia.org/wiki/%D0%92%D0%B8%D0%BA%D0%B8%D0%BF%D0%B5%D0%B4%D0%B8%D1%8F:%D0%A1%D1%81%D1%8B%D0%BB%D0%BA%D0%B8_%D0%BD%D0%BO_%D0%B8%D1%81%D1%82%D0%BE%D1%87%D0%BD%D0%B8%D0%BA%D0%B8)

#### Цифровые (моделирующие) усилители

С появлением микропроцессоров и цифровой обработки сигналов в конце 1990-х гг. были разработаны «моделирующие усилители», которые могут симулировать звук многих известных ламповых усилителей без использования ламп в схеме. Новые усилители со специальными процессорами и программным обеспечением могут эмулировать ламповый звук классических усилителей. Главный недостаток цифровой обработки сигнала состоит в том, что теряются некоторые оттенки, нюансы звукоизвлечения и звучания различных гитар, в то время, как для опытного гитариста каждый инструмент «звучит» по-своему. При этом само звучание остается довольно хорошим.

Разница в «поведении» усилителей. У лампы прямолинейный участок вольт-амперной характеристики значительно длиннее, проще говоря, ее непросто перегрузить. Это очень хорошо для аппаратуры, предназначенной для воспроизведения фонограмм, в том числе бытовой, но, как известно, перегрузка по входу для рок-гитаристов является одним из наиболее популярных эффектов. Поэтому ламповые усилители чаще применяют музыканты в тех стилях, где перегрузка не применяется, например, джаз, кантри. Очень хорошо ламповые усилители подходят для бас-гитары, где перегрузка совсем ни к чему. Лампа нуждается в прогреве, поэтому, чтобы можно было играть сразу по выходе на сцену, в ламповых усилителях имеется выключатель standby, который отключает анодную цепь, оставляя включенным подогрев катода.

Звук в усилитель не идет, но при включении цепи усилитель мгновенно заработает. Исполнителю только надо помнить об этой особенности устройства ламповых усилителей [3].

Выбор усилительного устройства обычно осуществляется с учетом его основных технических характеристик, к которым относятся: частотный диапазон, выходная мощность, коэффициент нелинейных искажений, отношение сигнал/шум, потребляемая мощность и массогабаритные показатели. С учетом изложенных выше соображений, а также принимая во внимание многообразие представленных моделей различного ценового диапазона, проблема выбора комбинированного усилителя звуковой частоты может быть успешно решена.

### Библиографический список

1. *Морган Д.* Ламповые усилители. Второе издание, переработанное. М.: ДМК-Пресс, 2016 . 983 с.
2. <http://marshallcombo.ru/>
3. <http://guitar-news.ru/tube-amp-combo.html>
4. [http://www.muzbar.ru/articles/Kak\\_vybirat\\_instrumentalnyj\\_usilitel\\_v\\_chastnosti\\_gitarnyj\\_usilitel/](http://www.muzbar.ru/articles/Kak_vybirat_instrumentalnyj_usilitel_v_chastnosti_gitarnyj_usilitel/)

УДК 004.42

**Ю. Д. Гнутова**

магистрант кафедры компьютерных технологий и программной инженерии

**А. А. Ключарев**

доц., к.т.н., доц. – научный руководитель

## ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ СОЗДАНИЯ РАБОЧИХ ПРОГРАММ ДИСЦИПЛИН НА КАФЕДРЕ

Планирование и организация учебного процесса невозможна без мощной информационной поддержки. Слабое развитие информационных систем в образовательной отрасли делает актуальной задачу разработки автоматизированных информационных систем поддержки планирования и организации учебного процесса.

Обязательным элементом документационного обеспечения образовательного процесса в университете являются учебно-методические комплекты дисциплин, в состав которых входят утвержденные рабочие программы дисциплин по всем реализуемым формам обучения. Под рабочей программой дисциплины (далее – РПД) понимается нормативный документ, определяющий объем, содержание, порядок изучения и преподавания каждой учебной дисциплины базового учебного плана, а также способы контроля результатов ее усвоения, соответствующий требованиям направлений подготовки и учитывающий специфику подготовки студентов по избранному направлению. Вследствие больших объемов разрабатываемых материалов появилась необходимость использования информационных технологий для автоматизации процессов создания, проверки и утверждения рабочих программ дисциплин.

Главной целью работы является проектирование, реализация и последующее внедрение автоматизированной информационной системы для автоматизации создания рабочих программ на кафедре №43 «Кафедра компьютерных технологий и программной инженерии» ГУАПа.

Качественное проектирование является основной предпосылкой эффективного функционирования системы при постоянном совершенствовании ее обеспечивающих и функциональных составляющих. Вследствие того, что планируется продолжение работы над проектом другими разработчиками, проектированию был уделен значительный промежуток времени. В разрабатываемой системе используется шаблон проектирования модель-представление-контроллер (далее MVC), который облегчает управление сложными структурами путем разделения приложения на модель, представление и контроллер, вследствие чего

разработчики могут легко изменять отдельные части приложения, не затрагивая другие. В архитектуре MVC представление отвечает за пользовательский интерфейс (например, текст, поля ввода), модель предоставляет данные и правила бизнес-логики, а контроллер обеспечивает взаимодействие между моделью и представлением.

В состав автоматизированной информационной системы включена реляционная база данных, которая содержит все текстовые и количественные данные, относящиеся к разрабатываемым рабочим программам дисциплин. Автоматически в систему рабочих программ выгружается вся информация из базы данных, относящаяся к данной дисциплине, как текстовая, так и количественная. Данные выгружаются как в виде абзацев текста, списков, так и в табличной форме.

Разрабатываемая система почти полностью автоматизирует процесс создания рабочих программ. Например, приложение выводит сообщение об ошибке преподавателю, если количество часов, введенных разработчиком, не соответствует заданной трудоемкости при автоматическом расчете трудоемкости дисциплины [1].

С системой можно работать в локальном и сетевом режимах [2]. Сетевой вариант работы обеспечивает централизованное хранение, загрузку и просмотр рабочих программ дисциплин на сервере кафедры. Локальный вариант предназначен для автоматизированного закрепления рабочих программ дисциплин за преподавателями.

Автоматизированная информационная система имеет разграниченный доступ к данным. В системе на данный момент есть два вида прав доступа: составитель(преподаватель) и методист. Составителю доступен список всех разрабатываемых им РПД. При выборе одного из РПД для разработки, составитель имеет возможность выбирать, какой раздел рабочей программы дисциплины он будет разрабатывать в следующий момент. Он имеет право изменять все доступные для редактирования данные и автоматически генерировать печатную форму рабочей программы дисциплины.

Преимущества создания рабочих программ дисциплин с помощью разрабатываемой автоматизированной информационной системы:

- генерация рабочих программ дисциплин происходит точно по утвержденному шаблону;
- существенно снижаются трудозатраты авторов рабочих программ, основная задача которых состоит в том, чтобы в заданные разделы, пункты и подпункты документа ввести содержательную часть программы;
- существенно снижается бумажный документооборот, связанный с процессом согласования и утверждения документов;
- полностью согласованная рабочая программа утверждается в электронном виде, в базе данных, в соответствии с регламентом;
- хранение утвержденных рабочих программ в базе данных обеспечивает их доступность и целостность;
- упрощается процедура обновления рабочих программ и контроля данного процесса.

Разработанный функционал автоматизированной системы создания рабочих программ дисциплин на кафедре обеспечивает:

- выгрузку данных из внутренней БД системы в пользовательский интерфейс с учетом всех требуемых правил представления данных;
- автоматическую загрузку информации о закреплении рабочих программ дисциплин за преподавателями из XML-документа;
- разграничение прав доступа к системе. Например, преподаватели могут разрабатывать рабочие программы дисциплин, которые закреплены только за ними, а методист может обновлять данные о закреплении рабочих программ дисциплин за преподавателями;
- использование механизма сессий. Обращение к сессиям в объектно-ориентированном стиле при работе с одной рабочей программой, что исключает ошибки при за-

писи данных в базу данных системы;

- сохранение связанных данных из одной формы в разные таблицы БД;
- удобный пользовательский интерфейс системы и наглядность предоставления информации;
- динамическое добавление полей на страницу(асинхронные аjax-запросы);
- возможность автоматического формирования документа.

Разработанное техническое задание и интерфейс системы, реализованный базовый функционал являются основой для создания автоматизированной информационной системы создания рабочих программ дисциплин на кафедре.

### Библиографический список

1. Порядок разработки, оформления и утверждения рабочей программы дисциплины по образовательной программе бакалавриата, специалитета, магистратуры и аспирантуры [Электронный ресурс] URL:[http://download.guar.ru/local/iuko/rds/rdo\\_guar\\_smk\\_2.74.doc](http://download.guar.ru/local/iuko/rds/rdo_guar_smk_2.74.doc) (Дата посещения 01.06.2016).
2. Автоматизированная информационная система для разработки основных образовательных программ (ООП) на основе федеральных государственных образовательных стандартов (ФГОС) – АИС РООП [Электронный ресурс] URL: [http://aisroop.ru/page/product\\_use](http://aisroop.ru/page/product_use) (Дата посещения 27.05.2016).

УДК 629.735.33

**Д.В. Горелик, А.А. Чурбанов**

студенты кафедры вычислительных систем и сетей

**А.В. Гордеев**

проф., докт. техн. наук – научный руководитель

## СОЗДАНИЕ ДИСТАНЦИОННОЙ СИСТЕМЫ ОБУЧЕНИЯ НА ПЛАТФОРМЕ MOODLE

В последние годы получил широкое распространение термин E-learning, означающий процесс обучения в электронной форме через сеть Интернет или Интранет с использованием систем управления обучением. Программное обеспечение для E-learning представлено как простыми статическими HTML страницами, так и сложными системами управления обучением и учебным контентом.

Система дистанционного обучения (СДО) – это комплексная система, охватывающая широкий набор программно-технических коммуникаций, методических указаний и организационных мер, позволяющих обеспечить учащихся образовательной информацией. Она также включает проведение необходимой проверки полученных знаний и имеет ряд неоспоримых преимуществ перед другими стационарными образовательными формами. К ним можно отнести следующие: возможность обучаться в комфортных условиях - дома или в офисе; индивидуальный подход к каждому учащемуся; развитие таких важных качеств, как организованность и самостоятельность; значительная экономия времени и сил; доступность общения с преподавателем – возможность задать интересующие вопросы в индивидуальном режиме; персональные сроки, учитывающие способности учащегося.

Система дистанционного обучения можно структурировать следующим образом:

1. Программа обучения.
2. Тесты, опросы.
3. Пользовательские инструкции, регламенты.
4. Поддержание актуальности информации, предоставляемой СДО.
5. Предоставление учебной литературы.
6. Планирование обучения и регулярная аттестация учащихся.

7. Ведение электронного журнала успеваемости студента.
8. Предоставление доступа к учебным материалам.

Средства организации СДО делятся на 4 группы[1]:

1. Авторские программные продукты (Authoring Packages), специально разработанные для преодоления тех затруднений, с которыми сталкиваются преподаватели при использовании языков программирования. Эти программы обычно позволяют преподавателю самостоятельно разрабатывать учебный контент на основе визуального программирования.

2. Системы управления (Content Management systems-CMS) системы управления контентом позволяют создавать каталоги графических, звуковых, аудио- видео-, текстовых и других файлов и манипулировать ими. Такая система представляет собой базу данных, снабженную механизмом поиска по ключевым словам, позволяющим преподавателю или разработчику курсов быстро найти то, что ему нужно.

3. Системы управления обучения (Learning Management Systems – LMS).

Для того чтобы система считалась LMS (в русскоязычной терминологии используется аббревиатура СДО – «система дистанционного обучения»), она должна автоматизировать такие задачи, как предоставление учебного контента нужным людям в нужное время, контроль использования учебных ресурсов, администрирование отдельных слушателей и групп, организация взаимодействия с преподавателем, отчетность и т.д. Они представляют собой платформу для развертывания e-Learning, но в ряде случаев может использоваться и для администрирования традиционного учебного процесса.

4. Системы управления учебным контентом (Learning Content Management Systems – LCMS). Последние два года развивается новый класс систем, реализующих управление учебным контентом (Learning Content Management System, LCMS). В отличие от LMS, подобные системы концентрируются на задачах управления содержанием учебных программ, а не процессом обучения, и ориентированы не на менеджеров и студентов, а на разработчиков контента, специалистов по методологической компоновке курсов и руководителей проектов обучения. В основе LCMS лежит концепция представления содержания обучения как совокупности многократно используемых учебных объектов со своей целевой аудиторией и определенным контекстом использования.

На кафедре вычислительных систем и сетей проводятся работы по созданию ряда учебных курсов по дисциплинам, которые изучают студенты. Для этой цели используется программное обеспечение Moodle.

Moodle – аббревиатура от Modular Object-Oriented Dynamic Learning Environment (модульная объектно-ориентированная динамическая обучающая среда). Moodle – это свободная система управления обучением, ориентированная прежде всего на организацию взаимодействия между преподавателем и учениками, хотя подходит и для организации традиционных дистанционных курсов, а также поддержки очного обучения.

Основными параметрами, которые помогли сделать выбор в пользу CMS Moodle являются:

- переведен на десятки языков, в том числе и русский, и используется в 197 странах мира;
- написан на php с использованием sql-базы данных;
- moodle распространяется в открытом исходном коде – это дает возможность адаптировать систему под особенности конкретного образовательного проекта, а при необходимости и встроить в нее новые модули;
- система создает и хранит портфолио каждого обучающегося: все сданные им работы, все оценки и комментарии преподавателя к работам, все сообщения в форуме;
- преподаватель может создавать и использовать в рамках курса любую систему оценивания. все отметки по каждому курсу хранятся в сводной ведомости;

- позволяет контролировать «посещаемость», активность студентов, время их учебной работы в сети;

- модульная структура системы обеспечивает простоту использования системы для обучающихся и преподавателей;

- возможность эффективного обмена опытом с помощью открытого сообщества, состоящего из более 50000 пользователей системы и более 3000 внедренных систем дистанционного обучения во всем мире.

Возможности Moodle, актуальные для администраторов:

- работает без модификаций в unix, linux, windows, mac os x, netware и любой другой операционной системе, поддерживающей php (который поддерживается любым хостинг-провайдером);

- проектируется как набор модулей и позволяет гибко добавлять или удалять элементы на различных уровнях;

- легко обновляется от версии к версии (он имеет внутреннюю систему для обновления собственной базы и восстановления);

- требует только одну базу данных и может быть использован совместно с другими приложениями;

- включает базу данных широкого назначения, которая поддерживает различные типы баз данных;

Недостатком Moodle, можно считать не полную адаптацию для работы с большими проектами. Он отлично справляется с работой для колледжей и университетов небольших и средних размеров, но не может эффективно работать с крупными образовательными учреждениями или служить способом проведения всех занятий. Чем больше студентов имеют доступ к платформе, тем медленнее становится система. Это может быть крайне неудобно для студентов, когда они пытаются ответить на вопросы тестов или просто пытаются получить доступ к содержимому курса.

Пользователи часто жалуются на неприятности, связанные с индивидуальными настройками системы. Программа предназначена для доставки контента, но было бы неплохо, если бы система предоставляла больше возможностей для человеческого взаимодействия, чтобы помочь студентам и сотрудникам эффективно общаться. Возможности оценки также ограничены в Moodle. Оценками слушателя можно оперировать только внутри курса. Нет возможности составить итоговую ведомость, но есть возможность добавлять непроверенные файлы, HTML-коды, содержащие JavaScript, что может быть потенциально использовано с целью получения прав администратора.

Еще одним недостатком системы Moodle является то, что в ней отсутствует понятие академической группы. Группы в Moodle существуют не для управления правами доступа к курсам, а для разделения групп слушателей в одном курсе.

В Moodle так же существует гибкая система ролей, которые назначаются конкретным пользователям на определенных уровнях контекстов (или пространств) системы («система», «категория», «курс» и т.д.). При этом пользователю предоставляются права, определенные этой ролью, не только для текущего уровня, но и всех «более низких» уровней в системе. Например, если пользователю на уровне некоторого курса назначена роль «Студент», то он также будет иметь эту роль во всех элементах и блоках этого курса.

В системе также существует механизм переопределения ролей. Например, один и тот же пользователь в одном курсе может быть Преподавателем, а в другом – Студентом. Кроме этого, пользователи могут иметь и глобальную роль в системе. В этом случае им могут делегироваться возможности по администрированию курса. Принципиальное различие ролей Преподавателя и Студента заключается в режиме отображения контрольных мероприятий. Для студента предлагается выполнить задания.



Преподавателю, а также Ассистенту предоставляется возможность проверить результаты студентов, но не выполнить задание. Ассистент без права редактирования имеет возможность только просмотра тех материалов, которые предоставили студенты (или результатов тестов). Это, например, позволяет легко подключать студентов старших курсов к учебному процессу младших курсов, просто повысив роль студента до Ассистента именно для этого курса. Системный администратор присваивает права доступа вновь зарегистрированных участников системы.

В этом инструменте, позволяющем разрабатывать системы СДО, очень большое разнообразие инструментов, позволяющих в разной форме предоставлять студентам учебный материал. Для нашего курса мы решили использовать следующие: урок (лекция), задания.

Урок используется для представления разветвленной информации. Этот инструмент требует иметь структурированный материал и последовательность его представления. Настройки могут оказаться достаточно сложными. Позволяет после каждого блока материала вставлять контрольные вопросы. В зависимости от ответа можно повторять теоретический блок, либо осуществлять переходы на другие блоки.

Задания - используется для сбора работ студентов, оценки и обеспечения обратной связи на задания. Выполнение заданий возможно в режиме on-line и off-line. Для полного функционирования и максимально полезного использования необходимо установить сроки выполнения и максимальные оценки, собрать задания, оценить, прокомментировать и обеспечить обратную связь. Так же в этот инструмент мы интегрируем такие программы как Virtual Box, LibreOffice. Они необходимы для выполнения лабораторных работ, предусмотренных в курсах. Четкой инструкции, каким именно образом это возможно сделать, нет, поэтому мы активно ищем возможные пути решения этого вопроса.

Самой основной задачей работы этого курса является настройка связи между встраиваемыми программами и студентами. Эта связь должна быть настроена так, чтобы студент имел возможность выполнить лабораторные работы непосредственно в системе Moodle. Речь идет о возможности запускать на сервере необходимое программное обеспечение и отображать на стороне клиента (студента) окно той программы, которую он запустил. Такими программами должны быть Oracle VirtualBox и OpenOffice.org. Для этого необходимо организовать работу сервера таким образом, чтобы студент, заходя под своим логином, попадал в отдельно созданный и закрепленный только за ним инструмент для выполнения лабораторных работ, и при этом все вычисления производились на сервере, а результат этих вычислений отображался на web-станции в moodle.

Moodle создавался в Linux с использованием Apache, MySQL и PHP.

Apache – свободный веб-сервер.

PHP-язык сценариев PHP, скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений.

MariaDB – ответвление СУБД MySQL, разрабатываемое сообществом.

Изучение пакета Moodle позволило сделать вывод, что поставленные задачи создания учебных курсов по дисциплинам: “Корпоративные сети со службой каталога” и “Открытые сети” могут быть реализованы таким образом, чтобы пользователь не только изучал предмет и проходил тестирование, но и выполнял лабораторный практикум удаленно на сервере с Moodle.

### **Библиографический список**

1. Готская И.Б., Жучков В.М. Кораблев А.В. [http://ra-kurs.spb.ru/2/0/2/1/?id=13#\\_Тoc177795514](http://ra-kurs.spb.ru/2/0/2/1/?id=13#_Тoc177795514) (дата обращения 21.04.2016)

УДК 681.883.074

**Е. К. Григорьев, А.С. Журавлев, И.А. Юдин**

студенты кафедры проблемно-ориентированных вычислительных комплексов

**В. А. Ненашев**

ассистент – научный руководитель

## СПОСОБ ОПРЕДЕЛЕНИЯ РАЗМЕРА ЗОНЫ ПОИСКА РАДИОЛОКАЦИОННЫМИ СРЕДСТВАМИ ДЛЯ РЕШЕНИЯ ЗАДАЧ ПО СПАСЕНИЮ ЛЮДЕЙ СТРУКТУРАМИ МЧС В ЧРЕЗВЫЧАЙНЫХ СИТУАЦИЯХ

В данной статье рассматривается алгоритм определения размера зоны поиска для движущегося объекта в некотором заданном пространстве с помощью радиолокационных средств, находящихся на борту летательного аппарата. Способ основывается на вычислении вероятности накрытия объекта, рассчитанный по двумерному закону распределения в полярной системе координат. Особое внимание уделяется рассмотрению зависимостей между размерами зоны обнаружения и параметрами движущегося объекта.

Тема посвящена определению минимального размера зоны поиска движущегося объекта по заданной вероятности обнаружения в передней зоне обзора летательного аппарата (ЛА). В качестве движущегося объекта в зоне бедствия могут выступать, обломки тонущего корабля, катер, шлюпка, люди и т.п. Способ имеет широкое практическое применение в реальных условиях для решения прикладных задач при обнаружении объектов. Способ основан на идее сужения размеров зоны поиска для движущегося объекта, что является жизненно важным фактором для обеспечения безопасности в случае природных катаклизмов, катастроф и может применяться при решении задач в интересах МЧС России.

Сформулируем задачу следующим образом. Летательный аппарат движется к объекту, при этом рассчитывается дальность до объекта, относительно ЛА. Через время  $t$ , при достижении расчетной точки для движущегося объекта, на дальности  $L$  осуществляется алгоритм радиолокационного обзора в передней полусфере по курсу движения ЛА, в которой и происходит обнаружение объекта. Координаты центра зоны поиска объекта на плоскости относительно системы координат  $XOY$ , связанной с ЛА, положим равными  $(X=L, Y=0)$ , а размеры зоны по дальности и по азимуту обозначим через  $(\Delta L, \Delta\alpha)$ , соответственно (рис. 1). Для описанного процесса обнаружения найдем размеры зоны поиска объекта с заданной вероятностью накрытия  $P_{накр}$ . Для этого опишем математическую модель процесса обнаружения объекта по априорной информации.

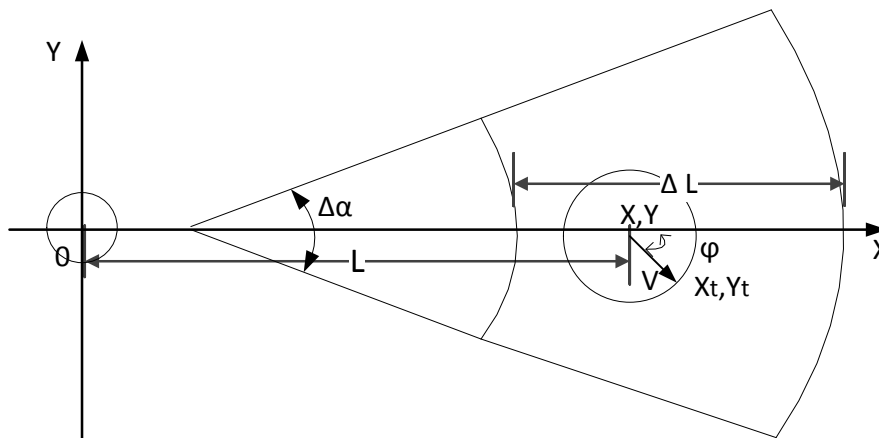


Рис. 1. Модель движущегося объекта и зоны поиска

За время  $t$  объект движется в неизвестном направлении  $\varphi$  со скоростью  $V$ . Начальные координаты объекта  $(X, Y)$  определяются с некоторой случайной ошибкой, при расчете координат объекта в течение времени полета  $t$  также происходит накопление ошибки. Обусловленной особенностями навигационной системы ЛА, поэтому к моменту начала поиска координаты объекта на плоскости  $XOY$  относительно ЛА являются случайными векторами. Будем считать, что к моменту начала поиска начальное положение объекта находится в точке  $(X_0, Y_0)$  на плоскости  $XOY$  и определяется двумерным нормальным законом распределения [1]:

$$f_0(X_0, Y_0) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{1}{2}\left(\frac{(X_0 - L)^2}{\sigma_x^2} + \frac{Y_0^2}{\sigma_y^2}\right)\right), \quad (1)$$

где  $\sigma_x^2$  и  $\sigma_y^2$  - дисперсии ошибок координат объекта в зоне поиска, которые обусловлены неточностями первоначального определения координат объекта и ЛА, ошибками при вводе координат объекта в навигационную систему и ошибки самой навигационной системы ЛА[2], а также системы, определяющей координаты объекта в течение времени полета до расчетной точки включения активного поиска объекта в зоне  $(\Delta L, \Delta \alpha)$ .

Размеры зоны поиска будут наибольшими, если в качестве скорости  $V$  взять максимальную скорость  $V_{max}$ , а  $\varphi$  считать случайной величиной, равномерно распределенной в интервале  $(-\pi, \pi)$ :

$$f_t(X_t, Y_t) = \frac{1}{4\pi^2\sigma_x\sigma_y} \times \int_{-\pi}^{\pi} \exp\left(-\frac{1}{2}\left(\frac{(X_0 - L - V_{max}t \cos(\varphi))^2}{\sigma_x^2} - \frac{(Y_0 - V_{max}t \sin(\varphi))^2}{\sigma_y^2}\right)\right) d\varphi, \quad (2)$$

где  $L$  – дистанция до объекта.

Рассмотрим частный случай выражения (2), а именно, положим что  $\sigma_x = \sigma_y = \sigma$ , тогда имеем

$$\begin{aligned} f_t(X_t, Y_t) &= \frac{1}{4\pi^2\sigma^2} \int_{-\pi}^{\pi} \exp\left(-\frac{1}{2}\left(\frac{X_0^2 + L^2 + V_{max}^2 t^2 \cos^2(\varphi) - 2X_0L + 2LV_{max}t \cos(\varphi)}{\sigma^2} \times \right.\right. \\ &\quad \left.\left. \times \frac{-2X_0V_{max}t \cos(\varphi) + Y_0^2 + 2Y_0V_{max}t \sin(\varphi) - V_{max}^2 t^2 \sin^2(\varphi)}{\sigma^2}\right)\right) d\varphi = \\ &= \frac{1}{4\pi^2\sigma^2} \exp\left(-\frac{X_0^2 + L^2 - 2X_0L + Y_0^2 + V_{max}^2 t^2 \cos^2(\varphi) - V_{max}^2 t^2 \sin^2(\varphi)}{2\sigma}\right) \times \\ &\quad \times \int_{-\pi}^{\pi} \exp\left(-\frac{2LV_{max}t \cos(\varphi) - 2X_0V_{max}t \cos(\varphi) + 2Y_0V_{max}t \sin(\varphi)}{2\sigma}\right) d\varphi = \\ &= \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(X_t - L)^2 + Y_t^2 + (V_{max}t)^2}{\sigma^2}\right) I_0\left(\frac{V_{max}t \sqrt{(X_t - L)^2 + Y_t^2}}{\sigma^2}\right). \end{aligned} \quad (3)$$

Зона поиска объекта, при обнаружении с помощью радиолокационных средств, задается в полярных координатах  $(\rho_t, \alpha_t)$ , где  $\rho_t$  – дальность до объекта,  $\alpha_t$  – азимут,  $\alpha_t = 0$  вдоль оси  $(OX)$ . Перейдем от декартовой системы координат  $XOY$  к полярной [2]:

$$\begin{cases} \rho_t = \sqrt{X_t^2 + Y_t^2}, \\ \alpha_t = \arctg \frac{Y_t}{X_t}, \end{cases} \quad \begin{cases} X_t = \rho_t \cos \alpha_t, \\ Y_t = \rho_t \sin \alpha_t. \end{cases} \quad (4)$$

получаем выражение

$$f_t(\rho_t, \alpha_t) = f_t(X_t, Y_t) \begin{vmatrix} dX_t & dY_t \\ d\rho_t & d\alpha_t \end{vmatrix} \cdot \begin{vmatrix} d\rho_t & d\alpha_t \\ dX_t & dY_t \end{vmatrix} \quad (5)$$

Подставив выражение (4) в выражение (5), получаем:

$$f_t(\rho_t, \alpha_t) = \frac{\rho_t}{2\pi\sigma^2} \exp\left(-\frac{\rho_t^2 - 2\rho_t L \cos \alpha_t + L^2 + (V_{\max} t)^2}{2\sigma^2}\right) \times \\ \times I_0\left(\frac{V_{\max} t \sqrt{\rho_t^2 - 2\rho_t L \cos \alpha_t + L^2}}{\sigma^2}\right), \quad (6)$$

которое и является исходным для расчета минимальной необходимой зоны поиска объекта, то есть следует рассчитать такие параметры  $\Delta L$  и  $\Delta \alpha$ , которые бы удовлетворяли соотношению (7) с заданной вероятностью «накрытия» движущегося объекта  $P_{\text{накр}}$ .

$$P_{\text{накр}} = 2 \int_{L - \frac{\Delta L}{2}}^{L + \frac{\Delta L}{2}} \int_0^{\Delta \alpha} f_t(\rho_t, \alpha_t) d\rho_t d\alpha_t, \quad (7)$$

где  $I_0$  – функция Бесселя нулевого порядка.

Выражение (7) определяет размеры зоны поиска неоднозначно, а именно, существует множество пар  $(\Delta L, \Delta \alpha)$  которые, удовлетворяют выражению (7). Поэтому на размеры зоны в полярных координатах можно наложить дополнительные условия. Например, фиксирование числа каналов поиска по дальности определяет параметр  $\Delta L$ , а фиксирование времени поиска (при сканировании заданной зоны радиолокационным лучом), ограничивает  $\Delta \alpha$ .

Зададим фиксированное число каналов по дальности, определив тем самым  $\Delta L$ , например  $\Delta L = 1000, 2500, 5000$  м и вероятность накрытия  $P_{\text{накр}} = 0,1 : 0,9$  определив при этом диапазон параметра  $\Delta \alpha$ .

Для этого произведем вычисления в пакете компьютерного моделирования MATLAB. Графики моделирования процесса (рис. 2)

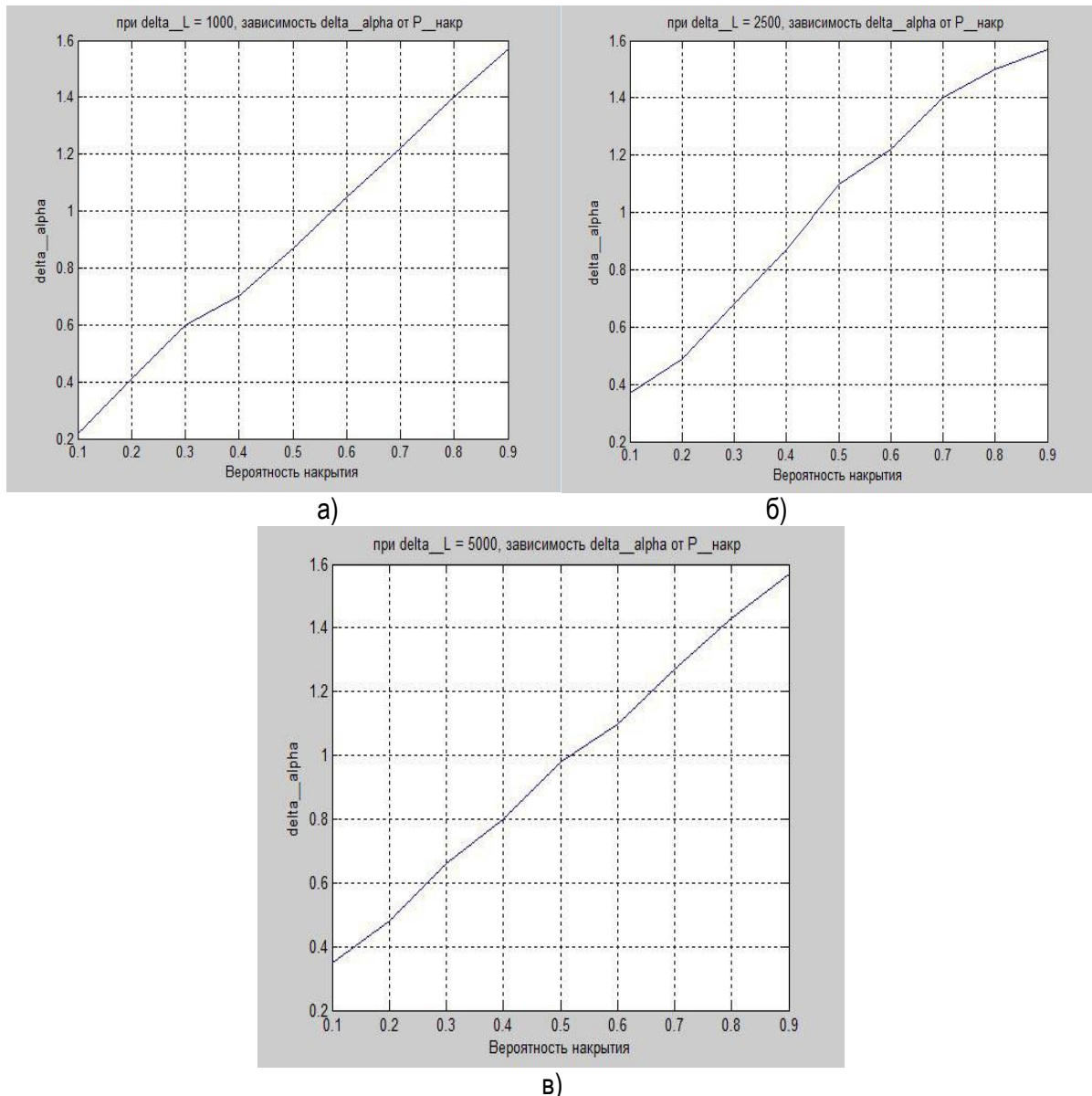


Рис.2. Зависимость параметра  $\Delta\alpha$  от вероятности обнаружения цели при а)  $\Delta L=1000$ ; б) 2500; в)  $\Delta L=5000$

Исходя из графиков, видно, что с увеличением величины  $\Delta L$  растет вероятность обнаружения цели, а значит, имея заданную априорную информацию об объекте, можно подобрать оптимальные наименьшие значения  $\Delta L$  и  $\Delta\alpha$ .

На графике видно что при увеличении  $\Delta\alpha$  получаем большие значения вероятности накрытия, однако в данном случае при  $\Delta L=2500$  большая вероятность накрытия достигается при меньших  $\Delta\alpha$ .

Данный алгоритм применим в реальных условиях для своевременного поиска и спасения людей в зоне чрезвычайных ситуаций, поскольку позволяет сократить время обнаружения.

Другим важным аспектом способа является тот факт, что при выполнении поиска в передней зоне обзора движение летательного аппарата возможно скорректировать, прежде чем зона бедствия будет достигнута или идентифицирована как зона чрезвычайной ситуации. Этот факт позволяет сэкономить время полета к зоне бедствия, по сравнению с другими методами определения размера зоны поиска, например в режиме бокового обзора при котором необходимо выполнять разворот. Кроме того, корректируя движение полета к объекту зоны бедствия, дальность до объекта уменьшается и тем самым увеличивается отношения сигнал/шум, что позволяет выполнить идентификацию зоны бедствия с меньшей вероятностью ошибки.

**Библиографический список**

1. *Быков В. В.* Цифровое моделирование в статистической радиотехнике. М.: Советское радио, 1971.328 с.
2. *Шелухин О.И. и др.* Моделирование информационных систем. М.: Радиотехника, 2005. 368 с.
3. *Шелета А.П.* Определение зоны поиска надводного объекта по данным предварительного целеуказания.// Информационные–управляющие системы.2012. №4(59).

УДК 681.518

**М. Р. Гусманова**

студент кафедры вычислительных систем и сетей

**Л.А. Мироновский**

д-р техн. наук, проф. - научный руководитель

**КОМПЬЮТЕРНЫЙ АНАЛИЗ ОБУСЛОВЛЕННОСТИ МАТРИЦ ФРОБЕНИУСОВА ТИПА**

В линейной алгебре, теории управления и теории моделирования важную роль играют матрицы Фробениуса [1-3] (другие названия: матрицы, сопровождающие характеристический полином; сопровождающие матрицы; companion matrices). Свое название они получили в честь немецкого математика Фердинанда Фробениуса. В частности, в теории управления получили распространение четыре Фробениусовы канонические формы систем [1]:

1. строчная управляемая кф;
2. столбцовая наблюдаемая кф;
3. столбцовая управляемая кф;
4. строчная наблюдаемая кф.

Классическая Фробениусова матрица имеет характерную структуру (рис. 1):

Ее правый верхний блок равен единичной матрице, а последняя строка образована коэффициентами ее характеристического полинома

$$|pE - A| = p^n + a_{n-1}p^{n-1} + \dots + a_1p + a_0. \tag{1}$$

В процессе исследования Фробениусовых матриц было обнаружено существование  $2^{n-1}$  форм Фробениусовых матриц. Так как обнаруженные матрицы отличаются от классических Фробениусовых матриц, будем называть их *F*-матрицами.

*F*-матрицы получаются путем перестановки коэффициентов последней строки матрицы Фробениуса, показанной на рис. 1.

$$F = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-2} & -a_{n-1} \end{bmatrix}$$

Рис. 1. Классическая Фробениусова матрица.

В частности, для  $n=4$  получаем следующие 8  $F$ -матриц:

$$\begin{aligned}
 F_0 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 & -a_3 \end{bmatrix}; & F_7 &= \begin{bmatrix} -a_3 & 1 & 0 & 0 \\ -a_2 & 0 & 1 & 0 \\ -a_1 & 0 & 0 & 1 \\ -a_0 & 0 & 0 & 0 \end{bmatrix}; \\
 F_1 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -a_3 & 1 \\ -a_0 & -a_1 & -a_2 & 0 \end{bmatrix}; & F_6 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ -a_2 & -a_3 & 1 & 0 \\ -a_1 & 0 & 0 & 1 \\ -a_0 & 0 & 0 & 0 \end{bmatrix}; \\
 F_2 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -a_2 & -a_3 & 1 \\ -a_0 & -a_1 & 0 & 0 \end{bmatrix}; & F_5 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -a_3 & 1 & 0 \\ -a_1 & -a_2 & 0 & 1 \\ -a_0 & 0 & 0 & 0 \end{bmatrix}; \\
 F_3 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -a_1 & -a_2 & -a_3 & 1 \\ -a_0 & 0 & 0 & 0 \end{bmatrix}; & F_4 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -a_3 & 1 & 0 \\ 0 & -a_2 & 0 & 1 \\ -a_0 & -a_1 & 0 & 0 \end{bmatrix}.
 \end{aligned}
 \tag{2}$$

Непосредственной проверкой легко убедиться, что характеристические полиномы всех этих матриц одинаковы и имеют вид (1).

В то же время их числа обусловленности различны. Целью настоящей работы является исследование этих чисел с помощью компьютерного моделирования.

Заметим, что матрицы  $F_0 - F_7, F_1 - F_6, F_2 - F_5, F_3 - F_4$  симметричны относительно побочной диагонали, следовательно, числа обусловленности этих пар матриц равны.

Сопоставим исходную матрицу  $F_0$  с линейной динамической системой с одним входом и одним выходом, заданную в пространстве состояний следующим описанием:

$$\dot{X} = FX + bu; \quad y = cX,$$

где  $F = F_0; b = [0 \ 0 \ 0 \ 1]^T, c = [1 \ 0 \ 0 \ 0]$ .

Схема моделирования этой системы представляет собой цепочку из  $n=4$  интеграторов, охваченных обратными связями с коэффициентами  $-a_0, -a_1, -a_2, -a_3$ . Соответствующий сигнальный граф схемы моделирования показан на рис. 2 слева вверху.

Сигнальные графы для остальных  $F$ -матриц также приведены на рис. 2. Несмотря на то, что матрицы имеют различные сигнальные графы, у них одна и та же передаточная функция  $W(p)$ :

$$W(p) = \frac{1}{p^4 + a_3p^3 + a_2p^2 + a_1p + a_0}, \tag{3}$$

где в знаменателе стоит характеристический полином матрицы  $F_0$

$$|pE - F_0| = p^4 + a_3p^3 + a_2p^2 + a_1p + a_0. \quad (4)$$

Для исследования  $F$ -матриц была написана программа на языке программирования Matlab, с помощью которой можно получить все  $F$ -матрицы для исходной классической Фробениусовой матрицы и провести анализ обусловленности этих матриц (рис. 3).

Ниже приведены результаты нескольких компьютерных экспериментов по анализу обусловленности  $F$ -матриц.

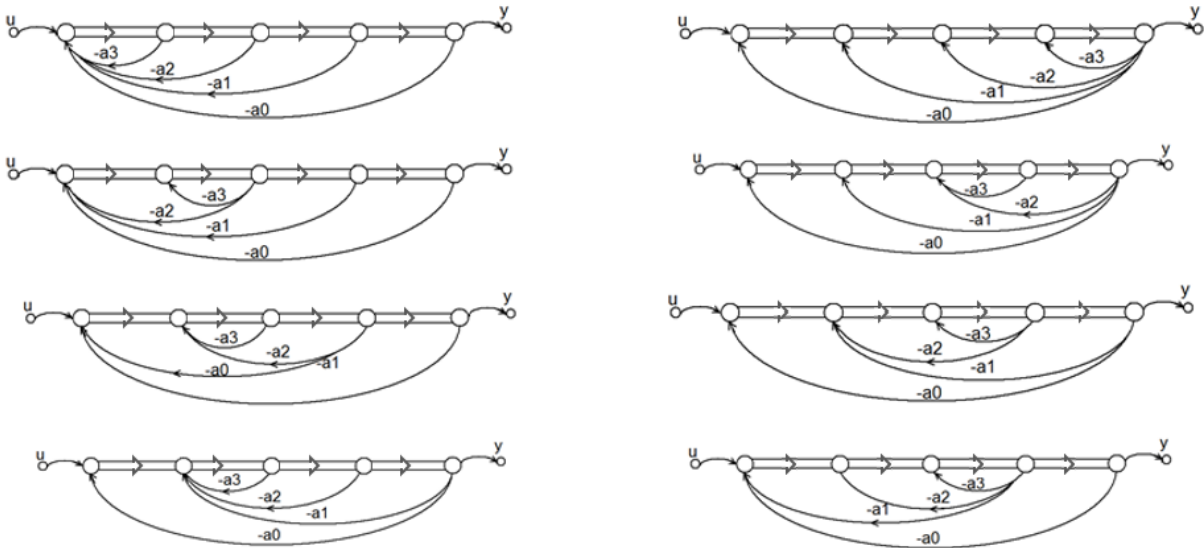


Рис. 2. Сигнальные графы схемы моделирования к матрицам  $F_0 - F_7$

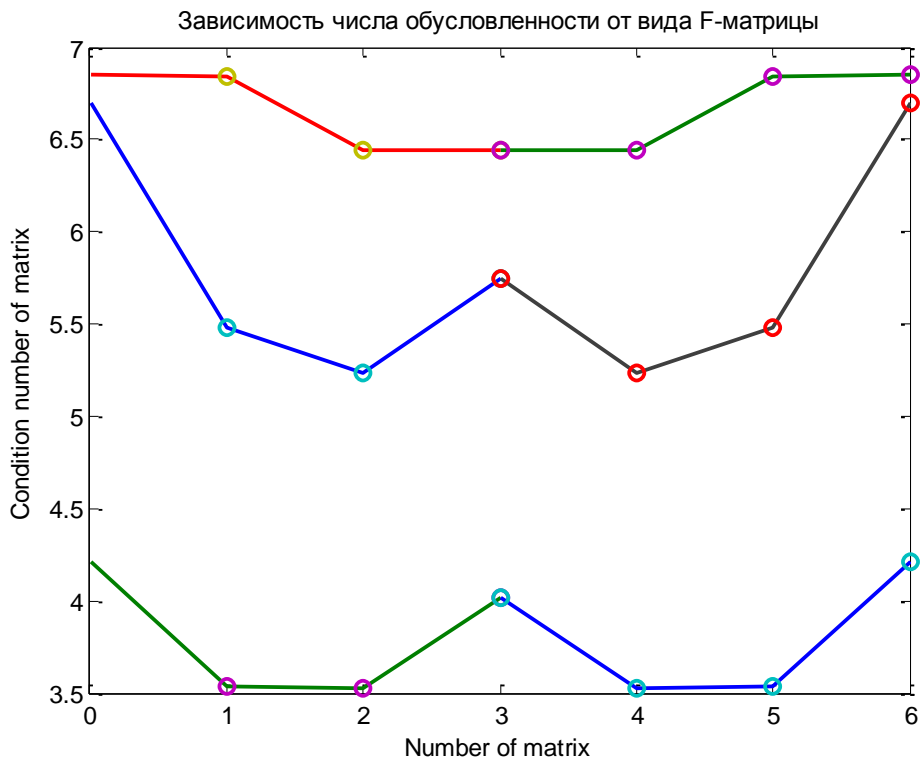


Рис.3. График зависимости числа обусловленности от вида  $F$ -матрицы



На рис. 3 показана зависимость числа обусловленности от вида  $F$ -матриц. Три классические Фробениусовы матрицы генерировались с помощью MATLAB-функции `rand`, которая возвращает случайные числа, распределенные по равномерному закону в интервале  $[0; 1]$ . Из рис. 3 видно, что полученные  $F$ -матрицы лучше обусловлены, нежели исходные Фробениусовы матрицы.

На рис. 4 показан случай, когда корни характеристического полинома образуют гармоническую прогрессию:

$$\left[ -1 \quad -\frac{1}{2} \quad -\frac{1}{3} \quad -\frac{1}{4} \quad -\frac{1}{5} \right]. \quad (5)$$

Код программы:

```
p=5 %порядок системы
h=1 %количество матриц
K=zeros(1,1); %матрица чисел обусловленности
V=zeros(1,1); %матрица количества вариантов ФМ
S=zeros(1,1); %матрица сингулярных чисел
q=1;
w=1;
l=0;
t=1;
V(q,w)=1;
H=poly([-1 -(1/2) -(1/3) -(1/4) -(1/5)])
H=-1*H;
P=flip(H)
P(p+1) = [ ]
for g=1:1:h %количество рассматриваемых матриц
F=[[zeros((p-1),1),eye(p-1)];P] %задаем матрицу
%randi([-10,-1],1,p)
[M,N]=size(F);
K(q,w)=cond(F);
cond(F)
S=svd(F)
u=1;
r=1;
t=1;
for m=M:-1:(M/2+1)
for n=m:-1:2
F(m-1,n-1)=F(m,n);
F(m,n)=0;

t=t+1;
B=svd(F);

for i=1:1:p;
S(u,t)=B(u,r);
u=u+1;
end
u=1;

cond(F)
V(q+1,w)=l+1;
K(q+1,w)=cond(F);
q=q+1;
l=l+1;
end
end
q=1;
l=0;
w=w+1;
```

```

mesh(S);
plot(V,K,V,K,'o',V+7,flipud(K),V+7,flipud(K),'o','LineWidth',2);
xlabel({'Number of matrix'});
ylabel({'Condition number'});
title({'Зависимость числа обусловленности от вида F-матрицы'})
end

```

На рис. 4 видно, что обусловленность  $F$ -матриц лучше, нежели обусловленность исходной Фробениусовой матрицы.

На рис. 5 показан случай, когда корни характеристического полинома образуют арифметическую прогрессию:

$$[-1 \ -2 \ -3 \ -4 \ -5]. \quad (6)$$

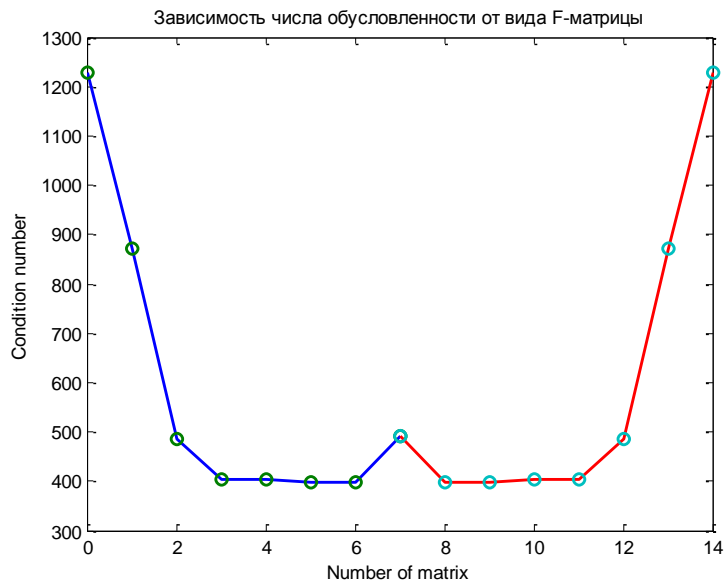


Рис. 4. График зависимости числа обусловленности от вида F-матрицы

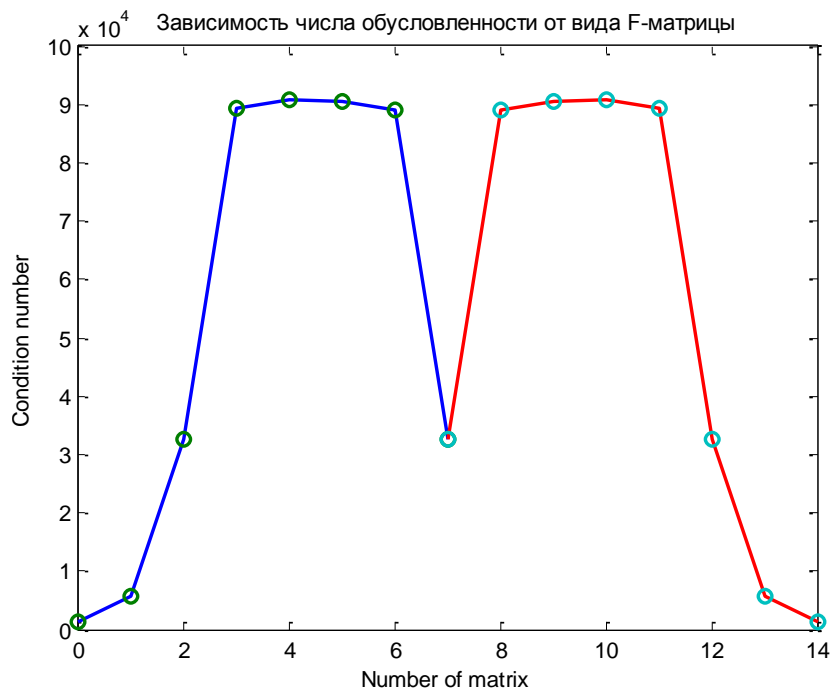


Рис. 5. График зависимости числа обусловленности от вида  $F$ -матрицы.

Код программы выглядит аналогично предыдущему случаю.

Из рисунка 5 видно, что в данном случае обусловленность полученных  $F$ -матриц хуже, нежели обусловленность исходной Фробениусовой матрицы.

*Заключение.* В статье описана новая разновидность Фробениусовых матриц. С помощью компьютерных экспериментов проанализирована обусловленность этих матриц и показано, что в некоторых случаях их обусловленность лучше, чем у классических матриц Фробениуса. Разработаны MATLAB-программы, генерирующие  $F$ -матрицы, и программы, анализирующие их обусловленность. Полученные результаты можно использовать для улучшения обусловленности компьютерных моделей.

### Библиографический список

1. Гантмахер Ф. Р. Теория матриц. М.: Физматлит, 2004. 560 с.
2. Мироновский Л. А. Моделирование линейных систем: учеб. пособие. СПб.: ГУАП, 2009. 244 с.
3. Мироновский Л. А. Аналоговые и гибридные модели динамических систем. Скалярные системы: учеб. пособие. Л.: ЛИАП. 1985. 114 с.

УДК 629.735.33

**А. И. Даванс**

студентка кафедры проблемно-ориентированных вычислительных комплексов

**А. В. Яковлев**

канд. техн. наук, доц. – научный руководитель

### ИССЛЕДОВАНИЕ ХАРАКТЕРИСТИК ПУЛЬСОМЕТРА ТИПА POLAR TEAM2 PRO

Проблема оценки функционального состояния организма (ФСО), под которым понимается совокупность характеристик его физиологических функций и психофизиологических качеств, которые несут наибольшую нагрузку в обеспечении профессиональной и поведенческой деятельности человека [1], крайне актуальна. Оценка ФСО важна для лиц, занимающихся физической культурой и спортом на различных уровнях – от посещения фитнес центров, увлечений командными играми – до профессиональных спортсменов и лиц в реабилитационных центрах.

Для всех этих людей главной задачей, помимо спортивных и профессиональных достижений, является сохранение здоровья. Здоровье определено Всемирной организацией здравоохранения (ВОЗ) как состояние полного физического, психологического и социального благополучия, для которого недостаточно лишь отсутствия заболевания и слабости [2]. Для поддержания и сохранения здоровья необходимо оптимизировать тренировки. Правильно подобранный режим тренировок позволяет улучшить физическое состояние человека, благоприятно отразиться на внешности, повысить качество его жизни. И наоборот, при неправильном подходе к физической нагрузке можно нанести большой вред организму человека. Когда рабочая нагрузка, независимо от ее природы, слишком тяжела, организм перенапрягается и происходит ухудшение здоровья.

Наиболее известным из непосредственно измеряемых физиологических показателей, позволяющим оценить ФСО человека в целом, служит частота сокращений сердца (ЧСС, HR – Heart Rate). ЧСС или ритм работы сердца зависит от возраста, пола, массы тела, тренированности. У молодых здоровых людей ЧСС составляет 60–80 уд. мин. ЧСС менее 60 уд. мин называется брадикардией, а более 90 – тахикардией. У здоровых людей может наблюдаться синусовая аритмия, при которой разница в продолжительности сердечных циклов в покое

составляет 0–0,3 с и более. Иногда аритмия связана с фазами дыхания [1]. Таким образом, ЧСС – это важнейшая характеристика сердечно-сосудистой системы организма человека, отражающая изменение сердечной активности во времени, показывающая ритм работы сердца, его функциональное состояние. Именно на основе ЧСС рассчитывается большое число вторичных показателей.

Проведение научных исследований в области физиологии, геронтологии, спортивной медицины требуют преимущественного доступа к данным первичных измерений. Существует ряд приборов для измерения ЧСС. Основными производителями таких приборов являются компании: Polar Electro Oy (Финляндия), Beurer (Германия), Cortex (Германия), SIGMA SPORT (Германия), Garmin (США), Zephyr Technology (США), Oregon Scientific (США). Одной из основных проблем применения техники указанных производителей в научных исследованиях является то, что в ряде случаев производители отказываются от всех гарантий и ответственности, связанных с использованием прибора. Отсюда возникает вопрос доверия к данным первичных измерений этих приборов.

В качестве иллюстрации рассмотрим прибор финской компании Polar Electro Oy – командную систему Polar Team 2 Pro (далее – прибор) стоимостью более 4000 евро. Прибор записывает и изучает данные тренировки в режиме реального времени, а также анализирует интенсивность тренировки и время восстановления. Он состоит из базовой станции, набора кардиодатчиков, закрепляемых на груди игрока и программного обеспечения (рис. 1). Результатом работы прибора являются дистанционно получаемые данные о ЧСС игрока (рис. 2) или группы игроков до 28 человек одновременно.

Для оценки степени достоверности данных о ЧСС, получаемых прибором, был проведен анализ первичных измерений и была предпринята попытка оценки методов их обработки. Через московское представительство компании Polar Electro Oy нами был сделан запрос на описание используемого в приборе алгоритма перевода интервалов между сокращениями сердца в ЧСС. Однако производитель предпочел не раскрывать эту информацию. Отсюда возникла задача оценки используемых в приборе методов преобразования интервалов между сердечными сокращениями в ЧСС.

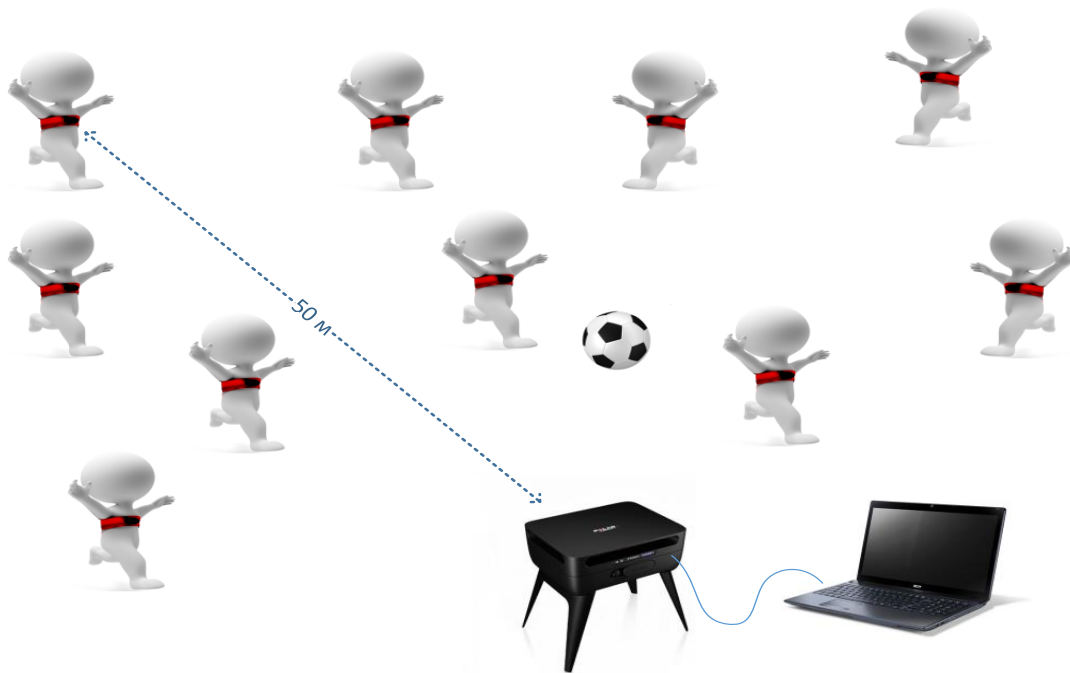


Рис. 1. Схема применения прибора

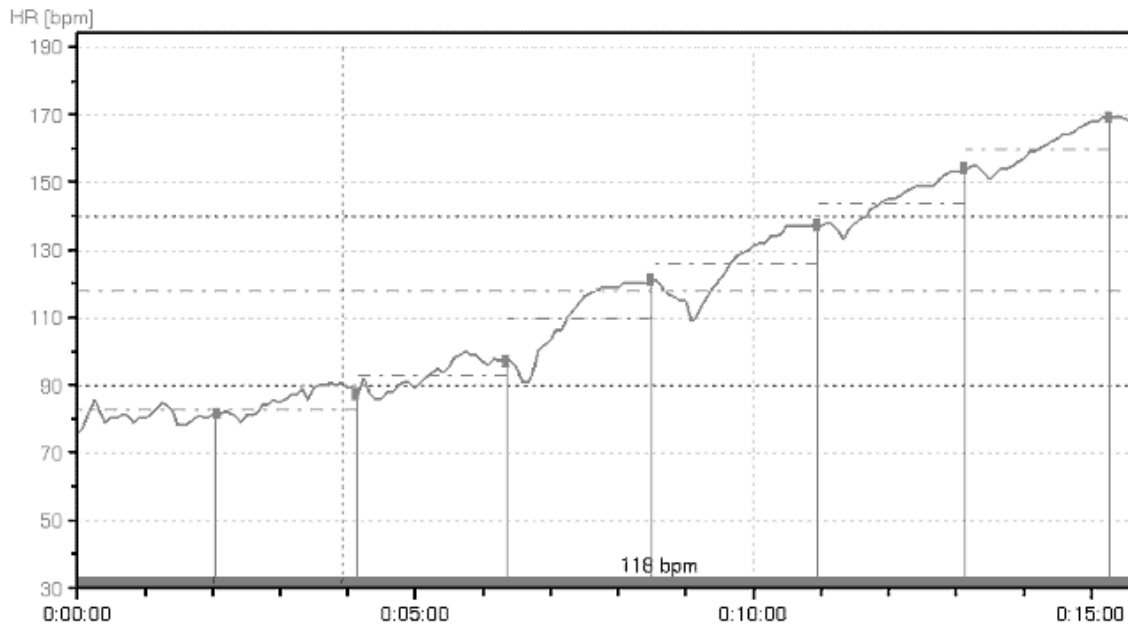


Рис. 2. График ЧСС при кардионагрузке одного игрока, полученный с помощью прибора [3]

В качестве исходных данных (рис.3) были взяты первичные данные интервалов между сердечными сокращениями и ЧСС, полученные от прибора. Для расчетов использовался пакет прикладных программ Matlab.

Первичные данные интервалов между сердечными сокращениями, полученные от прибора, экспортируем в файл (рис. 3,а). Также экспортируем ЧСС, рассчитанную на основе полученных интервалов (рис. 3,б).

1_bpm.txt — Блокнот					1_btb.txt — Блокнот				
Файл	Правка	Формат	Вид	Справка	Файл	Правка	Формат	Вид	Справка
09.12.2015 - 13:30 - Football					09.12.2015 - 13:30 - Football				
1 Player 4					1 Player 4				
Time		bpm			Beat-to-beat intervals				
00:00:00		102			608				
00:00:01		102			597				
00:00:02		101			595				
00:00:03		101			607				
00:00:04		101			615				
00:00:05		100			613				
00:00:06		100			592				
00:00:07		99			599				
00:00:08		99			609				
00:00:09		98			612				
00:00:10		98			605				
00:00:11		98			620				
00:00:12		99			629				
00:00:13		100			646				
00:00:14		100			615				
00:00:15		101			600				
00:00:16		102			610				
00:00:17		102			607				
00:00:18		101			608				
00:00:19		100			589				
00:00:20		99			590				

а)

б)

Рис. 3. Пример исходных данных

Для оценки ЧСС по интервалам сердечных сокращений был разработан алгоритм (рис.4) для сравнения оценки ЧСС с показаниями прибора. В его основе – определение соотношения интервалов между сердечными сокращениями в пределах одной секунды (1000 мс).

Рассмотрим из примера на рис. 3,6 интервал, равный 608 мс и переведем его в ЧСС (количество ударов в минуту). Один удар в данную миллисекунду произошел за 608 мс из 1000 мс, за оставшиеся 398 мс будем считать, что произошло нецелое сердечное сокращение. Из следующего интервала 597 мс вычтем 398 мс, соответственно на следующую секунду останется 205 мс сокращения. Разделим 398 мс на 597 мс и получим 0,6 сокращений, которое успело произойти в текущую секунду.

В результате получили, что в первую секунду после начала отсчета произошло 1,6 сокращения, умножим на 60 с в минуте и получим 100 уд.мин – это первое значение в последовательности ЧСС (рис.5).

Для получения адекватных значений были скорректированы полученные оценочные результаты. В частности, значения ЧСС, превышающие порог в 300 уд. мин, были приравнены к 300, а отрицательные значения ЧСС приравнены к 0. Так же для сравнения данных прибора и оценочных данных пришлось ввести сдвиг, так как ПО прибора начинает вести запись ЧСС не с первого сердечного сокращения, а после некоторой задержки. Величина сдвига определялась путем расчёта среднего квадрата ошибки.

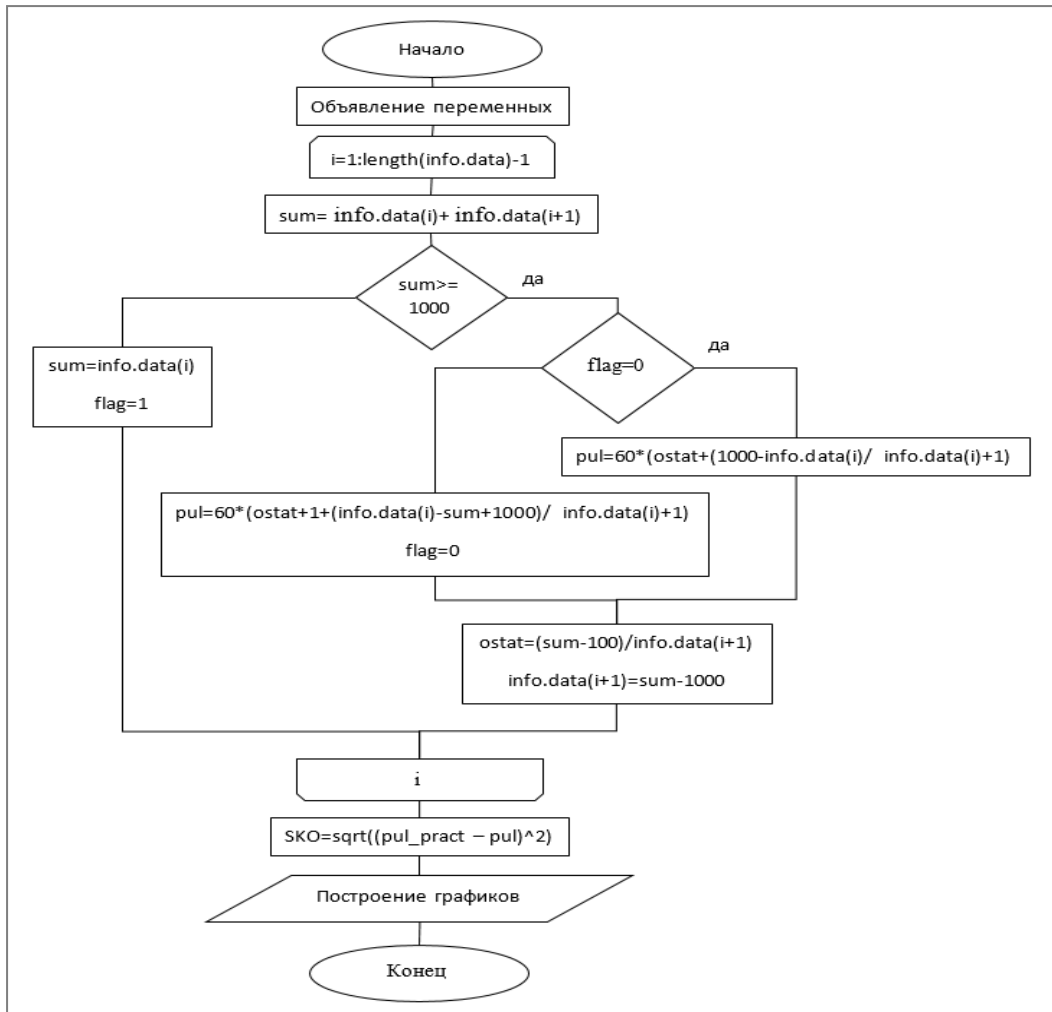


Рис. 4. Алгоритм оценки ЧСС

Светло-серой линией обозначены данные, полученные из разработанного алгоритма, жирной черной линией обозначены данные ЧСС, полученные из ПО прибора. В примере на

рисунке 5б был применен сдвиг на 48 с, в течении которых ПО прибора не фиксировало значения ЧСС;

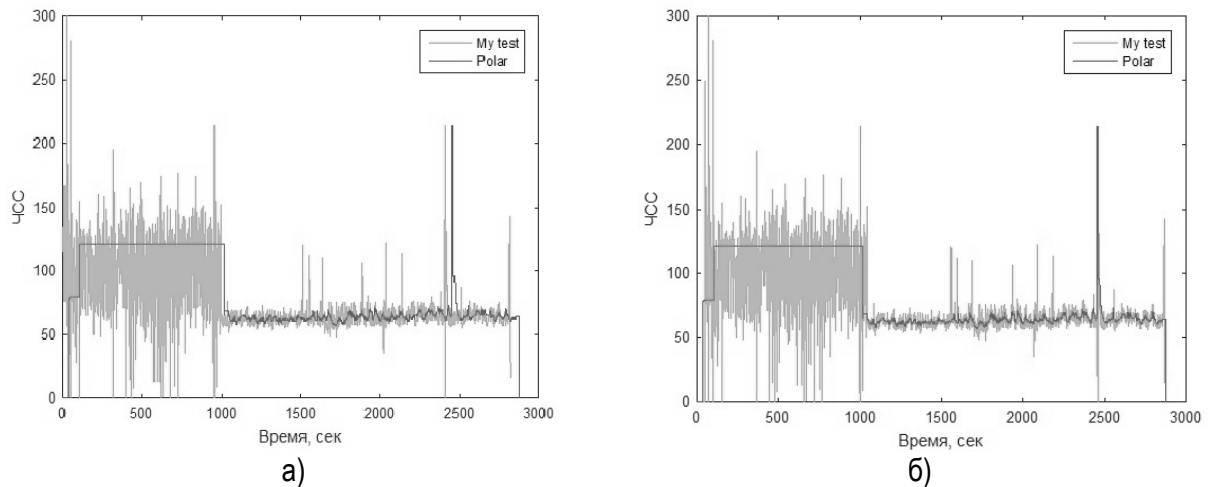


Рис. 5. График ЧСС: а – до коррекционного сдвига; б – после коррекционного сдвига

Проведенное исследование характеристик прибора Polar Team2 Pro финского производителя Polar Electro Oy позволяет сделать следующие выводы о его применимости для научных исследований:

1. Производитель прибора имеет собственный алгоритм расчета ЧСС, который он не разглашает. Все внутренние расчеты производятся по собственным алгоритмам, которые также не оглашаются. Данные, полученные при нашей оценке ЧСС, близки к исходным значениям прибора, однако ряд расхождений заставляют задуматься о достоверности расчета ЧСС прибором. Все это ставит под сомнение степень доверия к измерениям прибора при проведении серьезных научных исследований.

2. Несмотря на большие потенциальные возможности прибора в дистанционном контроле ЧСС одновременно у нескольких человек, их использование невозможно ввиду «закрытости» интерфейсов прибора. Соответственно, использование прибора в научных исследованиях с целью собственного анализа получаемых прибором данных о ЧСС затруднено.

3. Прибор слабо адаптирован для российского рынка, программное обеспечение не переведено на русский язык, руководство по эксплуатации имеет существенные недостатки и не дает возможности эффективно использовать прибор в работе.

4. Присутствуют эксплуатационные недостатки – небольшая емкость аккумуляторов, строгие гигиенические требования, слабая техническая поддержка – все это усложняет эксплуатацию прибора.

5. Прибор не входит в Госреестр средств измерений и не сертифицирован, как медицинский прибор. т.е. при оперировании таким важным жизненным показателем, каким является ЧСС, производитель не дает никаких гарантий на достоверность оцениваемых показателей. Этот недостаток приводит к вопросу о мере ответственности – ведь при неправильной трактовке первичных показателей прибором можно нанести серьезный вред организму.

Вышеперечисленные факты ставят под сомнение целесообразность использования этого, достаточно интересного прибора, при проведении исследований ЧСС у человека.

#### Библиографический список

1. Солодков А.С., Сологуб Е.Б. Физиология человека. Общая. Спортивная. Возрастная: учебник. Изд. 5-е, испр. И доп. М.: Спорт, 2015. 620 с.
2. Физиология человека. В 3-х томах. Т. 3. Пер. с англ./ Под ред. Р. Шмидта и Г. Тевса. М.: Мир, 1996. 198 с.
3. Пульсометры Polar [Электронный ресурс] <http://www.polarteam.ru/> (Дата обращения: 19.02.2016).

УДК 629.735.33

**Д. Даванс**

студент кафедры моделирования вычислительных и электронных систем

**А. В. Яковлев**

доц., канд. техн. наук – научный руководитель

## ИССЛЕДОВАНИЕ КОЛЕБАНИЙ НЕПОДВИЖНОГО ЧЕЛОВЕКА С ИСПОЛЬЗОВАНИЕМ ПАКЕТА LABVIEW

Колебания объекта могут дать достаточно важную информацию о его состоянии. Например, если взять технические объекты, то, отслеживая колебания двигателя автомобиля, можно оценить его техническое состояние [1]. В медицинской диагностике исследования распределения центра масс человека используется для оценки его функционального состояния при помощи прибора «Стабилан» [2].

Цель данной работы – создание программно-аппаратного комплекса для исследования колебаний человека на основе специализированного программного пакета NI LabVIEW (производитель – компания National Instruments, США) [3]. Для фиксации колебания были использованы профессиональные и веб-камеры. В основе нашего подхода лежало использование возможностей современной аппаратуры и программного обеспечения для решения классической задачи диагностики состояния человека.

При использовании пакета программ NI LabView для обработки сигналов от видеооборудования были выявлены проблемы совместимости с оборудованием, нехватка драйверов для поддержания работоспособности. Также были проблемы с расширенными библиотеками: более старые версии несовместимы с более новыми, а из-за многочисленных вариантов библиотек сложно выбрать и найти нужное. Также мы столкнулись с отсутствием серьезных научных публикаций по вопросам применения видеоаппаратуры в диагностических целях, выбора оборудования для подобной диагностики, разработки алгоритмов анализа подобных колебаний. Опытным путем приходилось определяться с выбором оптимального расстояния до объекта, освещенности в помещении и фона, на котором размещался исследуемый объект.

Для решения поставленных задач был разработан испытательный стенд в следующем составе (рис.1):

- современный персональный компьютер (процессор Intel Core i 3, оперативная память 4–6 Гб, жесткий диск емкостью 1 Тб, графический адаптер с интерфейсом памяти 128 Бит или более и 1 Гб выделенной памяти);

- мультипрограммный комплекс LabView с расширенными библиотеками NI LabVIEW Vision Analyzer Toolkit, NI Vision Development Module, NI Vision Acquisition;

- web-камера (минимальное разрешение 640x480, минимальная частота кадров 15 кадр/с);

- набор маркеров.

Приблизительная стоимость стенда (без стоимости программного обеспечения) – 40 тыс. руб. Трудоемкость создания представленного стенда – 300 чел.-ч.

Для увеличения точности исследования испытательный стенд можно дополнить видеокамерой технического зрения или инфракрасной камерой, специализированными маркерами и профессиональным освещением объекта.

Для проведения исследования необходимо задать в пространстве исходное положение объекта с закрепленными маркерами (рис. 2). Места расположения маркеров фиксируются в программном пакете как исходные, далее при помощи веб камеры отслеживаются колебания маркеров, затем информация выводится в графическом виде (рис. 3).



## Испытательный стенд



**ХАРАКТЕРИСТИКИ ПК**

- Процессор Intel Core i3
- Память - DDR3, 4ГБ,
- Видеокарта - 1048МБ, GDDR3
- Жесткий диск 1ТБ



**ХАРАКТЕРИСТИКИ КАМЕРЫ**

- Видеоразрешение - 640x480
- Интерфейс - USB 2.0



**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ**

- LabView 2013
- NI Vision Acquisition 2013
- NI Vision Development Module 2013
- NI LabVIEW Vizion Analyzer Toolkit 2013

**ХАРАКТЕРИСТИКИ МАРКЕРОВ**



Диаметр круглого маркера не менее 3 см



Размер квадратного маркера не менее 3 на 3 см

Рис 1. Структура испытательного стенда.

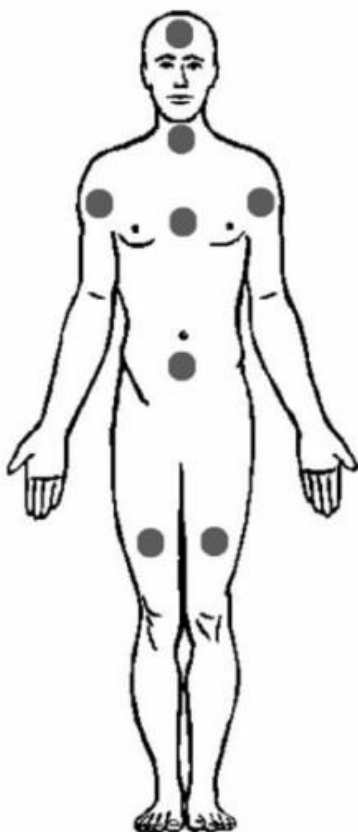


Рис. 2. Расположение маркеров

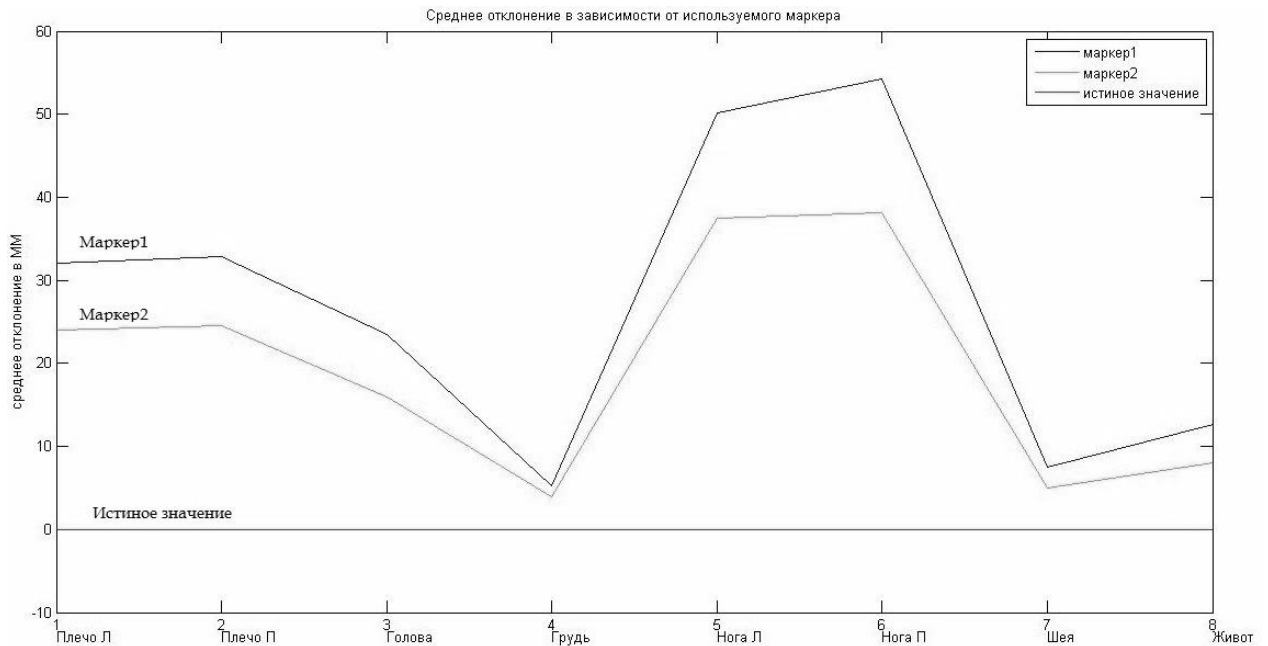


Рис. 3. Среднее отклонение в зависимости от используемого маркера

Полученные данные отображаются в графическом виде для каждого из маркеров, где мы видим его колебания, величина колебаний измерялась в миллиметрах (рис. 4)

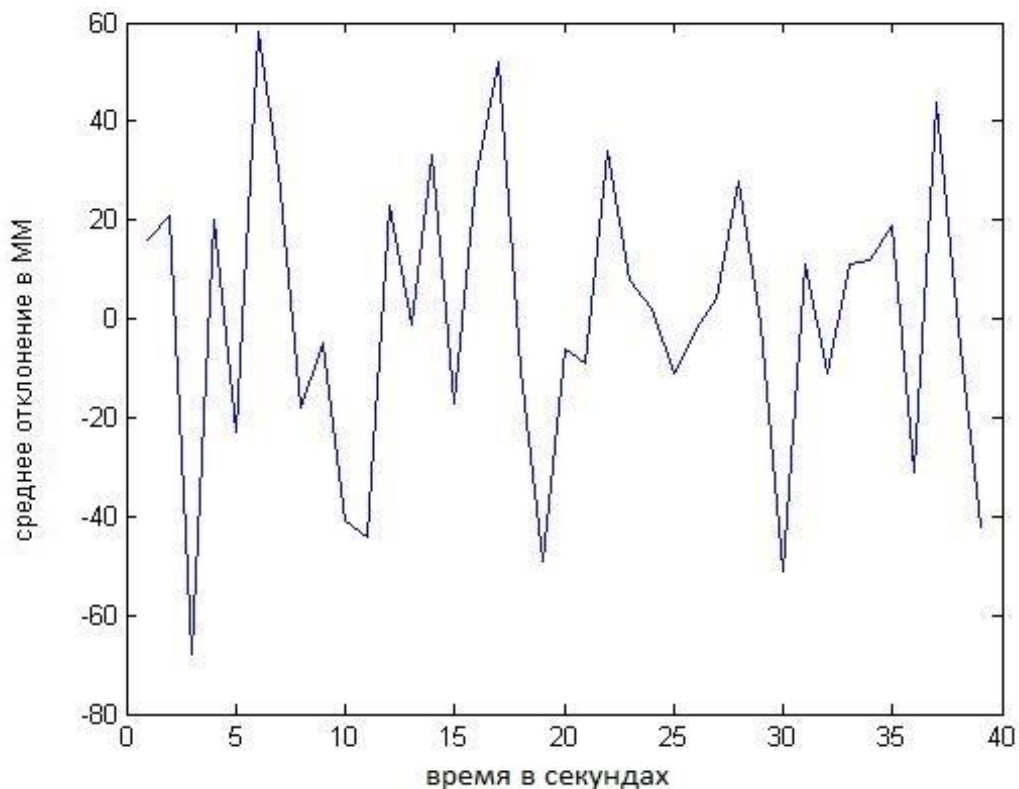


Рис. 4. Отклонение маркера на левом плече

В ходе проделанной работы были разработаны стендовое оборудование и программное обеспечение, позволяющие фиксировать колебания статичного объекта с использованием закрепленных маркеров. В результате исследования были выявлены наилучшие условия для

проведения исследования. Для более точных данных за испытуемым объектом должен находиться однотонный светлый фон, а также установлен направленный свет, лучше – профессиональный. Были выявлены оптимальные размеры и форма маркеров: окружность радиусом не менее 3 см черного цвета. Тестирование испытательного стенда на различных объектах продемонстрировало работоспособность и пригодность для дальнейшего развития. Данный проект в процессе дальнейшей доработки может быть дополнен дополнительными датчиками, такими как датчики тремора, вибрационными микрофонами для отслеживания малейших вибраций, датчики состояния окружающей среды, и использован в качестве полноценного диагностического комплекса для изучения состояния различных объектов.

#### **Библиографический список**

1. *Челомей В. Н.* Вибрации в технике: справочник в 6 т.. М: Машиностроение. 1981.
2. Стабилизатор компьютерный с биологической обратной связью [Электронный ресурс] <http://www.stabilan.ru/production/stabila/index.htm> (Дата обращения: 18.02.2016).
3. Возможности LabVIEW [Электронный ресурс] <http://www.labview.ru/labview/> (Дата обращения: 10.03.2016).
4. *Клюев В.В.* (Ред.) Неразрушающий контроль. Том 2. Книга 1, 2 М.: Машиностроение, 2003. 688 с

**УДК 629.735.33**

***О. В. Давыдова***

магистрант кафедры компьютерных технологий и программной инженерии

***Н. В. Богословская***

доц., канд. техн. наук – научный руководитель

***А. А. Ключарев***

доц., канд. техн. наук – научный руководитель

### **ОРГАНИЗАЦИЯ ЭЛЕКТРОННОГО ПОРТФОЛИО СТУДЕНТА**

Портфолио – комплект документов, подтверждающих индивидуальные достижения студента по различным направлениям деятельности, отражает не только жизнь конкретного студента, но и максимально раскрывает его творческий потенциал, успешность, перспективы и способности [1]. Портфолио – это орудие для проверки качества студента и вуза в целом. Именно поэтому портфолио можно по праву считать официальным документом, который подобно диплому о высшем образовании может повлиять на всю дальнейшую судьбу студента.

Основная цель такого документа – ознакомиться с личностными качествами и достижениями студента, который обучается на том или ином курсе университета.

Актуальностью создания портфолио так же является требование приказа Минобрнауки России от 30.10.2014 N 1406 «Об утверждении федерального государственного образовательного стандарта высшего образования», в котором говорится о формировании электронного портфолио обучающегося, в том числе сохранение работ обучающегося, рецензий и оценок на эти работы со стороны любых участников образовательного процесса [3].

Разработанная система представляет собой веб-приложение на платформе ASP.Net и языке C#. Все данные о студенте, его успеваемости в университете, работах и достижениях хранятся в базе данных MS SQL. Система хранит не только название работ и достижений, но и сами файлы. Пользователи системы, студенты и преподаватели, после осуществления авторизации (рис. 1) попадают на свою страницу (рис. 2,3).

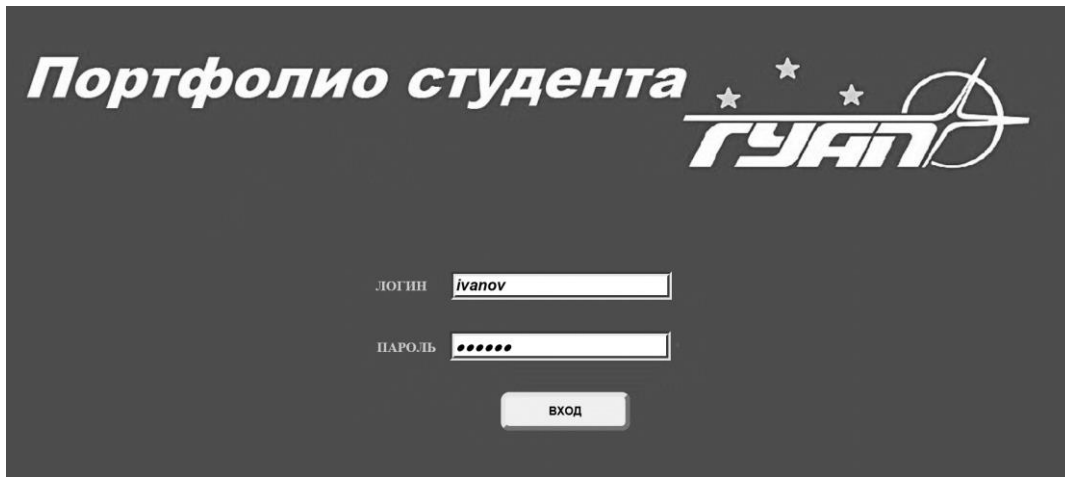


Рис. 1. Вход в систему

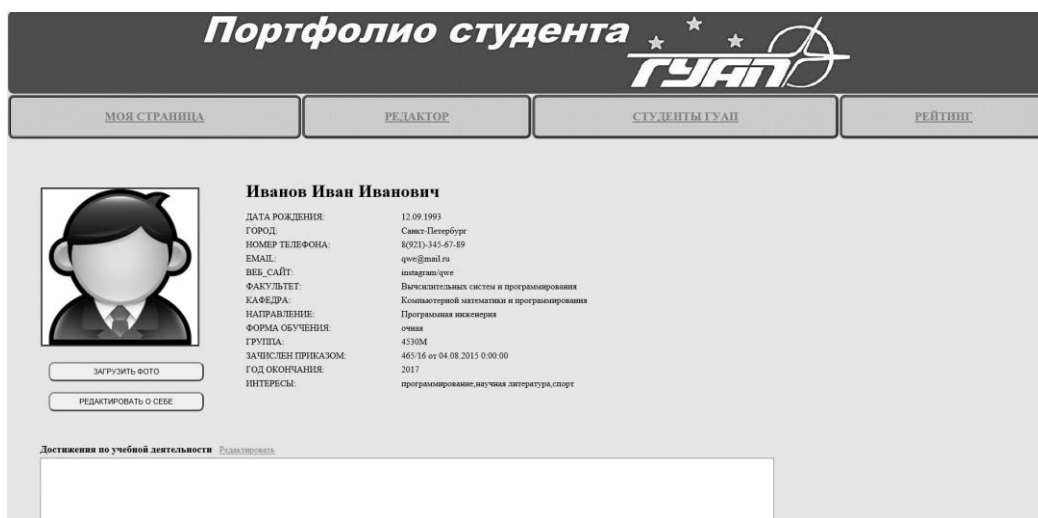


Рис. 2. Страница студента

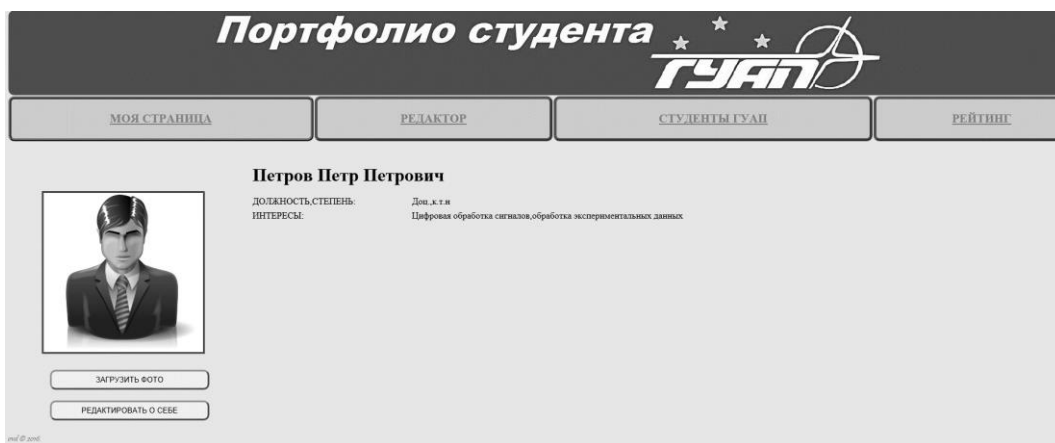


Рис. 3. Страница преподавателя

Далее пользователь имеет возможность отредактировать свои данные, добавить в портфолио свои работы и достижения с возможностью изменения видимости работы другим пользователям. Так же авторизованный пользователь может просмотреть список всех студентов, отсортировать их по разному рейтингу и увидеть свое место в этом рейтинге. Ко всему прочему каждый студент и преподаватель могут написать отзыв-рецензию к другим работам и поставить оценку. Отзывы и оценки влияют на рейтинг. На рис. 4 приведена классификация по рейтингу.



Рис. 4. Классификация рейтинга студентов

Данная система находится в стадии доработки и в дальнейшем будет тестироваться на кафедре. В заключении нужно отметить, что в настоящее время обучение в университете приобретает более автоматизированный характер. Электронное портфолио является частью этой автоматизации, и оно необходимо для развития социальных отношений с миром, для перспективного профессионального роста, и для персонального развития.

#### Библиографический список

1. <http://советстуденту.рф/obshhestvennaya-zhizn/chto-takoe-portfolio-studenta/> (дата посещения 10.10.15).
2. Положение об электронном портфолио студентов в Гуап, разработано: Центром управления качеством образования, Исполнители: Л.В. Рудакова; Санкт-Петербург, 2015
3. Приказа Минобрнауки России от 30.10.2014 N 1406 «Об утверждении федерального государственного образовательного стандарта высшего образования»

УДК 004.514

**Д. И. Дьяконов**

магистрант кафедры вычислительных систем и сетей

### ТРЕХМЕРНЫЙ ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС В СТРАТЕГИЧЕСКОЙ ИГРЕ

Современная стратегическая игра – это постоянное взаимодействие между виртуальной сценой, зачастую изображающей поле боя, режиссером – какое-либо строительство, и игроком. И правильно разработанный 3DUI (3DUserinterface – трехмерный пользовательский интерфейс) решает саму проблему взаимодействия между внутренней средой и пользователем, причем именно трехмерный пользовательский интерфейс является тем необходимым инструментом, который используют разработчики, так как все чаще баталии разворачиваются во всех трех измерениях. Кроме того, с каждым годом появляются новые и новые устройства, позволяющие по новому организовать взаимодействие и улучшить эффект погружения.

Основной целью работы является создание 3DUI в стратегической игре, что позволит пользователю взаимодействовать с виртуальным миром игры, а значит, основными задачами, которые мне необходимо будет решить являются:

– Организация навигации в виртуальном пространстве. Разработка и создание механизмов перемещения обзорной камеры в соответствии с геймплеем. Поскольку

предполагается наличие не только стратегического вида, но и активной составляющей, выражающейся в управлении летающим аватаром, необходимо реализовать несколько механизмов, как для управления стратегическим видом, так и видом от лица аватара.

– Организация селекции объектов для взаимодействия. Любая стратегия подразумевает некоторое управление происходящим, а прежде, чем начинать управлять какими-либо объектами, их необходимо выбрать, и этот механизм также надо разработать и привести в соответствие техническим заданием и внутренней логикой игры.

– Организация манипуляций с объектами.

В нашем проекте разрабатывается стратегическая игра, в которой все внимание сосредотачивается не на управлении юнитами, а на выборе зданий для строительства. Здания генерируют различных юнитов с привязкой ко времени, раз в несколько секунд. После чего юниты самостоятельно направляются к вражескому замку и атакуют его или юнитов противника, если таковые имеются.

Если брать во внимание весь спектр стратегий, то аналогов данной игры довольно много, но нужно понимать, что жанр стратегии довольно обширный и довольно сильно отличается по геймплею. Единственный полностью кросс-платформенный аналог – стратегия «Total War Battles: Kingdom», однако, это пошаговая стратегия, от которой геймплей разительно отличается.

На текущее время не существует стратегии в реальном времени, в которой игроки взаимодействовали бы, как с PC, так и с мобильных платформ, при данном геймплее. Похожий режим был реализован в игре Warcraft3, карта «Битва Замков», там было 2 команды, у них – по одному главному замку, необходимо было строить здания, в которых генерировались бы юниты за определенный промежуток времени. Ими нельзя было управлять, они самостоятельно двигались в направлении противника, атакуя все, что встретится на пути. Целью было уничтожение вражеского замка. Эта идея и взята за основу игры.

Из представленной ниже таблицы видно, что в последнее время все реже выпускаются игры не имеющие более гибкого трехмерного интерфейса, позволяющего изменять угол и высоту обзора с камеры, способствующего лучшей ориентации на местности.

Организация трехмерного пользовательского интерфейса в современных стратегиях.

Игровой продукт	Дата выхода	3d UI	Изменение угла обзора	изменение высоты обзора
StarCraft	31 марта 1998 г.	-	-	-
Homeworld	28 сентября 1999 г.	+	+	+
BlitzkriegII	23 сентября 2005 г.	+	+	+
MEDIEVAL II TOTAL WAR	7 декабря 2006 г.	+	-	+
UFO Afterlight	9 февраля 2007 г.	+	+	+
Command & Conquer - Red Alert 3	5 ноября 2008 г.	+	-	+
Warhammer 40k Down of War 2	31 марта 2009 г.	+	+	+
Supreme Commander 2	5 марта 2010 г.	+	+	+
StarCraft II	26 июля 2010 г.	+	+	+
WarCraft 3	4 июля 2013 г.	+	-	+
Europa Universalis IV	13 августа 2013 г.	+	-	+
XCOM Enemy Within	15 ноября 2013 г.	+	+	+
Jagged Alliance Back in Action	15 мая 2014 г.	+	+	+
Ancient Space	23 сентября 2014 г.	+	+	+
Civilization - Beyond Earth	24 октября 2014 г.	+	+	+
Hard West	18 ноября 2015 г.	+	+	+

На текущий момент была проведена апробация возможностей движка:

- Были выбраны оптимальные настройки камеры позволяющие не сильно загружать процессор графическими объектами находящимися на большом расстоянии и качественно отображать близкие предметы, являющиеся частью пользовательского интерфейса.

- Была отработана схема получения изображения с камеры для дальнейшего его использования в качестве текстуры для других объектов в нужном качестве, которая будет использована в дальнейшем, при решении задачи навигации.

- Был разработан скрипт для перемещения камеры с заданным положением по высоте и углом наклона параллельно плоскости карты, что является первым шагом на пути решения задачи навигации в приложении и манипуляции.

В заключении скажем, что по окончании разработки, в приложении будет реализовано управление перемещением камеры за счет наклона устройства (для мобильных устройств) и нажатия на необходимое место на карте; будет реализовать осмотр строения и способ непосредственного управления юнитом-наблюдателем.

### Библиографический список

1. <http://docs.unity3d.com/ru/current/Manual/UnityOverview.html> от 24.05.2016.
2. *Doug A. Bowman* 3D User Interfaces Theory and Practice /Westford, Massachusetts, 2011. 457с
3. *Ryan Henson Creighton* Unity 3D Game Development by Example Beginner's Guide / Olton Birmingham, 2010. 384с
4. [ru.wikipedia.org](http://ru.wikipedia.org) от 24.05.2016
5. *Патрик Тэрбин*. Стратегические игры М.: Баланс-Клуб 2003. 264 с.

УДК 004.023

**А. Ю. Еремченко**

студент кафедры компьютерных технологий и программной инженерии

**Л. Ю. Леонтьев**

ст. преп. – научный руководитель

## СОЗДАНИЕ АНАЛИТИЧЕСКОЙ МОДЕЛИ ДЛЯ РЕШЕНИЯ ЗАДАЧ АНАЛИЗА ДАННЫХ

При создании аналитической модели может возникнуть проблема подбора ее характеристик для получения наилучшего результата. Для того чтобы решить эту проблему, модель необходимо представить в виде черного ящика, работа которого зависит от набора входных параметров. При реализации такой структуры процесс подбора параметров можно автоматизировать. Так же такая реализация увеличит масштабируемость системы, потому что теперь ее можно разбить на независимые друг от друга модули.

Если работа системы происходит в предметной области, где появление новых данных и их анализ должны происходить в режиме реального времени, то использование такой схемы позволит спроектировать конечную систему таким образом, чтобы процесс подбора параметров был интегрирован в общую рабочую структуру системы и происходил непрерывно.

Задача создания правильной комбинации набора параметров – это классическая задача оптимизации (найти такой набор параметров, при котором разница между прогнозируемым значением и реальным значением должна быть минимальна). Дать гарантированный ответ на этот вопрос можно только при помощи полного перебора всех вариантов, что в большинстве случаев не может быть реализовано за разумное время.

Тогда такую задачу можно попытаться решить при помощи эвристического алгоритма, которым, например, может являться генетический алгоритм [1]. В терминологии генетических алгоритмов функционал, который необходимо оптимизировать, называют фитнес-функцией [2,

3]. Однако, важно помнить тот факт, что в силу своей идеи у генетических алгоритмов есть важный недостаток – преждевременная сходимость, т.е. остановка алгоритма до достижения глобального максимума.

Существует большое количество аналитических моделей: древовидные, модели на основе правил, линейные, метрические, вероятностные и другие. Использование генетических алгоритмов для подбора параметров лучше всего подходит для линейных моделей. Такие модели являются параметрическими – они имеют фиксированную форму с конечным количеством числовых параметров, которые должны быть найдены в ходе обучения по данным [4].

Примером такой линейной модели может служить однофакторный регрессионный анализ. Параметрами для модели, построенной на основе однофакторного регрессионного анализа, могут быть (подразумевается, что уравнения удовлетворяют критериям Фишера и Стьюдента):

- объем выборки исторических данных;
- показатель степени полинома для уравнения регрессии;
- длина промежутка времени, на который будет строиться прогноз;
- нужно ли усреднять значения, полученные при помощи нескольких уравнений регрессии разных степеней.

Программная реализация данной схемы может быть выполнена, как и в рамках одной программной платформы, так и с использованием различных инструментов, предоставленных различными программными платформами. В рамках данной работы была реализована следующая схема:

1. Происходит агрегация данных при помощи приложения на языке C#;
2. При помощи генетического алгоритма генерируется набор параметров;
3. На основе полученных данных генерируется исходный текст скрипта на языке MatLab (реализация самого регрессионного анализа) с заданными параметрами;
4. Все скрипты отправляются на выполнение, результаты их работы сохраняются в текстовый файл;
5. Полученные результаты анализируются (опять работает приложение на C#) и на основе результатов определяются сильнейшие особи, на основе которых будет создано следующее поколение;
6. Возврат ко второму шагу (до тех пор, пока не будет найдена особь, удовлетворяющая требованиям).

В качестве примера можно привести результаты работы двух моделей: изначальной и выведенной алгоритмом (после 5 итераций).

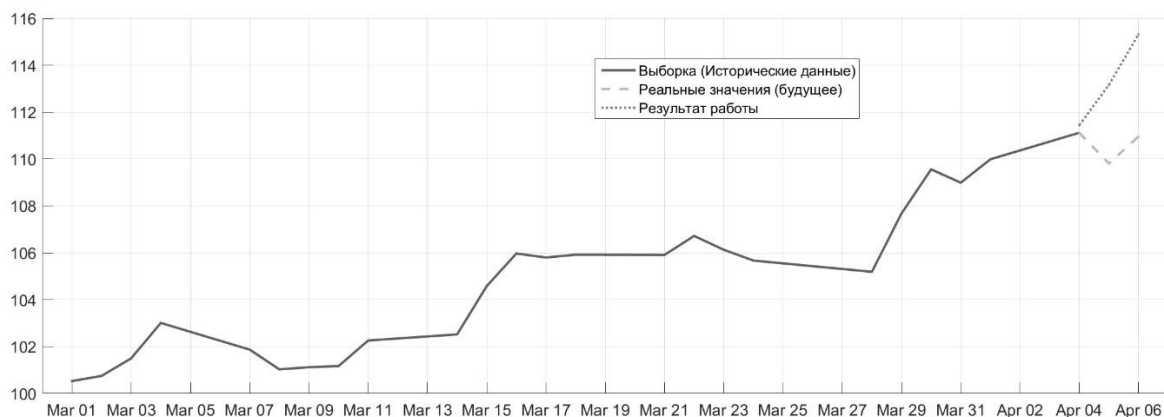


Рис. 1. Результат работы изначальной модели



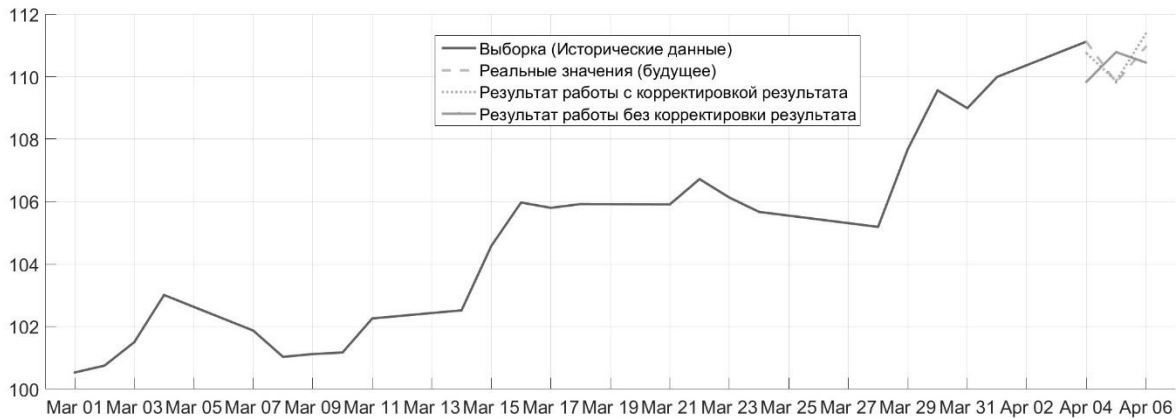


Рис. 2. Результат работы выведенной модели

Изначальная модель (рис. 1.): объем исходной выборки – 30 дней; прогноз осуществляется на 7 дней; уравнение регрессии является полиномом 5-й степени; усреднение не происходит;

Выведенная модель (рис. 2.): объем исходной выборки – 30 дней; прогноз осуществляется на 3 дня; происходит усреднение значений, полученных при помощи уравнений регрессии 2, 3 и 4-й степеней;

В качестве исходных данных были выбраны стоимость акций компании Apple в период с 01.03.2016 по 06.04.2016 без учета выходных дней.

В данном примере используется классический генетический алгоритм, где в качестве метода селекции используется – «Метод ранжирования» (рис. 3). Это означает, что вероятность выбора особи зависит от ее позиции в списке особей, отсортированном по значению фитнес-функции [5].

$$p_i = \frac{1}{N} \left( k - (k - b) \frac{i - 1}{N - 1} \right), \text{ где } k \in [1, 2]; b = 2 - k;$$

$p_i$  – вероятность выбора  $i$  особи;  $N$  – количество особей в популяции;  $i$  – порядковый номер особи в списке особей, отсортированном по значению функции приспособленности, т.е.

$$\forall i \forall j > i$$

$$f_i \leq f_j \text{ – при минимизации}$$

	1	2	...	j	...	m
1	$a_{11}$	$a_{12}$	...	$a_{1j}$	...	$a_{1m}$
2	$a_{21}$	$a_{22}$	...	$a_{2j}$	...	$a_{2m}$
...	...	...	...	...	...	...
i	$a_{i1}$	$a_{i2}$	...	$a_{ij}$	...	$a_{im}$
...	...	...	...	...	...	...
n	$a_{n1}$	$a_{n2}$	...	$a_{nj}$	...	$a_{nm}$

Рис. 3. Матрица, которая строится при методе ранжирования

В матрице, изображенной на рис. 3  $a_{ij}$  – это значение фитнес-функции,  $n$  – количество параметров,  $m$  – количество итераций.

В качестве вывода можно отметить, что и при работе с простыми моделями, такими как линейные, можно достичь приемлемо высокой точности, если правильно подобрать ее параметры. Для правильного подбора параметров можно использовать эвристические

алгоритмы. Использование даже простых модификаций таких алгоритмов, как, например, генетический алгоритм, позволяет найти оптимальные параметры за разумно минимальное время.

### Библиографический список

1. *Mitchell, Melanie* (1996). *An Introduction to Genetic Algorithms*. Cambridge, MA: MIT Press. ISBN 978-0-5850-3094-4.
2. *Рутковская Д., Пилиньский М., Рутковский Л.* Нейронные сети, генетические алгоритмы и нечеткие системы. М.: «Горячая линия – Телеком», 2006. 452 с. ISBN 5-93517-103-1.
3. *Michael O'Neill*, Student Member, IEEE, and *Conor Ryan*. Grammatical Evolution. IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 5, NO. 4, AUGUST 2001.
4. *Петер Флах.* Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных = *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*. М.: «ДМК Пресс», 2015. 400 с. ISBN 978-5-97060-273-7.
5. *David E. Goldberg.* *Genetic Algorithms in Search, Optimization, and Machine Learning* «Addison-Wesley Professional», Retrieved 2 July 2013. 432 с. ISBN 978-0-2011-5767-3.

УДК 004.021

**С. С. Золотухин**

студент кафедры компьютерных технологий и программной инженерии

**В. В. Бураков**

д-р. техн. наук, доц. – научный руководитель

## ВЫБОР АЛГОРИТМА ПОВЕРХНОСТНОГО КОДИРОВАНИЯ И КОРРЕКЦИИ ОШИБОК ПРИ ПРОЕКТИРОВАНИИ ПРОГРАММНО-ОПРЕДЕЛЯЕМОЙ СИСТЕМЫ ХРАНЕНИЯ ДАННЫХ

Бурный рост онлайн-сервисов, хранящих все возрастающие объемы неструктурированных данных, позволяет предположить, что в ближайшие годы станут доминировать объектные системы хранения данных, чему способствует и их экономическая эффективность, неограниченность масштабирования, высокая отказоустойчивость и производительность.

Стремясь помочь заказчикам успешно пройти период, который некоторые специалисты считают крупнейшим тектоническим сдвигом в истории ИТ-отрасли, компании выпускают новые программно-определяемые системы хранения, которые помогают организациям управлять традиционными корпоративными приложениями, быстрее и эффективнее развертывать приложения следующего поколения и преобразовывать новые источники аналитических данных в новые источники прибыли [1].

Прежде всего, необходимо максимально понизить стоимость хранения (\$/Гбайт), что особенно актуально на больших кластерах, но в то же время данные должны быть постоянно доступны в любой момент времени. Здесь на первый план выходит выбор алгоритма поверхностного кодирования и коррекции ошибок. Ключом к коррекции ошибок является избыточность. Действительно, простейший код коррекции ошибок – это простое повторение всего несколько раз. Если, к примеру, вы ожидаете возникновения не более одной ошибки при пересылке, то повторение каждого бита три раза и «голосование большинством» на принимающем конце гарантирует, что сообщение будет услышано правильно (например, «111 000 011 111» будет правильно услышано как «1011»). В общем случае  $n$  ошибок может быть скомпенсировано путем повторения чего-то  $2n + 1$  раз [5].

Но такая коррекция ошибок «в лоб» препятствовала бы цели высокоскоростной, высокоплотной обработки информации. Можно было бы предпочесть подход, при котором добавляется только несколько дополнительных бит к исходному сообщению. Конечно, вы не можете всегда получать то, что хотите, – напоминает Мик Джаггер, – но если вы попробуете, то иногда, в самом деле, можете найти то, что вам нужно. Успех кодов Рида–Соломона подтверждает это.

Процесс, который основывается на кодах Рида–Соломона получил название Erasure Code (далее EC). Этот процесс применяется только к полностью заполненным логическим контейнерам (чанкам). Он осуществляет кодирование данных для защиты от потерь и снижения избыточности. Не кодируется только индекс, поскольку система постоянно обращается к нему. Поэтому защита индекса осуществляется с помощью трехкратной репликации в разные чанки и эширования в памяти [5].

Чанк разбивается на 16 модулей – 12 модулей данных и 4 модуля кода, которые записываются на разные узлы (рис. 1). При этом чем больше количество узлов, тем выше отказоустойчивость. Обратите внимание, что блоки записываются в одном экземпляре.



Рис.1 Разбиение чанков на блоке. Erasure Coding

Процедура декодирования используется только для восстановления после сбоев, для операций чтения этого не требуется. Более того, для чтения небольших объектов системе даже не нужно обращаться ко всем содержащим его чанкам. Все это положительно сказывается на общей производительности системы.

Суть EC заключается в том, что после кодирования некоторого файла получается  $n$  фрагментов (в практической реализации это так же файлы). Любые  $m$  из этих фрагментов равны по размеру оригинальному файлу. При этом  $n > m$ . Каждый из фрагментов сохраняется в отдельном облачном хранилище. Для восстановления первоначального файла достаточно собрать  $m$  любых фрагментов и декодировать. Остальные  $n - m$  фрагменты могут быть удалены, испорчены, содержащие их облачные хранилища не доступны и так далее. Таким образом, система, использующая erasure codes, может справиться с появлением  $n - m$  ошибок.

Соотношение параметров  $n$  и  $m$  можно выбирать. Например, можно задать параметры так, что  $n = 2m$ , в таком случае размер закодированных данных будет ровно в два раза больше оригинального. К примеру, пускай  $m = 5$ ,  $n = 10$ . Тогда фрагментов после кодирования получается 10 штук. Размер одного фрагмента будет равен отношению размера файла к  $m$ . Если собрать любые 5 из этих фрагментов и посчитать общий размер, то он будет равен размеру первоначального файла. Каждый из фрагментов сохраняется на одном из 10 облачных хранилищ. Любые 5 хранилищ при этом могут в один момент времени оказаться недоступны, на доступность же пользовательских данных это не повлияет – файл можно будет восстановить.

В математических обозначениях это выглядит так. Допустим, размер файла равен  $L$ . Тогда размер одного фрагмента составляет  $P = \frac{L}{m}$ , общий размер избыточных данных –  $2L$ .

Размер всех фрагментов, необходимых для восстановления равен  $L = pm$ .

Сравним с репликацией, чтобы иметь в систему, в которой любые 5 облачных хранилищ могут выйти из строя, необходимо иметь 6 облачных хранилищ, каждое из которых хранит копию файла. В таком случае размер данных, сохраненных в системе, будет в 6 раз больше размера файла. В то время, как при использовании ЕС, понадобится сохранить избыточные данные в два раза превышающие размер первоначального файла.

Таким образом, в системе, которая способна пережить появление определенного числа ошибок, при использовании Erasure Coding:

- на каждом из серверов нужно хранить меньшее количество данных;
- общий размер данных, которые нужно сохранить, будет существенно меньше;
- общее число облачных хранилищ требуется больше.

Если объем данных, которые необходимо сохранить, меньше, то и количество трафика, потраченного на сохранение, будет пропорционально меньше. Это неоспоримое преимущество использования ЕС в гипермасштабируемых системах. При восстановлении размер данных, которые требуется передать будет равен размеру изначального файла, т. е.  $L$ .

Кодирование с помощью кода Рида–Соломона может быть реализовано двумя способами: систематическим и несистематическим.

При несистематическом кодировании информационное слово умножается на некий неприводимый полином в поле Галуа. Полученное закодированное слово полностью отличается от исходного и для извлечения информационного слова нужно выполнить операцию декодирования и уже потом можно проверить данные на содержание ошибок. Такое кодирование требует большие затраты ресурсов только на извлечение информационных данных, при этом они могут быть без ошибок.

При систематическом кодировании к информационному блоку из  $k$  символов приписываются  $2t$  проверочных символов, при вычислении каждого проверочного символа используются все  $k$  символов исходного блока. В этом случае нет затрат ресурсов при извлечении исходного блока, если информационное слово не содержит ошибок, но кодировщик/декодировщик должен выполнить  $k(n-k)$  операций сложения и умножения для генерации проверочных символов. Кроме того, так как все операции проводятся в поле Галуа, то сами операции кодирования/декодирования требуют много ресурсов и времени. Быстрый алгоритм декодирования, основанный на быстром преобразовании Фурье, выполняется за время порядка  $O(\ln(n))^2$ .

При операции кодирования информационный полином умножается на порождающий многочлен. Умножение исходного слова  $S$  длины  $k$  на неприводимый полином при систематическом кодировании можно выполнить следующим образом [4]:

К исходному слову приписываются  $2t$  нулей, получается полином  $T = Sx^{2t}$ .

Этот полином делится на порождающий полином  $C$ , находится остаток  $R$ ,  $Sx^{2t} = QG + R$   $Sx^{2t} = QC + R$ , где  $Q$  – частное.

Этот остаток и будет корректирующим кодом Рида–Соломона, он приписывается к исходному блоку символов. Полученное кодовое слово  $C = Sx^{2t} + R$ .

Кодировщик строится из сдвиговых регистров, сумматоров и умножителей. Сдвиговый регистр состоит из ячеек памяти, в каждой из которых находится один элемент поля Галуа.

Существует и другая процедура кодирования (более практичная и простая). Положим  $a_i \in GF(q)$ ,  $(i = 1, 2, \dots, k-1)$ ,  $a \in GF(q)$  – примитивный элемент поля  $GF(q)$ .

пусть  $a = (a_0, a_1, \dots, a_{k-1})$   $\alpha = (\alpha_0, \alpha_1, \dots, \alpha_{k-1})$  – вектор информационных символов, а значит  $a(x) = a_0 + a_1x + \dots + a_{k-1}x^{k-1}$  – информационный многочлен. Тогда вектор  $u = (a(1), a(\alpha), \dots, a(\alpha^{q-2}))$   $u = (a(1), a(\alpha), \dots, a(\alpha^{q-2}))$  есть вектор кода Рида – Соломона, соответствующий информационному вектору  $a$ . Этот способ кодирования показывает, что для кода РС вообще не нужно знать порождающего многочлена и порождающей матрицы кода, достаточно знать разложение поля  $CF(q)$  по примитивному элементу  $\alpha$  и размерность кода (длина кода в этом случае определяется как  $n = q - 1$ ). Все дело в том, что за разностью  $n - k$  полностью скрывается порождающий многочлен  $g(x)$  и кодовое расстояние.

Декодировщик, работающий по авторегрессивному спектральному методу декодирования, последовательно выполняет следующие действия:

- Вычисляет синдром ошибки.
- Строит полином ошибки.
- Находит корни данного полинома.
- Определяет характер ошибки.
- Исправляет ошибки.

Вычисление синдрома ошибки выполняется синдромным декодером, который делит кодовое слово на порождающий многочлен. Если при делении возникает остаток, то в слове есть ошибка. Остаток от деления является синдромом ошибки.

Вычисленный синдром ошибки не указывает на положение ошибок. Степень полинома синдрома равна  $2t$ , что много меньше степени кодового слова  $n$ . Для получения соответствия между ошибкой и ее положением в сообщении строится полином ошибок. Полином ошибок реализуется с помощью алгоритма Берлекэмп – Мессис, либо с помощью алгоритма Евклида. Алгоритм Евклида имеет простую реализацию, но требует больших затрат ресурсов. Поэтому чаще применяется более сложный, но менее затратоёмкий алгоритм Берлекэмп – Мессис. Коэффициенты найденного полинома непосредственно соответствуют коэффициентам ошибочных символов в кодовом слове.

На этом этапе ищутся корни полинома ошибки, определяющие положение искаженных символов в кодовом слове. Реализуется с помощью процедуры Ченя, равносильной полному перебору. В полином ошибок последовательно подставляются все возможные значения, когда полином обращается в ноль – корни найдены.

По синдрому ошибки и найденным корням полинома с помощью алгоритма Форни определяется характер ошибки и строится маска искаженных символов. Однако для кодов РС существует более простой способ отыскания характера ошибок, используя формулу (1). Для кодов РС с произвольным множеством  $2t_d$  последовательных нулей  $a^b, a^{b+1}, \dots, a^{b+\delta}, \delta = 2t_d - 1$ .

$$e_{ij} = \frac{(a^{ji})^{2+b} \wedge (a^{ji})}{\sigma'(a^{ji})}, \tag{1}$$

где  $\sigma'(x)$  формальная производная по  $x$  многочлена локаторов ошибок  $\sigma(x)$ , а любой  $x = \sigma(x)S(x) \bmod x^{2t}_{d+1}$  [3].

Далее, после того, как маска найдена, она накладывается на кодовое слово с помощью операции XOR и искаженные символы восстанавливаются. После этого отбрасываются проверочные символы и получается восстановленное информационное слово.

Практическую ценность кодов Рида–Соломона трудно переоценить. С помощью них можно не только обнаруживать, но и частично восстанавливать информацию практически «из

пепла». К сожалению, коды Рида–Соломона у большинства специалистов ассоциируются только с помехоустойчивым кодированием в каналах передачи данных [2]. В действительности, их можно применять везде, где необходимо предотвратить модификацию данных:

- Обнаружение и коррекция неумышленного искажения при передаче данных по каналам связи, а также искажения данных на носителях информации при их повреждении.
- Обнаружение и коррекция умышленной модификации информационных сообщений с целью дезинформации (особенно в театре военных действий).
- Обнаружение и коррекция умышленной модификации информации об авторе или исполняемого кода с целью «взлома» программного обеспечения.
- Защита программного обеспечения или данных от копирования с лицензионного диска при помощи использования специальных «настоящих» и «ложных» ошибок в секторах.
- Восстановление данных одного или нескольких дисков в отказоустойчивых системах (массивах) хранения данных (RAID, FTDS). Восстановление одного или нескольких томов многотомного архива данных.
- Обнаружение и исправление ошибок в последовательности команд в конвейере (очереди) процессора, вычислительного узла или иной специализированной системе.
- Обнаружение и исправление ошибок в цепочках ДНК в геномной инженерии.

### Библиографический список

1. Павлов С., Системы высокопроизводительных вычислений в 2012-2013 годах: обзор достижений и анализ рынков. Часть I. Серверы, компьютеры, планшеты, смартфоны // CAD/CAM/CAE Observer, 2013. №5, с. 69-79; №6, с. 90-92.
2. Рахман П. А., Кодирование информации с применением кодов Рида-Соломона [электронный ресурс] URL: <https://bugtraq.ru/library/crypto/.keep/rscodes.pdf>.
3. Jerasure Org // Jerasure документация, [электронный ресурс] URL: <http://jerasure.org/jerasure-2.0/>.
4. Jerasure Org // Jerasure GF документация, [электронный ресурс] URL: <http://jerasure.org/gf-complete-1.02/>.
5. Хабрахабр // Коллективный блог, [электронный ресурс] URL: <https://habrahabr.ru/>.

УДК 004.054

**М.С. Иванова**

студентка кафедры проблемно-ориентированных вычислительных комплексов

**П.А. Липецкий**

студент кафедры проблемно-ориентированных вычислительных комплексов

**Г.С. Евсеев**

доц., канд. техн. наук – научный руководитель

## АНАЛИЗ МЕТОДОВ РАСПРЕДЕЛЕНИЯ РЕСУРСА КАНАЛА СВЯЗИ

### Введение

В последнее время проблема стоимости трафика отходит на второй план, а вот проблемы со скоростью передачи данных через Интернет остаются. Каким бы широким не был канал, один абонент, скачивая большой файл, может занять его весь, т. е. если одни абоненты скачивают большие объемы информации, то они могут существенно ограничить скорость других абонентов. Для решения этой проблемы необходимо распределять Интернет – канал между абонентами в соответствии с их потребностями.

В данной работе рассматриваются варианты распределения канального ресурса (КР) между абонентами.

Взаимодействие базовой станции (БС) с абонентами можно представить следующим образом (рис.1). В системе имеется  $N$  абонентов. Абонент  $i$  посылает запрос  $R_i$  (бит/с) на БС по своему каналу, а пропускная способность этого абонента равняется  $C_i$  (бит/с) при условии, что весь КР отдается этому абоненту.

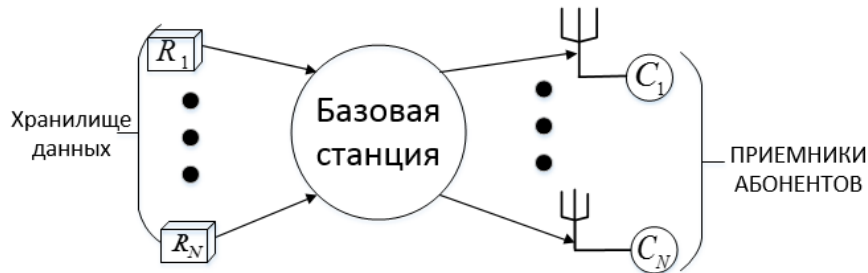


Рис. 1. Моделируемая система

Таким образом, требуемая  $i$ -ым абонентом доля КР высчитывается по формуле :

$$\alpha_i = \frac{R_i}{C_i} \quad (1)$$

На практике обычно выполняется допущение, что  $R_i \leq C_i$ , то есть  $\alpha_i \leq 1$ .

Также у системы есть такая характеристика, как  $\gamma$  – общая загруженность канала, которая равняется сумме запрашиваемых долей КР. Очевидно, что при выполнении условия  $\gamma \leq 1$  требования всех абонентов будут удовлетворены. Поэтому задача оптимизации актуальна тогда, когда требуемая нагрузка канала  $\gamma > 1$ .

Пусть  $\beta_1 \dots \beta_N$  – выделяемые доли КР, т. е.  $\sum \beta_i \leq 1$ . Если  $\gamma > 1$ , то  $\sum \beta_i = 1$ , если же  $\gamma < 1$ , то  $\sum \alpha_i = \sum \beta_i < 1$ .

Исходя из вышесказанного, можно сформулировать следующую оптимизационную задачу:

$$F(\beta_1, \beta_2, \dots, \beta_N) \rightarrow \max, \quad (2)$$

при следующих ограничениях:

$$\beta_i \leq \alpha_i, \quad i = \overline{1, N}; \quad (3)$$

$$\beta_1 + \dots + \beta_N \leq 1. \quad (4)$$

На практике обычно рассматривают три целевые функции (5)-(7) для решения задачи оптимизации (2):

$$F_1(\beta_1, \beta_2, \dots, \beta_N) = \beta_1 C_1 + \dots + \beta_N C_N; \quad (5)$$

$$F_2(\beta_1, \beta_2, \dots, \beta_N) = \sum_{i=1}^N \text{Ind}(\beta_i = \alpha_i), \quad (6)$$

где  $\text{Ind}(\text{условие}) = \begin{cases} 1, \text{условие выполняется} \\ 0, \text{условие не выполняется} \end{cases}$

Кроме того, используется критерий минимизации выделяемой максимальной доли абоненту, то есть

$$F_3(\beta_1, \beta_2, \dots, \beta_N) = \max(\beta_1, \beta_2, \dots, \beta_N) \rightarrow \min \quad (7)$$

Для этой задачи известен метод решения, так называемый Proportional Fair.

Функция (5) соответствует максимизации количества переданной информации, а функция (6) соответствует максимизации числа абонентов, чьи требования полностью удовлетворены.

Алгоритмы распределения КР

Для задачи максимизации суммарной скорости решение сводится к выполнению следующих действий:

1. Массив с пропускной способностью каналов абонентов  $[C_1, \dots, C_N]$  сортируется по убыванию, а затем происходит поочередное выделение запрашиваемых долей КР ( $\beta_i = \alpha_i$ ) начиная с абонента, у которого максимальная пропускная способность.

2. Алгоритм останавливается, когда  $\sum \beta_i = 1$ .

Для задачи максимизации числа абонентов, чьи требования были полностью удовлетворены, решение сводится к выполнению следующих действий:

1. Массив с запрашиваемыми долями КР  $[\alpha_1, \dots, \alpha_N]$  сортируется по возрастанию, а затем происходит поочередное выделение запрашиваемых долей КР ( $\beta_i = \alpha_i$ ) начиная с абонента, у которого доля запрашиваемого КР минимальна.

2. Алгоритм останавливается, когда  $\sum \beta_i = 1$ .

Для задачи минимизации выделяемой максимальной доли абоненту решение сводится к выполнению следующих действий:

1. Массив с запрашиваемыми долями КР  $[\alpha_1, \dots, \alpha_N]$  сортируется по возрастанию, после чего рассчитывается начальная доля выделяемого КР  $\delta = \frac{1}{N}$ .

2. Затем начинается поочередное выделение долей КР тем абонентам, у которых  $\alpha_i < \delta$ , то есть  $\beta_i = \alpha_i$ . Выделение долей происходит до тех пор, пока выполняется условие  $\alpha_i < \delta$ .

3. Как только находится абонент, у которого  $\alpha_j > \delta$ , происходит перерасчет начальной выделяемой доли КР:

$$\delta = \frac{1 - \sum_{j=1}^{i-1} \beta_j}{N - i + 1} \quad (8)$$

После перерасчета продолжается выделение абонентам долей КР.

4. Пункты (2)-(3) повторяются до тех пор, пока не будут выделены доли КР всем абонентам или не будет найден абонент, у которого после перерасчета  $\alpha_j > \delta$ . Тогда ему и всем последующим абонентам выделяются доли КР, равные  $\delta$ .

Из описания алгоритмов следует, что сложность 1-го и 2-го алгоритмов практически одинакова и определяется сложностью сортировки массива из  $N$  элементов. Алгоритм 3 имеет более высокую сложность, так как кроме сортировки может выполняться неоднократный перерасчет начальной доли  $\delta$ .

Результаты моделирования (рис.2)

Для решения описанных выше задач предполагалось, что запросы всех абонентов одинаковы, то есть  $R_1 = R_2 = \dots = R_N$ . Значения  $C_j$  задавались исходя из следующих соображений. При удалении абонента от БС пропускная способность его канала уменьшается, а число абонентов на заданном расстоянии от БС увеличивается приблизительно пропорционально этому расстоянию, т.е. более вероятным являются малые значения  $C_j$ .

Общие выводы, которые можно сделать по проведенным исследованиям, следующие:



1. Алгоритм 3 проигрывает по количеству удовлетворенных абонентов и по объему переданной информации.

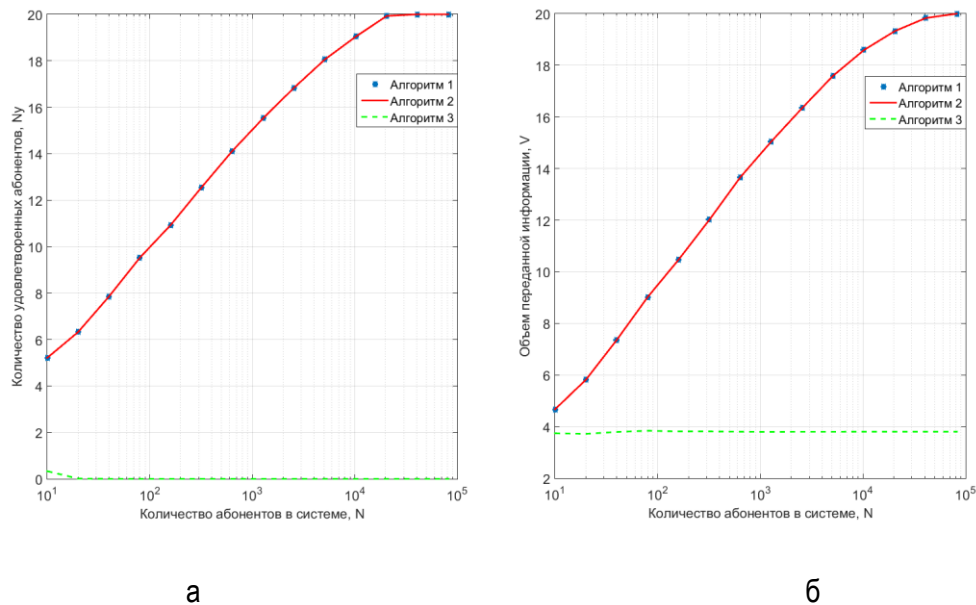


Рис.2. Результаты моделирования:

а – зависимость количества удовлетворенных абонентов от загрузки канала,  
 б – зависимость объема переданной информации от загрузки канала

2. Алгоритм 1 и алгоритм 2 приводят к одинаковому результату в связи с тем, что у всех абонентов одинаковые запросы ( $R$ ). Из этого следует, что сортировка в алгоритме 1 и сортировка в алгоритме 2 приводит к одинаковым результатам.

3. На графиках видно, что при большом количестве абонентов в системе и число удовлетворенных абонентов и объем переданной информации равны 20. Это связано с тем, что в системе увеличивается количество абонентов с пропускной способностью канала, близкой к максимальной. Следовательно, увеличивается количество переданной информации и количество удовлетворенных абонентов. Но в какой-то момент количество абонентов с пропускной способностью каналов, близкой к максимальной, становится слишком большим, и это перестает влиять на объем переданной информации и количество удовлетворенных абонентов.

4. По графикам были получены следующие регрессионные зависимости:

$N_y = 5 \log_{10}(N)$ ,  $V = 4.5 \log_{10}(N)$ , которые в интервале  $10 \leq N < 10^4$  хорошо аппроксимируют результаты моделирования.

5. В дальнейших исследованиях планируется усложнить алгоритмы следующим образом: ввести в задачу №2 ограничение снизу на суммарную скорость передачи и учесть таким образом в одной задаче – оба критерия.

**Библиографический список:**

1. Зюко А. Г., Кловский Д.Д., Коржик В. И., Назаров М.В., 1.2 Системы, каналы и сети связи // Теория электрической связи / Под ред. Д. Д. Кловского: учебник для вузов. М.: Радио и связь, 1999. С. 14–15. 432 с.
2. Олифер В.Г. Олифер Н.А./ Компьютерные сети. Принципы, технологии, протоколы. СПб.: Питер, 2001. 672 с.
3. Виснадул Б.Д., Лупин С.А., Сидоров С.В.; Под ред. Л.Г.Гагариной. учеб. пособие. М.: ИД ФОРУМ: НИЦ Инфра-М, 2012. 272 с.
4. Пролетарский А.В., Баскаков И.В., Чирков Д.Н. Беспроводные сети Wi-Fi/М.: Интуит, 2007. 177 с.

УДК 629.735.33

**В. В. Костин**

магистрант кафедры вычислительных систем и сетей

**А. В. Никитин**

канд. техн. наук, доц. – научный руководитель

### СТРАТЕГИЧЕСКАЯ ИГРА. МУЛЬТИЮЗЕР

Динамика развития компьютерных игр показывает, что игроки предпочитают многопользовательские игры перед одиночными. Согласно статистике (рис 1.) больше всего игроки затрачивают времени на многопользовательские игры.

В нашем проекте разрабатывается стратегическая игра, в которой все внимание сосредотачивается не на управлении юнитами, а на выборе зданий для строительства. Здания генерируют различных юнитов с привязкой ко времени, раз в несколько секунд. После чего юниты самостоятельно направляются к вражескому замку и атакуют его или юнитов противника, если таковые имеются.

Если брать во внимание весь спектр стратегий, то аналогов данной игры довольно много, но нужно понимать, что жанр стратегии довольно обширный и сильно отличается по геймплею. Единственный полностью кросс-платформенный аналог – стратегия «TotalWarBattles: Kingdom», однако, это пошаговая стратегия, из-за чего геймплей разительно отличается.

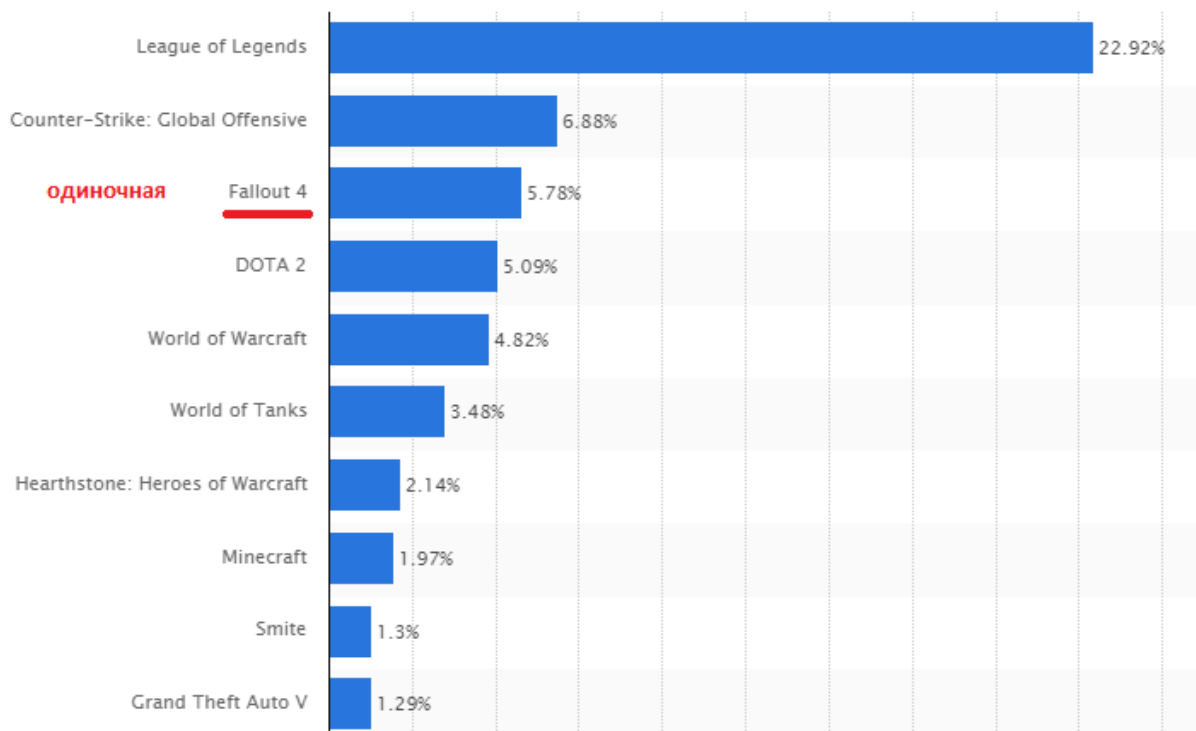


Рис 1. Время, затраченное на игру по статистике сайта statista.com[1]

На текущее время не существует стратегии в реальном времени, в которой игроки взаимодействовали бы, как с PC, так и с мобильных платформ, при данном геймплее. Похожий режим был реализован в игре Warcraft3, карте «Битва Замков», где были 2 команды и у каждой по одному главному замку. Необходимо было строить здания, в которых генерировались юниты за определенный промежуток времени, ими нельзя было управлять, они самостоятельно

двигались в направлении противника, атакуя все, что встретится на пути. Целью было уничтожение вражеского замка. Эта идея и взята за основу игры.

Целью работы является разработка мультиплеерной части стратегической игры, что позволило бы географически распределенным игрокам взаимодействовать друг с другом. Задачами являются анализ существующих технологий реализации мультиплеера, выбор наиболее подходящего решения, реализация мультиплеера на базе этой технологии.

Всего существуют 4 технологических решения: Peer-to-Peer или равный с равным, звездообразная связь, клиент-сервер и многосерверная модель. Peer-to-Peer это – одноранговая сеть, в которой каждый соединен с каждым и имеет равные права, вся информация обрабатывается на своей машине, требуется установленное соединение каждого с каждым, передается большое количество информации. Звездообразная связь это промежуточная модель между Peer-to-Peer клиент-сервером, в ней есть сервер, однако он выполняет не самую главную роль. Клиент-сервер представляет собой наличие сервера, к которому подключаются другие машины. Вся важная информация обрабатывается и синхронизируется через сервер, передается не так много информации, меньше нагрузка на клиентов. Многосерверная модель предполагает наличие более 1 сервера.

Для нашей игры удобно использовать только 2 из 4 возможных реализаций, это Peer-to-Peer и Клиент-Сервер. Учитывая, что игра предполагает кросс-платформенность, а это значит наличие совсем мощных клиентов, то удобнее использовать клиент-серверную модель, в которой вся нагрузка ложится на плечи сервера. Необходимости в молниеносной синхронизации всего мира нет, так как отсутствует возможность управлять юнитами, они двигаются по заранее определенному маршруту от замка к замку.

Для реализации проекта выбран Unity3D, так как он имеет ряд преимуществ. Начиная с версии 5 Unity является полностью бесплатным, однако имеет Проверку с дополнительными возможностями. Кроссплатформенность, код с минимальными изменениями будет запускаться на любых платформах. Обширная аудитория, благодаря чему доступно большое количество мануалов, и ускоряет исправление ошибок. Хранилище данных, в котором можно скачать уже готовые модели, процедуры, модули, значительно ускоряет разработку.

Возможны две реализации, это реализация на базе сервера Unity, а так же создание своего собственного сервера. Создание своего сервера предполагает написание своей серверной части, что является длительным по времени процессом и требует знаний в области программирования на языках типа Python, в связи с чем, предпочтительным был выбор сервера на базе Unity. Плюсы реализации на базе Unity в том, что его просто использовать, а также это экономит время, которое нужно было бы потратить на написание собственного сервера.

Для проверки возможностей Unity3D была создана тестовая сцена в 3D'sMax, импортирована в Unity3D, и на ее базе были произведена оценка возможностей Unity3D. После было решено, что сервер на базе Unity подходит для решения поставленной задачи.

Таким образом был осуществлен поиск и сравнение аналогов, произведен выбор технической реализации мультиюзера, создана тестовая сцена, опробованы возможности Unity3D.

В дальнейшем планируется реализовать прототип игры с начальным функционалом, после чего можно будет протестировать их на реальных устройствах, измерить нагрузку, оптимизировать игру и расширить функционал.

### Библиографический список

1. Сайт статистики [Электронный ресурс] :<http://statista.com>(дата обращения: 13.03.2016)
2. Тэрбин П. Стратегические игры. М.: Баланс-Клуб 2003. 264 с.
3. Васкевич Д. Стратегии клиент/сервер. Диалектика, Киев, 1997.
4. Жирнов А.Н. и др. Основы работы Unity3D. СПб:ГУАП, 2013. 63 с.
5. Полушкина Е.А. и др. Основы работы в Unity3D(Часть 2).СПб:ГУАП, 2015. 79с.

УДК 681.3.06

**Ю. А. Кузнецова**

магистрант кафедры проблемно-ориентированных вычислительных комплексов

**Г. С. Евсеев**

канд. техн. наук, доц.) – научный руководитель

### ТРЕНАЖЕР ДЛЯ ВВОДА СМЕСИ РАЗНОЯЗЫКОВЫХ ТЕКСТОВ

Рассматривается программный клавиатурный тренажер для ввода смеси разноязыковых текстов.

Существует большое число тренажеров по вводу текста. Самые популярные тренажеры VerseQ, Stamina, Klavagog и т.д. Общая цель этих тренажеров - достичь высокой скорости ввода текста при малом числе опечаток. На основе выполненного обзора [2] предлагается расширить функции тренажера, а именно, добавить следующие функции:

- изменение условий для ввода текста;
- оценка влияния таких изменений на характеристики оператора;
- оценка адаптируемости оператора.

В данном тренажере проводится испытание, которое позволяет оценить следующие характеристики (рис.1):

- скорость и качество ввода сплошного текста;
- скорость и качество ввода перемешанного текста;
- адаптируемость оператора к изменению условий ввода текста.

Испытание состоит из двух этапов. На первом этапе вводится сначала сплошной русский текст, а затем – сплошной английский текст. При этом измеряется время ввода каждого текста и число ошибок в нем.

На втором этапе вводится смесь из русского и английского текстов. Объемы текстов на обоих этапах одинаковы (1000 символов). При этом также измеряется время ввода текста и число ошибок в нем. На прохождение всего испытания дается время 30 – 40 мин.

Этапы испытания показаны на рис. 1.

## ЭТАПЫ

Этап 1		Этап 2	
Русский текст 400 сим.	Английский текст 400 сим.	Русский 50 сим.	Русский 25 сим.
			Английский 25 сим.
		Английский 50 сим.	Русский 25 сим.
			Английский 25 сим.
		Русский 50 сим.	Русский 25 сим.
			Английский 25 сим.
		Английский 50 сим.	Русский 25 сим.
			Английский 25 сим.
		Русский 50 сим.	Русский 25 сим.
			Английский 25 сим.
Английский 50 сим.	Русский 25 сим.		
	Английский 25 сим.		
Русский 50 сим.	Русский 25 сим.		
	Английский 25 сим.		
Английский 50 сим.	Русский 25 сим.		
	Английский 25 сим.		

Рис. 1. Этапы испытания

Качество работы оператора оценивается по двум параметрам:

- средняя скорость набора текста определяется временем ввода текста и его объемом;
- средняя частота опечаток (отношение числа опечаток к объему текста).

В испытании изменяется интенсивность перемешивания текстов (число разноязычных фрагментов во вводимом тексте). Как следует из рисунка 1, эта величина изменяется от 1 при вводе сплошного текста до 16. По результатам испытания строятся следующие зависимости (рис. 2):

- Инт- интенсивность перемешивания текста;
- Трус(Инт) – время ввода фрагментов русского текста;
- Танг(Инт) – время ввода фрагментов английского текста;
- Крус(Инт) – доля ошибок во фрагментах русского текста;
- Канг(Инт) – доля ошибок во фрагментах английского текста;

Для каждой из перечисленных зависимостей строятся линии линейной регрессии и вычисляются коэффициенты регрессии (tg угла наклона)  $RTрус$ ,  $RTанг$ ,  $RКрус$ ,  $RКанг$ .

На рис.2 показан пример построения зависимостей при вводе русского текста. Оба графика показывают, что с увеличением интенсивности перемешивания текста характеристики оператора ухудшаются: время, затраченное на ввод текста, увеличивается и число ошибок возрастает.

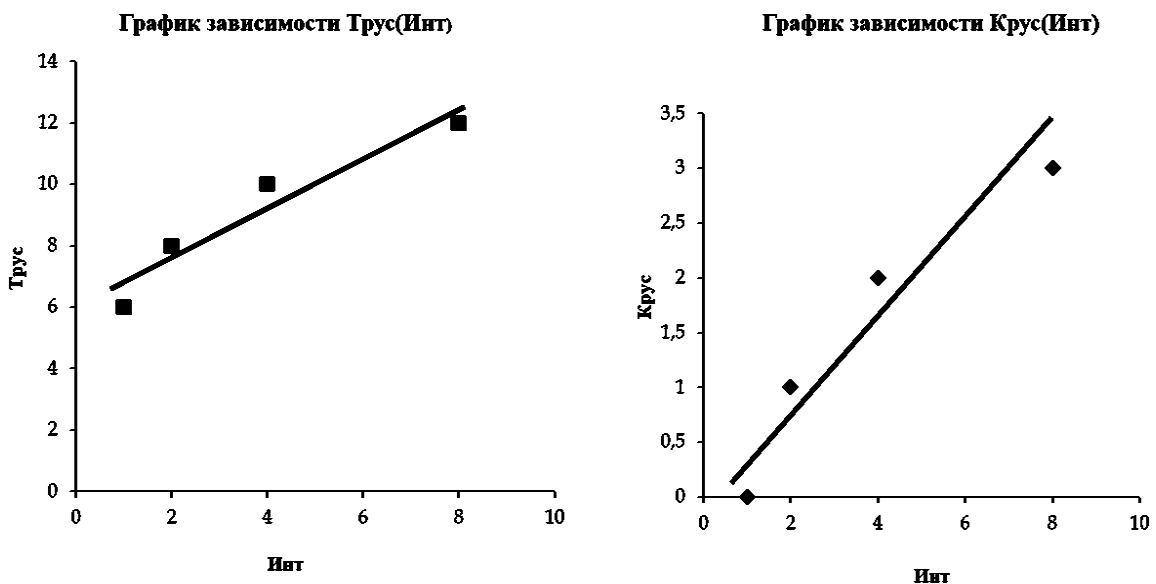


Рис. 2. Графики зависимостей

Критерий оценки адаптируемости оператора предлагается вычислять по формуле

$$Кадап = C_1 RTрус + C_2 RTанг + C_3 RКрус + C_4 RКанг,$$

где весовые коэффициенты задаются субъективно. Например, если на рабочем месте предполагается ввод информации преимущественно на английском языке, то значения для  $C_2$  и  $C_4$  устанавливаются выше, чем для  $C_1$  и  $C_3$ . Соответственно если ввод информации преимущественно на русском языке, то порог для  $C_1$  и  $C_3$  устанавливается выше, чем для  $C_2$  и  $C_4$ . Если графики зависимостей горизонтальны, то адаптируемость кандидата считается идеальной.

При использовании тренажера для приема на работу аттестация кандидатов может включать следующие действия. Если при прохождении первого этапа (ввод сплошного текста) у претендентов количество ошибок/опечаток больше установленного, то часть кандидатов

отсеивается. Аналогичное отсеивание происходит по второму критерию (время затраченное на ввод).

Таким образом, предлагаемая методика позволяет количественно оценивать и сравнивать уровни адаптируемости операторов при вводе разноязычных текстов.

#### **Библиографический список**

1. Попов М. П. Эффективные приемы набора и редактирования текста. СПб.: БХВ Петербург, 2006.
2. <http://www.urikor.net/rus/Art03.html> - общий сайт.

**УДК 378.16:004.383.4:004.31-022.53**

**Н. С. Легкий, П. А. Степанов**

студенты кафедры проблемно-ориентированных вычислительных комплексов

**О. А. Кононов**

доц., канд. техн. наук, доц. – научный руководитель

### **РАЗРАБОТКА КОМПЛЕКСА ЛАБОРАТОРНЫХ РАБОТ ПО КУРСУ "ОСНОВЫ МИКРОПРОЦЕССОРНОЙ ТЕХНИКИ" НА ОСНОВЕ ОЦЕНОЧНОЙ ПЛАТЫ STM32F4DISCOVERY**

В настоящее время все большую популярность приобретают 32-битные микроконтроллеры (далее МК), стоимость которых становится сопоставимой со стоимостью 8-битных при существенно большей производительности. В связи с этим представляется целесообразным ввести в курс изучения микропроцессоров практические работы с 32-битными МК. Программирование подобных устройств удобно изучать с использованием широко доступной на рынке отладочной платы STM32F4Discovery. В составе платы используется 32-битный ARM микроконтроллер STM32F407VGT6 семейства Cortex-M4. Микроконтроллер выполнен с возможностью питания 1,8–3,6 В; максимальной частотой работы до 168 МГц; с внутренней памятью программ (Flash) до 1 Мбайт и RAM 196 Кбайт; 4 Кбайт backup SRAM; имеется 82 контакта для программируемого ввода и вывода(I/O) или 82xGPIO; поддерживает инструкции DSP с возможностью отладки как по SWD так и по JTAG каналам. Кроме того на «борту» находятся 16- и 32-битные таймеры, full-speed-USB 2,0, MAC уровень10/000 Ethernet, IEEE 1588v2, MII/RMII, 3xSPI, 3xI2G, 4xUSART, 2xUART(10,5 Мбит/с), ISO 7816, LIN, IrDA, 2xCAN, SDIO(SD) DCMI( до 54 Мбайт/с), RTC, 3x12 бит, АЦП (52 МГц); 2x12битных ЦАП, DMA(16 каналов + FIFO, FSMC(Compact Flash, SRAM, PSRAM, NOR, NAND); отладчик STLink выведен разъемом mini-USB, предназначенный для отладки и прошивки программы МК; все 100 выводов МК выведены по бокам платы, среди них есть выводы для внешнего питания 5 и 3,3 В. На плате присутствует кнопка сброса, четыре светодиода и одна пользовательская кнопка. LIS302DL – MEMS-датчик движения, 3x-осевой цифровой акселерометр; MP4DT02 – цифровой MEMS-микрофон; CS43L22 – аудио-ЦАП со встроенным драйвером динамиков класса D. Для USB OTG выведен разъем micro-USB.

Программы для контроллеров ARM пишутся, как правило, на языке C. Для разработки программы и ее записи в память микроконтроллера существует много интегрированных сред разработки.

Keil MDK–ARM (Keil uVision) – среда разработки, представляющая собой набор утилит для выполнения полного комплекса мероприятий по разработке приложений для микроконтроллеров.

Keil MDK–ARM позволяет работать с проектами любой сложности, начиная с введения и правки исходных текстов и заканчивая внутрисхемной отладкой кода и программированием памяти микроконтроллера. Компиляторы Keil работают с текстами, написанными на Си или

ассемблере. Кроме того возможно использование компиляторов других производителей. Встроенный редактор облегчает работу с исходным текстом за счет использования многооконного интерфейса, выделения синтаксических элементов шрифтом и цветом. Имеется библиотека функций с типовыми задачами. Также стоит отметить наличие отладчика-симулятора, позволяющего отлаживать работу скомпилированной программы на виртуальной модели микропроцессора. В нем довольно достоверно моделируется работа ядра контроллера и его периферийного оборудования.

Keil MDK–ARM поддерживает огромное количество ARM- микропроцессоров, что делает его уникальным, по сравнению с другими средами разработки, также эта среда имеет огромную базу данных микроконтроллеров, содержащую подробную информацию обо всех поддерживаемых устройствах. В ней хранятся их конфигурационные данные и ссылки на источники информации с дополнительными техническими описаниями.

Интерфейс Keil MDK–ARM- на английском языке. Программа работает на операционной системе Windows. Распространяется программа в версиях: Demo – бесплатная версия с рядом ограничений, а также с ограничением размера кода программы 32 Кбайт и Standard и Professional – платные (от 1500 евро). Для целей обучения демонстрационной версии вполне достаточно, так как основной задачей является инициализация таймеров, портов, выводов, настройка периферии и взаимодействие с ней. Мы уже подготовили базовый комплекс лабораторных работ, опирающийся, главным образом, на примеры использования аппаратуры микроконтроллера и отладочной платы. В дальнейшем мы планируем расширить комплекс за счет применения оборудования, подключаемого к разъемам платы.

В базовый комплекс лабораторных работ входят:

1. Управление светодиодами с помощью кнопки.
2. ШИМ – управление яркостью светодиодов.
3. Работа с таймерами. Аппаратный ШИМ.
4. Работа с акселерометром.
5. RNG. Работа с генератором случайных чисел.

В расширенный комплекс лабораторных работ сейчас входят:

1. Прием-передача данных по UART.
2. Работа с АЦП.

В отличие, например, от 8-разрядных микроконтроллеров семейства MCS-51, применение микроконтроллеров семейства STM32F4 требует более сложной процедуры настройки портов ввода/вывода. Для демонстрации сказанного рассмотрим первую лабораторную работу.

Чтобы управлять светодиодами, необходимо знать, к каким выводам микроконтроллера они подключены. Для этого обратимся к рис 1.

Светодиоды подключены к выводам PD12, PD13, PD14, PD15.

Эти выводы нужно настроить. Фрагмент кода, который пригодится для их настройки:

```
RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIO,ENABLE); //Включаем тактирование порта D;
```

```
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_12|GPIO_Pin_13|GPIO_Pin_14|GPIO_Pin_15; //
```

Выбираем нужные выводы;

```
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_OUT; // Устанавливаем режим работы «выход»;
```

```
GPIO_Init(GPIO, &GPIO_InitStructure); //Запускаем вызов функции инициализации.
```

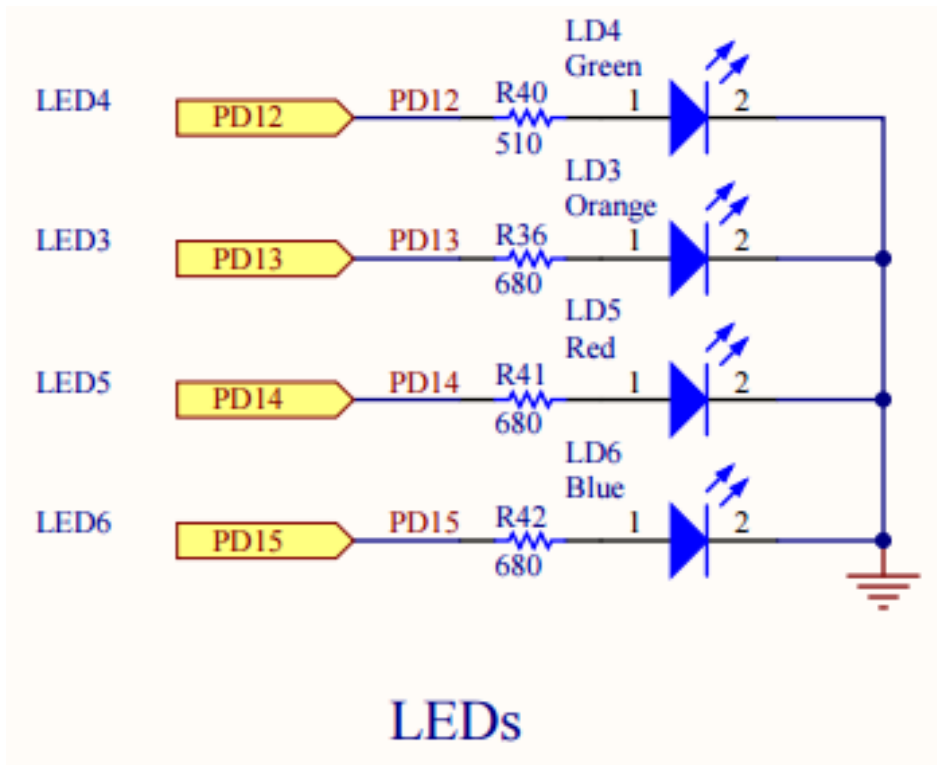


Рис. 1. Подключение светодиодов к выводам микроконтроллера

Кроме этого нам нужно знать, к какому выводу подключена кнопка пользователя. С помощью кнопки мы сможем управлять режимом работы светодиодов. Обратимся к рис 2.<sup>1</sup>

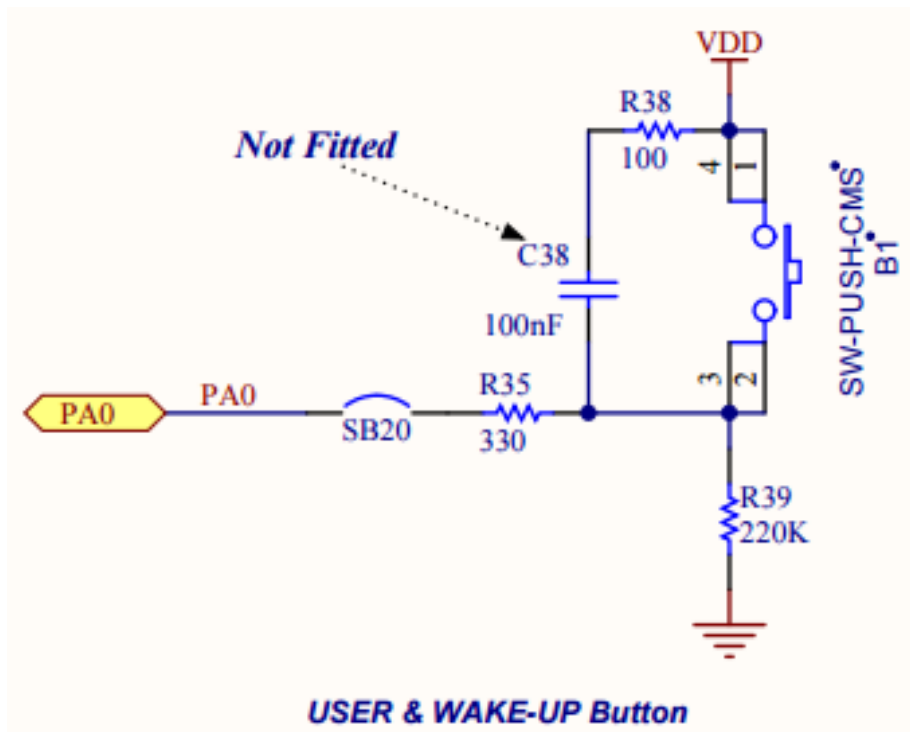


Рис. 2. Подключение кнопки к выводу микроконтроллера

<sup>1</sup> Техническая спецификация  
<http://www2.st.com/content/ccc/resource/technical/document/datasheet/ef/92/76/6d/bb/c2/4f/f7/DM00037051.pdf/files/DM00037051.pdf>



```

Кнопка подключена к выводу PA0. Этот вывод так же необходимо настроить:
GPIO_InitTypeDef GPIO_InitStructure; //Структура содержащая настройки порта;
RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIOA, ENABLE); //Включаем тактирование
порта A;
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_0; //Задаем номер вывода;
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_IN; //Устанавливаем режим работы
«цифровой вход, подтяжка к плюсу»
GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_DOWN; //Включаем подтяжку к «земле»;
GPIO_Init(GPIOA, &GPIO_InitStructure); //Вызываем функцию инициализации.
    
```

При работе с кнопкой нужно учитывать такое явление как дребезг контактов. Перед тем, как наступит плотный контакт, некоторое время будут продолжаться колебания, т.е. замыкания и размыкания, вследствие чего порождается множество срабатываний вместо одного.

В следующем году мы будем рады поделиться с вами опытом применения разработанного комплекса в учебном курсе «Основы микропроцессорной техники».

## УДК 004.6

### **Е. А. Логвинчук**

магистрант кафедры компьютерных технологий и программной инженерии

### **А. В. Бржезовский**

канд. техн. наук, доцент – научный руководитель

## **РАСШИРЕНИЕ ФУНКЦИОНАЛЬНОСТИ OLAP-СЕРВЕРА ДЛЯ ПОВЫШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ ОТ ЕГО ИСПОЛЬЗОВАНИЯ**

Проблема обработки и хранения больших объемов данных, которые со временем накапливаются и расширяются, рассматривается первоочередной задачей во многих сферах деятельности. Существуют разнообразные способы решения данной задачи: реорганизация существующих решений на основе работающих систем управления базами данных (СУБД), внедрение дополнительных модулей для работы с большими объемами данных, переход к новым СУБД с полным или частичным отказом от старой конфигурации базы данных (БД), но в любом случае, использование одной лишь нормализованной модели данных или даже одних OLTP-систем (OnLine Transaction Processing) оказывается недостаточно для решения многих задач.

В большинстве организаций периодически встает задача проведения определенного анализа работоспособности различных программных компонентов: выявление периодичности отказов сервисов, определение количества вызовов программных методов, поиск самого долгого метода по времени выполнения и т.д.

Для анализа данных традиционной работы с таблицами БД оказывается недостаточно. Решением данной задачи может быть использование активно развивающейся технологии OLAP (OnLine Analytical Processing) – технологии обработки данных в режиме реального времени, заключающейся в подготовке агрегированной (суммарной) информации на основе больших объемов данных, структурированных по многомерному принципу. В структуре реляционных БД сложные многотабличные запросы, присущие аналитической обработке, будут выполняться крайне медленно. Благодаря особой структуре БД OLAP-хранилища происходит ускорение доступа к данным, а многомерные запросы на выборку данных строятся интуитивно понятнее и ближе к предметной области, чем традиционные SQL-запросы.

Целью настоящей работы является исследование способов взаимодействия с OLAP-сервером и поиск оптимального варианта хранения данных в БД, в частности за счет расширения функциональности OLAP-сервера.

Перед работой были поставлены следующие задачи:

1. Реализовать надежное и управляемое взаимодействие с OLAP-сервером.
2. Изменить структуру базы данных OLAP для более эффективного использования механизма очистки таблиц.
3. Расширить функциональность OLAP-сервера для увеличения скорости выполнения запросов.

В основе технологии OLAP лежит многомерная модель данных. Информация в многомерной модели содержится в виде многомерных наборов данных [1], которые именуются гиперкубами. Многомерным набором данных называется отношение с  $n$  независимыми атрибутами (множество измерений) и с определенным количеством зависимых атрибутов (множество мер). Измерения задают систему координат пространства данных и являются набором доменов, по которым строится многомерное пространство. Каждое измерение состоит из конечного множества элементов, которые могут группироваться в иерархии по одному или нескольким уровням. Иерархия является важным свойством многомерных кубов, так как позволяет добавлять новые уровни в аналитическое поле пользователя. Меры представляют собой анализируемые характеристики, как правило, представленные в числовой форме. Пользователи получают для анализа так называемые срезы или проекции куба, которые можно отобразить в виде таблиц или графиков.

По способу хранения данных [2] OLAP-продукты делятся на многомерные, реляционные и гибридные. В первом варианте все данные хранятся в многомерной БД, представленной в виде гиперкуба. Во втором хранение происходит на уровне реляционной БД с возможностью создавать отдельные таблицы для хранения агрегированных данных. Третий вариант представляет комбинацию двух предыдущих подходов, сохраняя агрегированные данные в многомерной БД, а все детальные данные – в реляционной БД, тем самым обеспечивая компромисс между двумя подходами.

В данной работе в качестве продукта аналитической обработки данных рассматривается OLAP-сервер Mondrian компании Pentaho. Сервер имеет открытый исходный код и написан на языке Java. Mondrian не является так называемым «коробочным» продуктом, т. е. его нельзя установить на диск и запустить. Исходные файлы сервера выкачиваются из открытого репозитория и затем собираются в единый проект. Сервер работает по системе ROLAP. Все детальные данные хранятся в таблицах фактов, а суммарные – в агрегационных таблицах. В качестве СУБД будет рассматриваться PostgreSQL (Postgres).

Первая поставленная задача имела следующие рассуждения и решения. В организации, для которой проводится исследование, основным языком разработки является C++, а сервер Mondrian работает с Java. Поэтому изначально при взаимодействии с OLAP-сервером использовался механизм JNI (Java Native Interface), который позволяет запускать функции Java из кода на C/C++ и наоборот. Но у данного способа присутствуют недостатки. Первой проблемой оказался рост оперативной памяти процесса с Java. Библиотека JNI и сервер Mondrian работают на виртуальной Java-машине, которая запускается в отдельном платформенном процессе. Специальные программы-диспетчеры внутри компании отслеживают данные процессы и при достижении объема памяти процесса в 400 Мб убивают его. Второй проблемой стала организация работы с JNI. Периодически возникали утечки памяти. Велась работа по их устранению, но до конца она не была доведена. Также у JNI имелись свои ошибки в реализации, которые не были исправлены, а последние версии выходили более десяти лет назад, соответственно исправленных версий не предвиделось. Решением описанных проблем стало написание собственного приложения на Java, которое работает с сервером Mondrian без посредников. Приложение запускает процесс Mondrian, и дальше клиенты взаимодействуют с

приложением, которое отправляет пользовательские многомерные запросы на сервер, обрабатывает их и возвращает результат пользователю. Работа с сервером на родном для него языке Java оказалась наилучшим вариантом использования для дальнейшей поддержки и оптимизации.

В рамках второй задачи следует пояснить, что для анализа информации рассматривается куб данных, состоящий из трех измерений: сервис, метод и время. Данные по измерению времени могут запрашиваться по трем уровням иерархии: день, час и десятиминутка. Поскольку каждый запрос содержит время, то агрегационных таблиц всего две: ВремяСервис и ВремяМетод. Изначально было понятно, что хранить все статистические данные в одной таблице фактов и извлекать информацию из агрегационных таблиц слишком накладно в плане времени выполнения запросов, так как таблицы могут хранить около десятка миллиона записей. Первым шагом стало использование трех баз данных, имеющих одинаковую структуру и хранящих статистику за разные временные интервалы (три дня, десять дней и 60 дней). БД за десять дней хранила данные с уровнем не ниже часов, БД за 60 дней – с уровнем не ниже дней. Такой подход позволял обращаться к конкретным таблицам и не хранить ненужную для аналитики информацию, например, часы или десятиминутки за давнее число. Но вместе с этим были и недостатки. Во-первых, поддержка трех баз данных: если необходимо что-то изменить в структуре базы, то приходилось менять во всех трех, это не всегда быстро и удобно. Во-вторых, объем таблиц, который все равно остался на уровне миллиона записей. В-третьих, применение операции удаления записей, которая в Postgres не сразу очищает таблицу от старой версии кортежа данных. Такой подход позволяет получить преимущества версионного контроля, но из-за лишних пустых строк растет дисковое пространство. Команда VACUUM удаляет старые версии строк и индексов, а команда VACUUM FULL еще и уплотняет таблицы, но на все эти действия тратится дополнительное время и память. Хорошим решением является применение команды TRUNCATE [3], полностью очищающей таблицу от данных. Но такой подход применим, только если данные из таблицы можно полностью удалить без потери информативности. Решением стала реализация секционированных таблиц. Процесс секционирования реализуется через логическое разбиение одной большой таблицы на маленькие куски (секции), что обеспечивает ряд преимуществ: возможность применять команды полной очистки таблицы; прирост производительности за счет изолированности индексов от операций в других секциях; обращение к таблицам с меньшим количеством данных. В качестве реализации был предложен механизм разбиения на секции по дням. То есть тридцать одна версия таблицы фактов и столько же версий каждой из агрегированных таблиц.

Проблема встала в использовании OLAP-схемы самим сервером. OLAP-схема, представленная в виде xml-файла, описывает многомерную модель данных. В схеме задаются: куб данных, его измерения, уровни иерархии, меры, а также таблицы фактов и агрегационные таблицы. Для связи с базой данных необходимо задать каждому из компонентов схемы куба соответствующее название из базы. Проблема заключается в том, что сервер однозначно идентифицирует указанное в схеме название таблицы или колонки в БД. А при использовании секционированных таблиц необходимо обращаться к разным таблицам в зависимости от условия. Такое условие можно внедрить только в сам OLAP-сервер. Запросы на сервер приходят на языке MDX (MultiDimensional Expressions), который по синтаксису напоминает язык запросов к реляционным БД SQL, но отличается по основным концепциям. Основные этапы работы сервера заключаются в генерации SQL-запросов из MDX-запроса, записи результатов в свой кэш и выдачи их пользователю. Процесс генерации достаточно сложный. После разбора (парсинга) MDX-запроса и проверки его на корректность по OLAP-схеме, сервер генерирует так называемые пакеты, по которым формируются SQL-запросы. Количество SQL-запросов и их структура зависят от количества и структуры пакетов. Каждый пакет содержит следующие атрибуты: колонки, по которым будет сформирован запрос, названия мер, набор данных, которые будут использоваться в условии запроса, и ключ пакета. Если сервер видит, что запросы по одному

уровню иерархии можно объединить, он создает один пакет. Чтобы каждый запрос обращался к своей секционированной таблице необходимо увеличить количество формируемых пакетов. Для этого в состав ключа пакета было добавлено поле номера запрашиваемого дня. После этого сервер стал генерировать большее количество пакетов и соответственно запросов. Но название таблиц сервер по-прежнему брал из OLAP-схемы. Было принято решение дописывать к названиям таблиц в SQL-запросе номер дня, по которому будут формироваться данные. На этом задача расширения функциональности OLAP-сервера была решена. Она оказалась не быстрой в основном вследствие анализа работы самого сервера. После данной реализации стал наблюдаться прирост скорости выполнения запросов от 1.5 до 7.5 раз в зависимости от количества запрашиваемых дней и уровня временной иерархии.

Результаты, полученные в ходе данной работы, были успешно внедрены в компанию по разработке системы электронной отчетности и документооборота. Благодаря этим результатам скорость обработки аналитической информации стала выполняться быстрее, а сама обработка стала стабильнее.

### Библиографический список

1. Многомерные хранилища данных [Электронный ресурс] <<http://bourabai.ru/tpoi/olap01-4.htm>>.
2. Differences Between OLAP, ROLAP, MOLAP, and HOLAP [Электронный ресурс] <<http://social.technet.microsoft.com/wiki/contents/articles/19898.differences-between-olap-rolap-molap-and-holap.aspx>>.
3. PostgreSQL 9.4.8 Documentation. TRUNCATE [Электронный ресурс] <<https://www.postgresql.org/docs/9.4/static/sql-truncate.html>>.

УДК 004.777

**М. В. Манкевич**

магистрант кафедры проблемно-ориентированных вычислительных комплексов

**В. С. Павлов**

д-р. техн. наук, проф. – научный руководитель

## ПРИМЕНЕНИЕ МУЛЬТИПЛЕКСИРОВАННОГО АСИНХРОННОГО СЕТЕВОГО ВЗАИМОДЕЙСТВИЯ В ИНТЕРАКТИВНЫХ ОБУЧАЮЩИХ СИСТЕМАХ

В последние несколько лет в области информационных технологий в целом и в веб-индустрии в частности произошло большое количество качественных изменений, появились средства, которые значительно упрощают разработку программных систем и которые могли бы лечь в основу разработки эффективной электронной системы интерактивного обучения.

Объектом исследования являются системы обеспечения многопользовательского сетевого взаимодействия, применяемые для организации интерактивного обучения.

Предметом исследования является эффективность технологий организации многопользовательского сетевого взаимодействия.

Задачей данной работы является обоснование практической целесообразности использования технологии мультиплексированного асинхронного сетевого взаимодействия WebSocket для создания модулей интерактивного взаимодействия в системах электронного обучения.

Понятие эффективной обучающей системы предполагает возможность использования такой системы не только в некоторых локальных рамках, например, в рамках одного университета, но и в рамках некоторого региона, имеющего для всех своих элементов набор общих характеристик, например, в рамках всей страны.

При потенциально возможных масштабах развертывания системы, с учетом материально-технической базы учреждений образования нашей страны, единственной платформой, которая позволит обеспечить такое развертывание, является веб.

Большинство современных веб-ориентированных систем обучения не отличается широким набором предоставляемых для использования интерактивных сценариев. Это связано с тем, что до недавнего времени в веб-индустрии не существовало технологий, которые могли бы позволить реализовать такое взаимодействие. Основные технологии, используемые для организации передачи данных в веб – это технологии, базирующиеся на разработанном в 1991 году протоколе HTTP, который несет в себе ряд существенных ограничений для реализации некоторых сценариев сетевого взаимодействия<sup>1</sup>. Появление веб-ориентированной технологии мультиплексированного асинхронного сетевого взаимодействия WebSocket открывает большие возможности для реализации многих новых методик интерактивного обучения.

Для того чтобы оценить эффективность применения технологии WebSocket, был спроектирован прототип системы, реализующий основные сценарии сетевой передачи данных, которые используются при построении интерактивных веб-компонентов в многопользовательских веб-ориентированных системах электронного обучения.

Тестирование предполагало сравнение производительности наиболее совершенной из HTTP-совместимых веб-технологий, широко используемой в существующих обучающих системах, и технологии мультиплексированного асинхронного сетевого взаимодействия WebSocket.

Тестировались такие сценарии передачи данных, как передача переменного числа сообщений фиксированной длины и передача фиксированного числа сообщений переменной длины, а также проводилось измерение нагрузки на канал связи, возникающие при одновременном подключении к системе большого количества пользователей.

В ходе выполнения сценария передачи переменного числа сообщений фиксированной длины были получены результаты, представленные в табл. 1.

Таблица 1

Количество сообщений	HTTP (SSE), мс	WebSocket, мс	Отношение
10	19	14	1.32
100	120	21	5.60
500	564	72	7.79
1000	1132	124	9.15
5000	5311	537	9.89
10000	10783	1030	10.47

Данные из табл. 1 представлены на рис. 1 в виде диаграммы.

По полученным экспериментальным данным можно сделать вывод о том, что количество обращений к серверу значительно влияет на быстродействие и увеличение количества HTTP-запросов из-за увеличения накладных расходов на каждый из них, что значительно замедляет работу приложения. При данном сценарии применение протокола WebSocket позволяет в разы

<sup>1</sup> Флэнаган, Д. JavaScript – Карманный справочник. М.: Вильямс, 2015. 320 с.

увеличить скорость взаимодействия за счет отсутствия потерь на свойственные протоколу HTTP операции установления соединения.

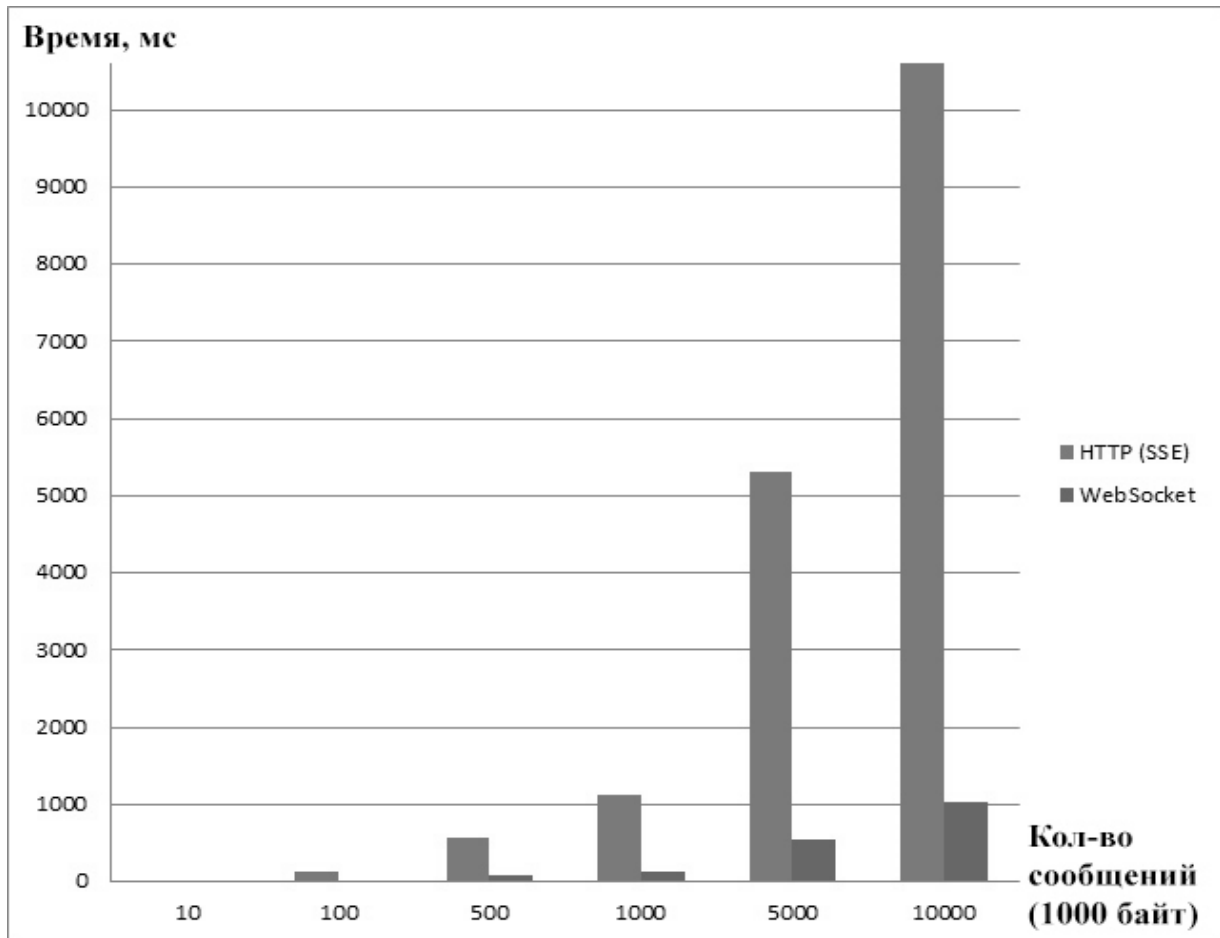


Рис. 1 – Результаты измерений при фиксированной нагрузке (1000 байт/сообщ.) для переменного числа сообщений

В ходе выполнения сценария передачи фиксированного числа сообщений переменной длины были получены результаты, представленные в табл. 2.

Таблица 2

Количество данных, байт	HTTP (SSE), мс	WebSocket, мс	Отношение
10	1003	81	12.38
100	1013	87	11.64
500	1032	113	9.13
1000	1044	119	8.77
5000	1173	243	4.83
10000	1289	394	3.27

Данные из табл. 2 представлены на рис. 2 в виде диаграммы.

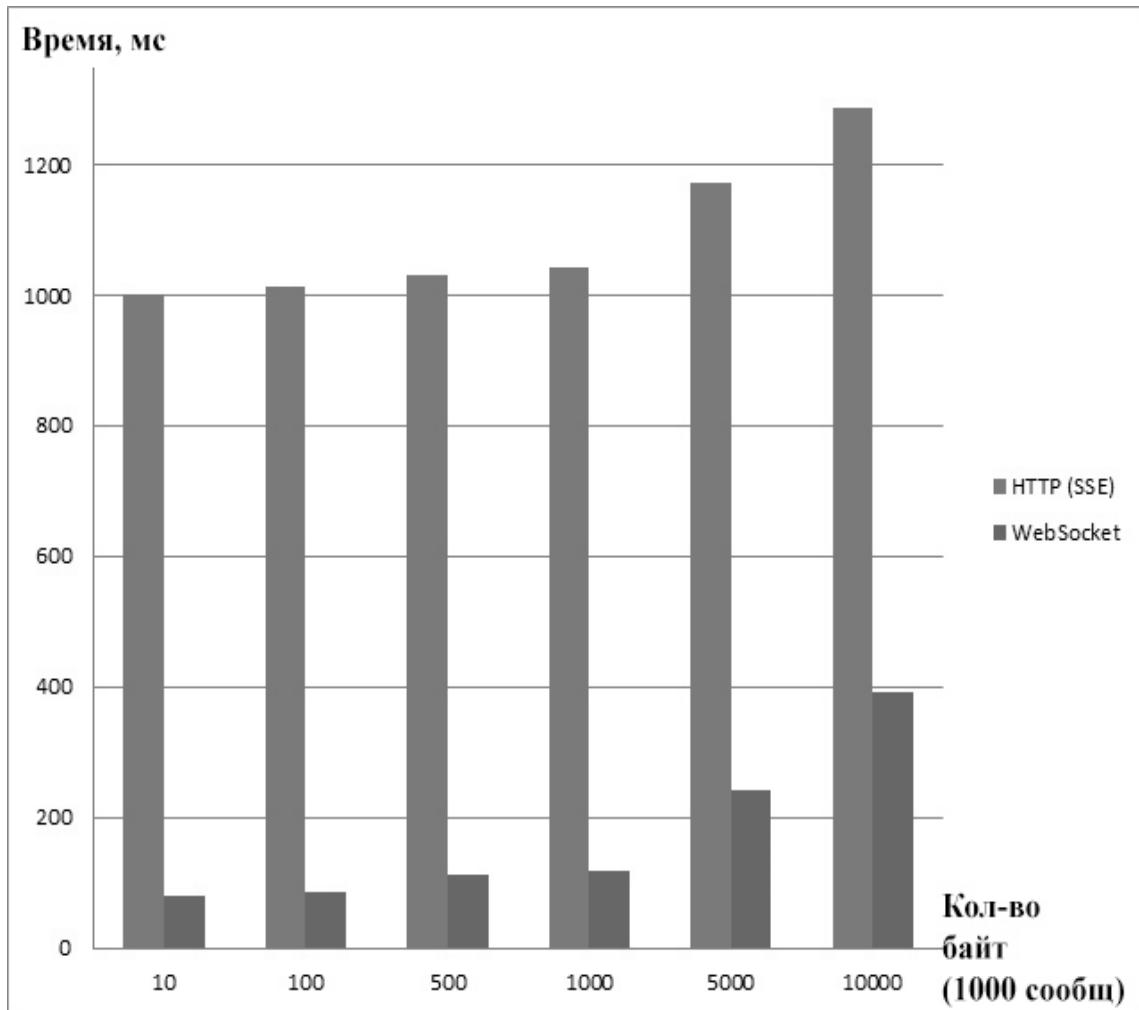


Рис. 2. Результаты измерений при переменной нагрузке для фиксированного числа сообщений

По полученным экспериментальным данным можно сделать вывод о том, что рост объемов передаваемых сообщений сказывается на относительной производительности минимально, т.к. основные потери времени происходят во время выполнения операций инициализации и закрытия соединений. При этом за счет отсутствия накладных расходов на передачу метаданных, использование протокола WebSocket позволяет в разы уменьшить время передачи сообщений малого размера. Следовательно, применение протокола WebSocket в данном сценарии позволяет получить ощутимый прирост производительности при передаче сообщений небольшого размера, а при передаче сообщений большого размера позволяет получить производительность не хуже, чем HTTP.

В ходе выполнения сценария подключения большого количества пользователей было выявлено, что минимальный размер HTTP сообщения, не содержащего никаких данных (только обязательный заголовок), составляет 871 байт, а размер аналогичного пустого сообщения для WebSocket равно 2 байта. На основании этой информации был проведен расчет, данные которого представлены в табл. 3.

Таблица 3

Количество клиентов	HTTP, МБит/с	WebSocket, МБит/с	Отношение
1000	6.6	0.015	440
10000	66	0.153	431
100000	665	1.526	435

По полученным экспериментальным данным можно сделать вывод о том, что использование протокола WebSocket в системах, предназначенных для одновременного подключения большого количества пользователей, оправдано с точки зрения расходов, связанных с обеспечением достаточно широкого входного канала связи.

Исходя из абсолютных значений, полученных в тестах, можно сделать следующие выводы.

В рамках небольшой локальной системы интерактивного многопользовательского взаимодействия эффективность применения WebSocket может быть не всегда оправдана за счет недостаточно высоких нагрузок в таких системах, при которых HTTP (SSE) сможет показать достаточную для комфортной работы пользователей производительность.

В рамках крупной многопользовательской системы, такой как система дистанционного обучения, предполагается возможность реализации различных высокоинтерактивных сценариев взаимодействия, при которых производительности HTTP (SSE) будет недостаточно, поэтому в таких системах оправдано использование WebSocket. Кроме этого, использование HTTP (SSE) не позволяет реализовать ряд сценариев, в которых серверное приложение может выступать полноценным участником взаимодействия, например, являясь поставщиком событий по расписанию. Протокол WebSocket является более универсальным и не содержит ограничений по направленности и типам взаимодействий. При этом использование WebSocket при большом количестве одновременных подключений создает меньшую нагрузку на сервер, чем HTTP (SSE), поэтому его использование в интерактивных многопользовательских приложениях более предпочтительно.

Опираясь на полученные данные можно с уверенностью заявить, что использование технологии мультиплексированного асинхронного сетевого взаимодействия WebSocket для создания модулей интерактивного взаимодействия для электронных обучающих систем более чем оправдано и несет в себе большой потенциал для дальнейшего развития такого рода систем.

**УДК 629.735.33**

***О. О. Масимчук***

магистрант кафедры вычислительных систем и сетей

***А. В. Гордеев***

док. техн. наук, профессор – научный руководитель

## **СРАВНИТЕЛЬНЫЙ АНАЛИЗ СИСТЕМ ДИСТАНЦИОННОГО ОБУЧЕНИЯ (СДО)**

### **Введение**

На сегодняшний день технология дистанционного образования является наиболее распространенным способом обучения, для тех, кто не имеет возможности лично посещать лекции, он дает возможность таким людям повысить свою квалификацию, получить достойное образование не выходя из дома.

По сути, Интернет-технологии становятся связующим звеном между студентом и преподавателем, которых могут разделять тысячи километров друг от друга. ДО – это образовательный процесс, базирующийся на самостоятельном обучении студента.

### **Актуальность**

С появлением большого количества различных программных средств, реализующих дистанционное обучение, возникает потребность их анализа для выбора более подходящего варианта. Целью данной статьи является, обзор и сравнительный анализ программных средств ДО.



Далее в статье будут рассмотрены следующие задачи:

1. Характеристика программных средств ДО.
2. Сравнительный анализ программных средств ДО.
3. Выбор наиболее подходящего программного продукта для реализации курса.

Первая часть посвящена обзору автоматизированных информационных систем дистанционного образования. Во второй части проведен сравнительный анализ систем. В заключении – выводы по результатам.

## 1. Характеристики СДО

### 1.1. Система Дистанционного Обучения Moodle.

СДО Moodle – это программное обеспечение, предназначенное для создания и использования дистанционных курсов. Система Moodle способна конкурировать с известными коммерческими системами дистанционного обучения.

Функционал Moodle:

- поддержка основных инструментов, типичных для такого типа сред до;
- публикация ресурсов в любых форматах и управление доступом к ним;
- банк заданий и гибкая система тестирования;
- связь через форумы и рассылки;
- сдача индивидуальных заданий в различных форматах (файл, несколько файлов, задание вне сайта, заполнение анкеты, открытый вопрос в текстах, глоссарии);
- опросы;
- дополнительные инструменты повышающие удобство и качество обучения;
- система оценок с настройками (промежуточные и итоговые оценки);
- анкеты для сбора и баз данных с возможностью подтверждения и оценивания учителем;

Данная система Moodle используется университетами, школами, компаниями и независимыми преподавателями в двухстах странах мира. Существует инсталляции, превышающие миллион пользователей.

Преимуществом по сравнению с другими СДО является открытый доступ к коду с возможностью модификации системы в зависимости от требований, предъявляемых к будущему курсу.

### 1.2. Система Дистанционного Обучения eLearning Server.

ELearning Server – это программное обеспечение, при помощи которого можно создавать учебные центры ДО в сети, выполняющие большое количество задач по организации дистанционного учебного процесса:

- создание учебных курсов для учащихся и преподавателей, ведение личных дел, регистрация;
- составление электронных ведомостей успеваемости (для преподавателей), электронных зачетов (для студентов или учащихся);
- создание и публикация тестов, упражнений, а также публикация учебных материалов;
- предоставление административных и пользовательских сервисов;
- формирование занятий и синхронизированное расписание по времени между участниками учебного процесса.

Возможность детальной проработки курса и учебного процесса посредством определения «роли» через права eLearning Server, будь то преподаватель или студент. Роль – это типовой набор функций и разрешений, предоставляемый участнику для выполнения его задач. По умолчанию система имеет четыре базовые роли:

- 1) Преподаватель.
- 2) Администратор.
- 3) Слушатель.

4) Учебная администрация.

Эти роли, за исключением администратора, имеют аналоги в традиционном очном обучении. Курс состоит из ссылок на учебные материалы в виде библиотеки учебных материалов. Расписание на сервере оформляется упорядоченно по времени. На сервере имеются методички, задания или тренинги, дополнительные средства общения.

Преимуществом по сравнению с другими СДО является легкая проработка учебного процесса, без дополнительных навыков программирования. Однако данная система не подлежит модификации и является платной.

1.3. Система Дистанционного Обучения REDCLASS Learning.

REDCLASS Learning – комплекс программно-аппаратных средств, методик обучения, учебных материалов, которые позволяют повышать квалификацию, улучшать знания в любых отраслях человеческой деятельности.

REDCLASS обладает следующими возможностями:

- Среда позволяет формировать и проверять навыки работы учащихся с системами, обладающими оконным интерфейсом. К примеру, может эмулировать работу Paint или MS Word для обучения студентов этой системы. Упражнения для среды создаются в конструкторе упражнений.

- Слушатели имеют возможность работать в виртуальных лабораториях с реальными (не эмулированными) программно-аппаратными комплексами в удаленном режиме.

- Мультимедийный контент на рабочем месте слушателя – электронный учебник. Курсы разрабатываются в автономной среде, называемой «Авторской системой».

- Для контроля успеваемости учащихся, есть система тестирования, которая позволяет осуществлять входной, выходной и промежуточный контроль знаний.

Преимуществом является возможность моделирования виртуальных лабораторий в режиме реального времени, в зависимости от потребностей учащегося. Однако данная система не подлежит модификации и является платной.

1.4. Система дистанционного обучения WebTutor.

Система позволяет организовать планирование, проведение и анализ результатов обучения пользователей (студентов, сотрудников компаний, клиентов) с помощью учебных курсов, а так же создает возможность общения, обмена информацией между учениками, преподавателями, экспертами и администраторами системы.

WebTutor обладает следующими возможностями:

- управление учебным процессом – назначение курсов (групповое, автоматическое по описанным в система правилам, индивидуальное, по итогам тестирования или другим оценкам). завершение курса обучения, рассылка уведомлений. есть механизм заявок;

- возможность работы с базой данных электронных курсов – курсы, соответствующие международным стандартам, изменение их параметров и управление каталогом курсов, права доступа к ним;

- для управления процессом обучения существует интерфейс выставления оценок на учебном портале для преподавателя и эксперта;

- для студента – интерфейс изучения электронных курсов и общение.

- есть возможность обмена информацией, между обучаемым, преподавателем и экспертом (чат, блог, форум и т.д.).

Дополнительные функциональные возможности:

- контроль учебного процесса руководителем – отчеты назначение курсов, уведомления, управления заявками;

- построение комбинированных (очных и дистанционных) программ;

- поддерживается режим реального времени – виртуальный класс;

- встроенный редактор курсов.

Система имеет базовый редактор электронных курсов, который дает возможность создавать курсы с произвольным количеством разделов и документов, содержащих тексты, иллюстрации, ссылки на другие документы и т.д.

#### 1.5. СДО Competentum.Magister.

Competentum. Magister – система для управления обучением, предназначена для средних, средних специальных и высших учебных заведений. Реализует электронные технологии обучения (e-learning) в очной, в заочной и дистанционной формах обучения.

Система управления обучением обеспечивает:

- календарное планирование.
- создание и воспроизведение учебных материалов.
- автоматизированный процесс проверки полученных знаний.
- анализ эффективности учебного процесса.
- информационные банки данных и знаний учебного заведения.
- дополнительные возможности:
- система поддерживает следующие типы пользователей:
- администратор системы.
- преподаватель.
- представитель администрации учебного заведения.
- студент.
- куратор.

#### 2. Критерии сравнительного анализа программных средств поддержки ДО

Критерием сравнительного анализа для систем дистанционного образования является соответствие стандарту SCORM. Sharable Content Object Reference Model (SCORM) – специальный сборник стандартов, разработанный для СДО.

В нем содержатся требования ко всей системе дистанционного обучения и к организации учебного процесса. Данный стандарт позволяет обеспечить совместимость компонентов и возможность их многократного использования: учебный материал доступен в отдельных небольших блоках, которые могут включаться в разные учебные курсы и использоваться системой дистанционного обучения независимо от того, кем, где и с помощью каких средств они были созданы. С помощью этого учебные объекты могут быть использованы в различных системах электронного дистанционного образования.

Выбраны критерии: тип архитектуры, тип лицензии, адаптация и конфигурирование, ориентировочная стоимость, самостоятельная регистрация учащихся, наличие модулей, функции формирования отчетов, поддержка электронного портфеля учащегося, наличие журнала посещений слушателей и их контроль, наличие зачетной книжки, средство общения для учащихся (чат, форум), поддержка проведения интернет конференций, количество обучающихся студентов в системе, сайт сдо в сети интернет (url), поддержка работы различных вузов, институтов, факультетов в составе вуза, нескольких филиалов.

#### 3. Сравнительный анализ систем ДО

Были рассмотрены пять видов программного обеспечения для дистанционного обучения, которые соответствуют стандарту SCORM:

- REDCLASS Learning,
- WebTutor,
- Competentum. Magister,
- eLearning Server,
- Moodle.

Сравнительный анализ СДО

Система Критерий	СДО WebTutor	СДО Competentum.Magister	СДО eLearning Server	СДО REDCLASS Learning	СДО Moodle
Лицензия	Коммерческая	Коммерческая	Коммерческая	Коммерческая	Бесплатная
Тип архитектуры	Закрытый	Закрытый	Закрытый	Закрытый	Открытый
Ориентировочная стоимость	Модуль «Дистанционное обучение» - от 150 тыс. руб.	Зависит от кол. пользователей в системе. Для 500 пользователей = 120 тыс. руб.	145 тыс. руб. на 1 сервер	100 тыс. на 1 сервер.	×
Адаптация и конфигурирование	Модуль «Управление порталом» – 50 тыс. руб.	Включено в стоимость	Включено в стоимость	Включено в стоимость	Самостоятельно
Наличие модулей	Администратор, слушатель, остальные модули дополнительно покупаются и интегрируются	Администратор, слушатель, тьютор	Студенты, преподаватели, учебная администрация, техническая администрация	Менеджер, администратор, пользователь	Администратор, слушатель, преподаватель
Регистрация слушателей через сайт СДО	×	×	✓	✓	✓
Наличие журнала и контроль пользователей	✓	✓	✓	✓	✓
Управление данными слушателей по курсам	×	×	✓	✓	✓
Формирование отчетов	✓	✓	✓	✓	✓
Редактирование контента сайта	✓	×	✓	✓	×
Система Критерий	СДО WebTutor	СДО Competentum.Magister	СДО eLearning Server	СДО REDCLASS Learning	СДО Moodle
Возможность просмотра расписания контрольных заданий	✓	×	✓	✓	✓
Зачетная книжка слушателя	✓	×	✓	✓	×

Система Критерий	СДО WebTutor	СДО Competentum.Ma gister	СДО eLearning Server	СДО REDCLASS Learning	СДО Moodle
Электронное общение слушателей	✓	✓	✓	✓	✓
Классная доска	✓	×	✓	✓	×
Сервис проведения Интернет- конференций	×	✓	✓	×	✓
Разработка лекций, тестов, контрольных заданий	✓	✓	✓	✓	✓
Количество обучающихся пользователе й	Не ограничено	Не ограничено только в версии enterprise	Не ограничено	Не ограничено	Не ограничено
Изменение базового дизайна системы	×	✓	×	✓	✓
Межвузовская система – поддержка работы разных вузов.	×	×	✓	✓	×
URL	<a href="http://www.websoft.ru">http://www.webs oft.ru</a>	<a href="http://competentum.ru">http://competentu m.ru</a>	<a href="http://www.learnware.ru">http://www.learnw are.ru</a>	<a href="http://www.redcenter.ru">http://www.redce nter.ru</a>	<a href="http://moodle.org/">http://moodle .org/</a>

Рассмотренные выше системы имеют набор инструментов, при помощи которых осуществляется управление учебным процессом. Делая выводы из сравнительной характеристики СДО, отметим главные плюсы коммерческой СДО «eLearning Server» и бесплатной СДО «Moodle»:

- Регистрация через сайт студентов самостоятельно.
- Большое количество ролей.
- В рамках единой СДО поддержка работы вузов и его филиалов.
- Контроль данными курсов.
- Возможность проведения интернет конференций.
- Поддержка электронного портфеля студента.

Заключение

Преимуществом системы «Moodle» является не только бесплатная лицензия, но и открытая архитектура, благодаря которой, существует возможность модификации самой структуры СДО, в зависимости от заявленных требований, а также контроль учащихся, удобная система проверки изученных материалов и возможность проведения общих лекций с неограниченным количеством учащихся.

Преимуществом системы «eLearning Server» является поддержка межвузовской системы: факультетов, филиалов, ВУЗов и т.д., а также техническая поддержка от производителя, но система является платной и без возможности изменения архитектуры .

Системы «REDCLASS», «Competentum.Magister», «WebTutor» имеют наименьший набор функциональных возможностей и являются платными, без возможности изменения архитектуры, а также система «Competentum.Magister» накладывает ограничения на количество учащихся в большинстве версий.

Рассматривая системы «eLearning Server» и «Moodle» видно преимущества каждой из них, но возможность изменения архитектуры, является более значительным преимуществом, в сравнении с другими. В результате отбора была выбрана система «Moodle».

#### **Библиографический список:**

1. Андреев А.А. Введение в дистанционное обучение: учеб.-метод. пособие. М.: ВУ, 1997 г.
2. Системы дистанционного обучения в Интернете –[Электронный ресурс] URL: [http://www.i2r.ru/static/221/out\\_18244.shtml](http://www.i2r.ru/static/221/out_18244.shtml)
3. Advanced Distributed Learning. Sharable Content Object Reference Model (SCORM) 20/ Перевод с англ. Е.В. Кузьминой. М.: ФГУ ГНИИ ИТТ «Информика» – [Электронный ресурс]URL: [http://www.scorm.com/ru/?adgroupname=strict\\_scorm&campaignname=scorm&keyword=learning%20scorm&gclid=CO63\\_4Krm8wCFSLicgod4IkFmQ](http://www.scorm.com/ru/?adgroupname=strict_scorm&campaignname=scorm&keyword=learning%20scorm&gclid=CO63_4Krm8wCFSLicgod4IkFmQ)
4. Системы дистанционного обучения E-Learning Софт/ «Преимущества внедрения СДО для учреждений образования».
5. Сайт moodle – [Электронный ресурс]URL: <https://moodle.org/>
6. Сайт– [Электронный ресурс]URL: <http://dstudy.ru/?page=5b09fc48-7f08-42fd-adce-324d923bf727&item=99a10969-f05d-4c7d-8464-b334fd9f3ab1&type=page>
7. Сайт – [Электронный ресурс]URL: <http://ra-kurs.spb.ru/2/0/2/1/?id=13>
8. Wikipedia – [Электронный ресурс]URL: - <https://ru.wikipedia.org/wiki/SCORM>
9. Сайт moodle – [Электронный ресурс]URL: <http://moodlelearn.ru/course/view.php?id=18>
10. Сайт moodle – [Электронный ресурс]URL: <http://moodlelearn.ru/course/view.php?id=23>

**УДК 629.735.33**

**А. Ю. Миронов**

магистрант кафедры компьютерных технологий и программной инженерии

**М. В. Фаттахова**

канд. физ-мат наук, доц. – научный руководитель

### **АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ ПРОИЗВОДСТВОМ ПО ДЕЛАМ ОБ АДМИНИСТРАТИВНЫХ ПРАВОНАРУШЕНИЯХ НА РЕГИОНАЛЬНОМ УРОВНЕ**

Вступление в силу с середины 2002 г. Кодекса Российской Федерации об административных правонарушениях почти в 2 раза расширило число составов административных правонарушений, в том числе за счет учета правонарушений должностных и юридических лиц. Появилось и развивается региональное административное законодательство. Набирает силу процесс декриминализации уголовно-наказуемых деяний с отнесением последних в разряд административных правонарушений.

В настоящее время на территории Санкт-Петербурга и Ленинградской области ежегодно составляется более 2 млн административных протоколов. Их необходимо рассмотреть по подведомственности, принять по ним процессуальные решения, обеспечить исполнение административных наказаний.

С 2008 г на подразделения органов государственной власти, исполняющих административное законодательство, дополнительным бременем легла реализация постановления Правительства РФ от 29.12.2007 №995, которым на них было возложено осуществление бюджетных полномочий администраторов доходов федерального бюджета и

главных администраторов доходов бюджетов субъектов РФ и их муниципальных образований. Наряду со сборами за выдачу лицензий, разрешений и справок, уплачиваемых в виде предоплаты, административные штрафы составляют наиболее разнообразную и весомую часть поступлений в доход РФ, субъекта РФ и его муниципалитетов, которая уплачивается в качестве наказания за совершение более 60% административно наказуемых деяний.

И весь этот функционал исполнения административного законодательства осуществляется десятками служащих каждого уполномоченного органа государственной власти в «ручном» режиме, если не считать локальные средства автоматизации отдельных административных процедур, такие как автоматизированный учет «Административная практика» в МВД России, автоматизированная система «Администратор доходов» и т.п.)<sup>1</sup>.

Механизм государственного реагирования на административные правонарушения предусматривает участие и взаимодействие около 75 правоохранительных и исполнительных органов государственной власти.

В рамках сложившейся несовершенной системы исполнения административного законодательства многоэтапное производство по делам об административных правонарушениях осуществляется множеством участников обособленно и неэффективно. Взаимодействие между ними производится на основе бумажных почтовых отправок. Управление на системном уровне отсутствует.

В результате, часть административных протоколов остается без решения, административные наказания не исполняются в полном объеме. Только от несобираемости административных штрафов федеральный и региональные бюджеты теряют десятки процентов доходной части или сотни миллионов рублей в год.

Ведомственная разобщенность участников административного производства усилена с 2012 года устранением федеральных правоохранительных органов и исполнительных органов государственной власти от исполнения регионального административного законодательства. В настоящее время региональные законы об административных правонарушениях фактически не исполняются из-за отсутствия механизма реализации.

Накопленные проблемы исполнения административного законодательства придают особую остроту и значимость комплексной автоматизации административного производства на региональном уровне, который характеризуется сложившейся инфраструктурой информационного взаимодействия исполнительных органов государственной власти РФ и субъекта РФ.

Создание единой информационно-аналитической системы управления полным циклом производства по делам об административных правонарушениях на региональном уровне позволит погрузить многоэтапный процесс административного делопроизводства в единое пространство автоматизированных процедур региона. Она призвана поставить под сквозной контроль весь комплекс проблем исполнения административного законодательства: от неотвратимости административной ответственности до собираемости административных штрафов в доход бюджетов РФ и субъекта РФ.

Реализуемость данной системы базируется на теоретических основах автоматизации управления производства по делам об административных правонарушениях, заложенных в последние десятилетия специалистами разных областей знаний. В 80-90 г. прошлого века американской и советской научными школами проработаны теоретические основы и доказана перспективность автоматизированных систем управления в работах Ф. Чаки, В.А. Бесекерского, И.М. Макарова и др. В 2000-е гг. теоретическое обоснование информационного обеспечения административного управления получило в трудах Г.В. Атаманчука, Д.Н. Бахраха, Ю.А. Тихомирова и др. Прикладные проблемы производства по делам об административных

<sup>1</sup> Юридическая статистика: учебник / В.В. Лунеев; Институт государства и права РАН. 3-е изд., перераб. и доп. М.: Норма: ИНФРА-М, 2014. 448 с.

правонарушениях с позиций информационного права рассматривались в работах В.Т. Батычко, А.С. Дугенца, М.Я. Масленникова и др.

Реальное внедрение информатики в производство по делам об административных правонарушениях берет свое начало с издания приказа МВД России от 01.09.1997 №575, который в ряду неотложных мер поручил изучить возможность использования в МВД, ГУВД, УВД субъектов РФ компьютерной программы типа «Административная практика» для контроля за деятельностью по исполнению административного законодательства.

Наставление по формированию и ведению централизованных оперативно-справочных, розыскных, криминалистических учетов, утвержденное приказом МВД России от 12.07.2000 №752дсп, предложило оперативно-справочный учет алфавитных карточек формы 1-б на лицо, совершившее административное правонарушение, если установить его личность иным способом невозможно. Приказ МВД России от 09.07.2007 №612дсп дополнительно ввел криминалистический учет карт КАП на административное правонарушение, по которому возбуждено дело об административном правонарушении в отношении иностранного гражданина или ЛБГ, либо на принятое решение о депортации. Наконец, межведомственный приказ МВД России от 12.02.2014 №89дсп разрешил на региональном уровне в интегрированном банке данных (ИБД-Р) только в электронном виде учитывать административную практику в объеме учетной формы 1-б на гражданина РФ, совершившего административное правонарушение и привлекавшегося к административной ответственности, или в объеме карты КАП на административное правонарушение, по которому возбуждено дело об административном правонарушении в отношении иностранного гражданина или лица без гражданства, либо на принятое решение о депортации<sup>1</sup>.

Реквизитный состав учетной алфавитной карточки формы 1-б и карты КАП соответствует обязательным реквизитам из Перечня входящих в состав сегмента «Административная практика» подсистемы «Картотека» интегрированного банка данных федерального уровня (ИБД-Ф): ФИО, дата и место рождения, место жительства и регистрации, орган, дата составления и номер документа об административном правонарушении, квалификация административного правонарушения, орган и дата принятия решения, вид административного наказания, дополнительные сведения. При наличии сведений о совершении лицом административного правонарушения в состоянии наркотического опьянения информация о виде наркотического средства или психотропного вещества отражается в поле «Вид наркотического средства или психотропного вещества».

Порядок передачи и сбора информации по административной практике в электронные банки данных информационных центров, контроля за ее достоверностью определяется территориальным органом МВД России на региональном уровне и органом, предоставляющим сведения по делам об административных правонарушениях. Однако Наставлением применение формализованных учетных документов ориентировано на картотечный принцип формирования и ведения централизованных оперативно-справочных, розыскных, криминалистических учетов, заложенный еще приказом МВД России от 31.08.1993 №400 для информационного обеспечения оперативно-розыскной деятельности на базе технологии статистических наблюдений. Учетная карточка составляется органом (должностным лицом), возбуждившим дело об административном правонарушении, принявшим решение о назначении административного наказания, депортации или прекратившим производство по делу об административном правонарушении, и в течение двух рабочих дней ставится на учет в информационном центре территориального органа МВД России на региональном уровне, которому на обработку, контроль соответствия, сверку и регистрацию предоставляется еще три рабочих дня.<sup>2</sup>

<sup>1</sup> *Ищенко Е.П., Топорков А.А.* Криминалистика: учебник/ под ред. Е.П.Ищенко. 3-е изд., испр. и доп. М.: Юридическая фирма «Контракт», Волтерс Клувер, 2011. 512с.

<sup>2</sup> *Потоцкий С.Н.* Формирование учетной информации по делам об административных правонарушениях. // Экономика и право. XXI в: Официальное издание Института международного права и экономики им. А.С.Грибоедова. М: 2012. №1 (январь-март).



В процессе заполнения, доставки, распознавания и ввода первичных документов в базу данных последовательно накапливаются ошибки передаточных звеньев цепи из должностных лиц, сотрудников, операторов, суммирование которых неминуемо ведет к неполноте и недостоверности базы данных и формируемых на ее базе заключений. Усилия на сверки и возвраты учетных документов для внесения исправлений способны лишь в гиперболической зависимости минимизировать негативный человеческий фактор. А неуклонное смещение основной сферы использования картотечных учетов от статистических и аналитических наблюдений в форме отчетности через информационное обеспечение оперативно-разыскной деятельности к документальному подтверждению статуса конкретного лица или деяния делает требования максимальной полноты и достоверности необходимыми и определяющими характеристиками электронных банков данных.

В рамках внедряемой в настоящее время Единой системы информационно-аналитического обеспечения деятельности (ИСОД) МВД России, включившей в свой состав после модернизации интегрированные банки данных федерального (ИБД-Ф) и регионального (ИБД-Р) уровней, учет «Административная практика» продолжает носить узко ведомственный и профилактический характер, учитывая лицо, на которое сотрудниками МВД России составлен административный протокол, с целью дополнения поисковой информации. При этом, фиксация правонарушителя по факту составления протокола не отслеживает этапы административного процесса, в том числе исполнения наказания.

Таким образом, назрели объективные условия, когда управляющим звеном исполнения административного законодательства, объединяющим разрозненные силы и средства может и должна стать автоматизированная система управления производством по делам об административных правонарушениях на региональном уровне. Она на научной основе, опираясь на сложившуюся информационную инфраструктуру в субъекте РФ, обеспечит взаимодействие между участниками производства, контроль своевременности и полноты этапов производства.

Рассматривая многоэтапный процесс производства по делам об административных правонарушениях, необходимо детально изучить и объединить в единый управляемый механизм сложные административные процедуры: возбуждение производства по административному делу; передачу административного дела в иной орган по подведомственности; рассмотрение административного дела; принятие решения по административному делу; исполнение административного наказания, в том числе путем исполнительного производства; администрирование доходов бюджетов РФ и субъектов РФ от административных штрафов; оказание государственных услуг населению в части исполнения и учета административных наказаний; учет привлечения к административной ответственности.

Концептуальная модель управления производством по делам об административных правонарушениях представляет собой классическую систему управления с обратной связью.

Процесс производства по делам об административных правонарушениях включает этапы документирования административного правонарушения (доставления, задержания, составления протокола об административном правонарушении), передачу дела об административном правонарушении по подведомственности, принятия решения по делу об административном правонарушении, и назначение административного наказания, контроль исполнения административного наказания, исполнительное производство судебных приставов по неисполненным наказаниям. Входными воздействиями на систему административного производства являются признаки административного правонарушения. Выходным сигналом является время и результат исполнения каждого этапа процесса производства. Обратной связью является подсистема контроля, своевременности и полноты исполнения этапа процесса производства. Управляющим сигналом подсистемы обратной связи является уведомление о необходимости завершения этапа процесса производства или команда на реализацию последующего этапа процесса производства.

Критериями эффективности системы автоматизированного управления являются минимизация времени осуществления административных процедур, стремление к максимальному количеству своевременно оконченных дел, полноте исполнения административных наказаний (в том числе собираемости административных штрафов).

Рассматриваемая система автоматизированного управления обеспечивает единый контроль исполнения федерального и административного законодательства внутри субъекта РФ. Её информационная модель строится с учетом особенностей развития информационной и технологической инфраструктуры в регионе.

Использование веб-технологий при реализации автоматизированной системы управления производством по делам об административных правонарушениях даст возможность:

- сопроводить административный процесс формированием процессуальных документов;
- увязать контроль производства по делам об административных правонарушениях и администрирования доходов с иными учетами исполнительных органов государственной власти региона;
- обеспечить ведение административного делопроизводства в региональном пространстве идентификаторов, словарей и классификаторов;
- осуществлять накопление и обмен информацией с реквизитами финансовой и казначейской систем учета поступлений в бюджеты;
- организовать интернет-службу, позволяющую гражданину по фамилии, имени, отчеству и номеру протокола (постановления) по делу об административном правонарушении получить достоверную информацию о долге по оплате административного штрафа и заполненную актуальными реквизитами квитанцию формы ПД-4 для его оплаты;
- участвовать в предоставлении государственных услуг.

Предлагаемая система автоматизированного управления производством по делам об административных правонарушениях позволит обеспечить руководство субъекта РФ, правоохранительных и исполнительных органов государственной власти региона аналитическими материалами о состоянии исполнения административного законодательства, организовать эффективную расстановку сил и средств, сопровождающих производство о делах об административных правонарушениях.

### **Библиографический список**

1. Юридическая статистика: учебник / В.В. Лунеев; Институт государства и права РАН. 3-е изд., перераб. и доп. М.: Норма: ИНФРА-М, 2014. 448 с.
2. Ищенко Е.П., Топорков А.А. Криминалистика: учебник/ под ред. Е.П.Ищенко. 3-е изд., испр. и доп. М.: Юридическая фирма «Контракт», Волтерс Клувер, 2011. 512с.
3. Потоцкий С.Н. Формирование учетной информации по делам об административных правонарушениях. // Экономика и право. XXI в: Официальное издание Института международного права и экономики им. А.С.Грибоедова. М: 2012. №1 (январь-март).

УДК 004.42

**Д. Е. Нарышкин**

студент кафедры компьютерных технологий и программной инженерии

**М. Д. Поляк**

ст. преподаватель – научный руководитель

## **УНИФИКАЦИЯ АДРЕСОВ МНОГОКВАРТИРНЫХ ДОМОВ В САНКТ-ПЕТЕРБУРГЕ С ИСПОЛЬЗОВАНИЕМ ФИАС**

На территории РФ расположено большое количество многоквартирных домов. Статья 44 Жилищного кодекса РФ устанавливает, что главным органом управления многоквартирными домами является общее собрание всех собственников помещений в многоквартирном доме. Собственники помещений (далее – собственники) обязаны ежегодно проводить общее собрание всех собственников в многоквартирном доме, порядок проведения которого установлен жилищным кодексом РФ. На собрании проводится голосование по некоторым актуальным вопросам, например, выборе способа управления многоквартирным домом, принятии решений о текущем ремонте общего имущества в многоквартирном доме, выборе администратора и т. д.

В соответствии с жилищным кодексом, голосование собственников помещений может быть проведено либо в традиционной бумажной форме, либо с использованием системы электронного голосования [1]. В соответствии с этим, перед программистом была поставлена задача: необходимо разработать систему (сайт, портал) для сети Интернет, с помощью которой можно было бы осуществлять процесс электронного голосования, а также управлять (т. е. создавать, удалять или изменять) информацией о собраниях многоквартирных домах, где соответствующие собрания проводятся.

Для того чтобы собственник мог голосовать с использованием данного сайта, он должен пройти процедуры регистрации и авторизации в системе. Поскольку каждый человек имеет возможность голосования по вопросам, поставленным на голосование на собрании только в собственном доме, необходимо при регистрации указывать собственный адрес проживания. В связи с этим появляется проблема, связанная с тем, что пользователи записывают адреса по-разному. Например, указывая в адресе «к.» один человек может иметь в виду «корпус дома», другой - «квартиру». В системе электронного голосования такая неопределенность неприемлема. Необходимо каким-нибудь образом сделать так, чтобы каждый адрес в системе записывался бы единственным заранее определенным образом, тогда не будет никаких неопределенностей, связанных с записью адресов пользователей.

Для решения этой проблемы целесообразно использовать специализированные адресные системы. На сегодняшний день имеются две системы – КЛАДР (классификатор адресов России) и ФИАС (федеральная информационная адресная система), разработанные и поддерживаемые Федеральной налоговой службой (ФНС).

КЛАДР – классификатор адресов России, который был введен в действие в 2005 г. Данный классификатор создавался как справочник, содержащий все адреса РФ. Вся адресная информация находится в открытом доступе и хранится в специализированном формате, разработанном специально для работы с КЛАДР. Однако, в течение достаточно большого промежутка времени этот справочник был очень странной базой адресов, поскольку содержал множество неточностей и ошибок в записи адресов. Поэтому, спустя некоторое время, специалисты ФНС приступили к разработке нового адресного справочника.

Новый адресный справочник получил название ФИАС – федеральная информационная адресная система. 1 июля 2014 года вступил в силу закон, согласно которому все государственные и муниципальные структуры должны опираться на этот справочник как на единственно верную базу адресов [2]. Так же, как и у КЛАДР, вся адресная информация в ФИАС

находится в открытом доступе. Первое, на что стоит обратить внимание при работе с ФИАС, информации гораздо больше, нежели в КЛАДР. Однако полезной информации прибавилось не так много. С 2011 г. КЛАДР – формат выгрузки данных из ФИАС в несколько урезанном виде, поэтому для обычного пользователя КЛАДР и ФИАС не отличаются ничем.

ФИАС доступен в трех видах: формат КЛАДР, DBF и XML. ФИАС в формате DBF занимает 9 Гб, в формате XML – 14 Гб (например, только с названиями улиц в ФИАС примерно 2,5 млн. записей). ФИАС в формате КЛАДР по наполнению представляет собой то же самое, что и сам КЛАДР, за редким исключением, и занимает он те же самые 330 Мб. Структура адресной информации в форматах DBF и XML представляет собой совокупность n-го количества файлов соответствующего формата, каждый из которых содержит часть адресной информации, например, номера домов конкретного региона РФ, названия всех улиц РФ и т. д. Также можно найти информацию о почтовых индексах домов, дате начала и окончания действия записи и многое другое. У каждого объекта есть идентификаторы, что позволяет ссылаться на записи из других файлов. Например, чтобы узнать, на какой улице находится дом, необходимо в файле с улицами найти запись, у которой идентификатор улицы совпадает с идентификатором улицы в файле с номерами домов у необходимого дома и т. д. Пример записи ФИАС в формате XML:

```
<?xml version="1.0" encoding="utf-8"?>
<AddressObjects>
<Object AOID="20e60730-ff9f-442a-8d7b-a3acd20cb87c" ...
FORMALNAME="Адыгея" OFFNAME="Адыгея" SHORTNAME="Респ" ...
REGIONCODE="01" AREACODE="000" ...
STARTDATE="1900-01-01" ENDDATE="2079-06-06"
UPDATEDATE="2015-09-15" />
```

Несмотря на то, что у ФИАС больше плюсов, чем у КЛАДР, до сих пор неточностей в этой системе довольно много. Не решена проблема с наличием дублей домов, наличием несуществующих домов, неправильной записью некоторых номеров домов и т. д. Каждый разработчик вынужден решать эти вопросы самостоятельно. Однако, несмотря на эти недостатки, на сегодняшний день это лучшая адресная база, поэтому ее необходимо использовать.

Для применения описанных выше адресных справочников в системе электронного голосования можно воспользоваться адресной информацией в формате DBF или XML. Разрабатываемая система для голосования использует базу данных MySQL, поэтому каждый файл, содержащий определенную адресную информацию, можно при помощи специальных программ или менеджеров баз данных преобразовать таким образом, чтобы на выходе получились либо готовые таблицы какой-либо базы данных, либо запросы на языке SQL для создания таблиц в БД и вставки необходимых данных в них. Поскольку разрабатываемую систему электронного голосования предполагается использовать только в Санкт-Петербурге, необходимо в БД сохранить названия улиц и номера домов только города Санкт-Петербурга. Данные в ФИАС хранятся по регионам, поэтому после создания SQL-таблиц было необходимо удалить их из записи, не относящиеся к 78-му региону.

Наконец, необходимо реализовать способ, который используется на многих веб-сайтах – подсказки при вводе адреса. Его принцип заключается в том, что пользователю при регистрации необходимо ввести всего лишь первые 2-3 буквы или цифры от названия своей улицы или номера дома, после чего на веб-сервер отправится запрос, ответом на который будет список возможных улиц или номеров домов. Он отобразится как подсказка к полю вводу (рисунок). Пользователь сможет ввести значение только из тех вариантов, которые ему будут предложены.

Улица \*

Сла
пл. Морской Славы
пр-кт. Славы
наб. Реки Славянки
наб. Славянская
ул. Славянская (Усть-Сл...
ул. Славянская (Лахта)
проезд. Славянский
ул. Славянская
дор. Славянская
тер. Московская Славянка

Подсказка при вводе названия улицы

Также для работы с ФИАС существует специальный сервис «КЛАДР в облаке», разработанный архангельской IT-компанией Primerix. Обмен данными происходит в формате JSON (специальный текстовый формат для обмена данными). Для использования данного сервиса команда программистов разработала специальные плагины для некоторых веб-языков программирования, которые упрощают взаимодействие с БД ФИАС. Данный проект является open source проектом, т. е. все исходные коды доступны абсолютно бесплатно. Можно работать с данной системой на коммерческой или на бесплатной основе. Для бесплатной версии, в отличие от платной, не гарантируется полноценная работа сервиса, ограничено количество запросов в день (примерно 5000 – 10000 адресов), а также запрещено использовать сервис в коммерческих проектах. Несмотря на эти недостатки, бесплатная версия сервиса работает уже больше года, отрицательных отзывов пользователей не было. Для работы с сервисом необходимо загрузить код соответствующего плагина и написать скрипт с параметрами настройки для конкретных полей ввода, чтобы определить, как именно должно происходить автодополнение адреса. Основным плюсом сервиса является то, что он решает все задачи, связанные с реализацией вывода подсказок при вводе, а программисту остается только определить параметры вывода адресной информации.

Существование ФИАС значительно упрощает разработку многих систем, поскольку довольно часто появляется задача унификации адресов. Несмотря на ошибки, присутствующие в ФИАС, на сегодняшний день это лучшая адресная база, которая не имеет аналогов. Важной особенностью ФИАС является то, что она хранит информацию в различных форматах, в том числе и в старом формате КЛАДР, что дает больше свободы при работе с этой системой. С появлением сервиса «КЛАДР в облаке» работа с ФИАС сильно упростилась, поскольку разработчикам больше не требуется загружать полную информацию из ФИАС, достаточно подключить специальный плагин, который сам будет обращаться к ФИАС для вывода необходимой адресной информации.

### Библиографический список

1. "Жилищный кодекс Российской Федерации" от 29.12.2004 N 188-ФЗ (ред. от 31.01.2016). Статья 47.1.

2. Федеральный закон № 443-ФЗ "О федеральной информационной адресной системе и о внесении изменений в Федеральный закон "Об общих принципах организации местного самоуправления в Российской Федерации" от 28.12.2013.

**УДК 629.735.33**

**Д. О. Пазенко**

магистрант кафедры вычислительных систем и сетей

**А. В. Никитин**

канд. техн. наук, доц. – научный руководитель

### **РЕКОНСТРУКЦИЯ ТРЕХМЕРНОЙ СЦЕНЫ НА ОСНОВЕ ВИДЕОПОТОКА**

Область применения трехмерной графики уже давно вышла за рамки обычных развлечений. Точные и высоко детализированные модели используются в промышленности для наглядной демонстрации изделия вне зависимости от его формы и габаритов. В архитектурной и музейной отраслях трехмерные модели позволяют воссоздать внешний вид предметов и памятников, которые сохранились лишь частично. Виртуальные тренажеры позволяют выработать у персонала навыки работы с устройствами с помощью компьютера и интерактивной трехмерной модели. Методы получения трехмерных моделей могут быть очень трудоемкими либо недоступными для рядового пользователя. Например, процесс создания модели объекта в трехмерном редакторе может занимать часы или даже дни; трехмерный сканер позволяет получить модель небольшого объекта, но стоит довольно дорого. Предлагаемое в данной работе решение позволит получить черновую трехмерную модель объекта или сцены.

Цель работы – анализ и разработка модели, алгоритма и программного решения, позволяющего получить трехмерную модель объекта или сцены, записанного на видеокамеру. В процессе анализа были выделены следующие задачи:

- получение последовательности изображений из видеофайла;
- анализ пар изображений;
- построение траектории движения камеры;
- получение облака трехмерных точек сцены;
- фильтрация трехмерных точек;
- построение трехмерного каркаса сцены на основе полученного облака (без нанесения текстуры);
- экспорт полученной модели в один из форматов хранения трехмерных моделей, либо вывод полученной модели на экран средствами программы.

Средой разработки была выбрана Qt(используется язык C++). В программной реализации используется библиотека OpenCV. Эта библиотека предоставляет доступ к алгоритмам компьютерного зрения и цифровой обработки изображений.

Обзор известных решений:

Фотограмметрия[1]. Основные области применения: топография, архитектура и т.д. Позволяет дистанционно определить форму и габариты обозреваемого объекта, сопоставляя разноракурсные снимки. Чаще всего применяется аэрофотосъемка. Необходимая аппаратура, недоступна рядовому пользователю, процесс съемки может оказаться очень сложным. Область применения выходит за рамки данной работы, высокая сложность и точность алгоритмов теряет смысл при построении черновой модели сцены. Поэтому из данной области была взята основная идея: получение трехмерных координат точек объекта на основе разноракурсных изображений.

В работе[2] предлагается использовать последовательность разноракурсных изображений для получения облака трехмерных точек сцены. Особенностью предложенного решения является определение траектории движения камеры на основе данной последовательности и расчет координат трехмерных точек в единой системе координат. Полученное облако аппроксимируется трехмерными примитивами (в рассмотренных примерах – цилиндром и параллелепипедом). Данное решение не позволяет построить трехмерную модель сцены. Из рассмотренной работы взят метод получения траектории движения камеры на основе последовательности кадров.

В работе [3] рассмотрен способ получения облака трехмерных точек на основе стереопар изображений. Для получения стереопар предлагается использовать две откалиброванные камеры.

Ни один рассмотренный аналог не выполняет поставленную в данной работе цель.

Алгоритм получения облака трехмерных точек сцены основан на использовании «Особых точек» изображения[4] – точек, в окрестности которых происходит резкий перепад градиента по обеим осям изображения (то есть угловые точки). Область вокруг найденной особой точки разбивается на блоки, в каждом из которых вычисляется преобладающее значение градиента. Набор данных градиентов является идентификатором особой точки и хранится в специальной структуре – дескрипторе (рис. 1).

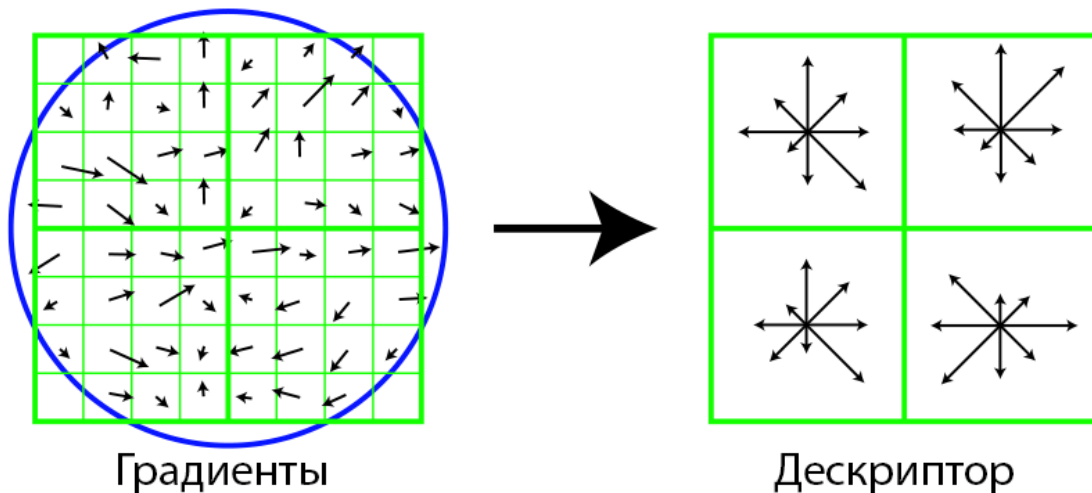


Рис. 1. Дескриптор особой точки

Поиск особых точек и вычисление их дескрипторов выполняется такими алгоритмами как SIFT, SURF[5] и т.д. Videopоследовательность раскладывается на отдельные кадры. На каждом выполняется поиск ключевых точек. Далее производится поиск одинаковых точек на соседних кадрах (текущем и предыдущем). После нахождения совпадений вычисляются параметры перспективной трансформации второго кадра относительно первого, по которым вычисляется изменение положения камеры. На основе этих данных выполняется построение траектории движения камеры. Следующий этап - получение карты глубины кадра. С ее помощью вычисляются координаты особых точек в общем трехмерном пространстве (зная текущее положение и направление камеры). На выходе будет получено облако трехмерных точек сцены.

Заключительный этап – фильтрация точек (удаление ошибочных точек, сглаживание) и построение полигональной сетки.

Выводы: на данный момент рассмотрены известные решения, разработан общий алгоритм реконструкции сцены, подготовлена среда разработки и необходимые библиотеки, получены параметры перспективной трансформации кадров и реализована процедура получения траектории камеры.

**Библиографический список**

1. Описание дисциплины фотограмметрия [Электронный ресурс]. – Режим доступа: <http://dic.academic.ru/dic.nsf/bse/145613/Фотограмметрия>, свободный, дата обращения – 09.05.2016.
2. Конушин А. С. «Алгоритмы построения трехмерных компьютерных моделей реальных объектов для систем виртуальной реальности» [Электронный ресурс]. – Режим доступа: <http://keldysh.ru/council/1/konushin.pdf>, свободный, дата обращения – 09.05.2016.
3. Кургалин С. Д., Крыловецкий А. А. и Божок А. О. Методы и алгоритмы построения трехмерных моделей в системах компьютерного зрения [Электронный ресурс]. – Режим доступа: <http://tm.ifmo.ru/tm2014/src/090b.pdf>, свободный, дата обращения – 09.05.2016.
4. David G. Lowe «Distinctive Image Features from Scale-Invariant Keypoints» [Электронный ресурс]. – Режим доступа: <http://people.eecs.berkeley.edu/~malik/cs294/lowe-ijcv04.pdf>, свободный, дата обращения – 29.05.2016.
5. Обзор дескрипторов ключевых точек [Электронный ресурс]. – Режим доступа: <http://www.intuit.ru/studies/courses/10621/1105/lecture/17983?page=2>, свободный, дата обращения – 29.05.2016.

УДК 681.518

**Д. В. Печенин**

студент кафедры вычислительных систем и сетей

**Л. А. Мироновский**

д-р техн. наук, проф.– научный руководитель

**КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ ОДНОГО КЛАССА ЗАДАЧ  
ТЕРМИНАЛЬНОГО УПРАВЛЕНИЯ**

Задача терминального управления – один из видов задач теории оптимального управления. В ней требуется рассчитать управляющий сигнал  $u(t)$ , обеспечивающий перевод динамического объекта управления из заданного начального состояния в заданное конечное состояние за фиксированное время  $T$ . Сигнал  $u(t)$  должен минимизировать функционал вида

$$I_0^T = \int_0^T f(X(t), \dot{X}(t), u(t)) dt \rightarrow \min.$$

При этом возникает вопрос учета различных возмущающих воздействий и оценка их влияния на качество управления. Целью данной работы было проведение компьютерного моделирования для исследования указанного влияния в конкретной задаче терминального управления, известной как задача о двух интеграторах.

Структурная схема объекта управления приведена на рис.1. Она содержит два интегратора с известными начальными и конечными условиями  $X(0) = X_0$  и  $X(T) = X_T$ .

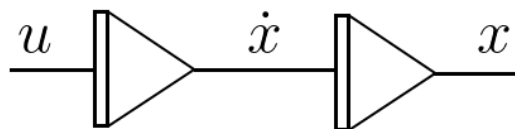


Рис.1. Структурная схема задачи о двух интеграторах

Одной из интерпретаций задачи о двух интеграторах служит задача об управлении материальным телом, движущимся вдоль прямой (Рис.2)



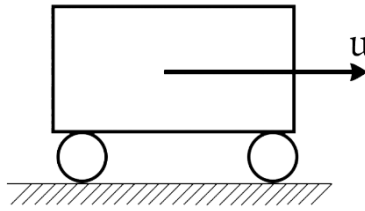


Рис. 2. Механическая интерпретация задачи о двух интеграторах

Компьютерное моделирование указанной задачи выполнялось в пакетах Matlab и Wolfram Mathematica. Выбор данных пакетов основан на том, что Matlab предоставляет широкие возможности для численных вычислений, а Wolfram Mathematica является одним из лидеров среди математических пакетов для символьных вычислений.

При моделировании использовалось описание объекта управления в пространстве состояний:

$$\dot{X}(t) = AX(t) + bu(t); \quad Y(t) = cX(t),$$

где  $A$  – матрица динамики объекта;  $b$  – матрица входов;  $c$  – матрица выходов.

Матрицы  $A$ ,  $b$ ,  $c$  имели следующий вид:

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, c = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Также было задано время  $T$ , за которое необходимо осуществить перевод объекта из начального состояния  $X_0$  конечное состояние ( $X_T$ );

$$X_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}; X_T = \begin{bmatrix} 9 \\ 1 \end{bmatrix}; T = 15.$$

В качестве критерия требовалось минимизировать энергию управляющего сигнала:

$$E = \int_0^T u^2 dt \rightarrow \min.$$

Закон оптимального управления для заданного критерия оптимизации носит линейный характер [1, 2]:

$$u(t) = \frac{-2}{375}(t - 20).$$

Формулы для скорости и координаты имеют вид:

$$x(t) = -\frac{(-60 + t)t^2}{1125},$$

$$v(t) = -\frac{1}{375}(-40 + t)t.$$

Соответствующие графики оптимального управления и скорости приведены на рис. 3, а график координаты – на рис. 4.

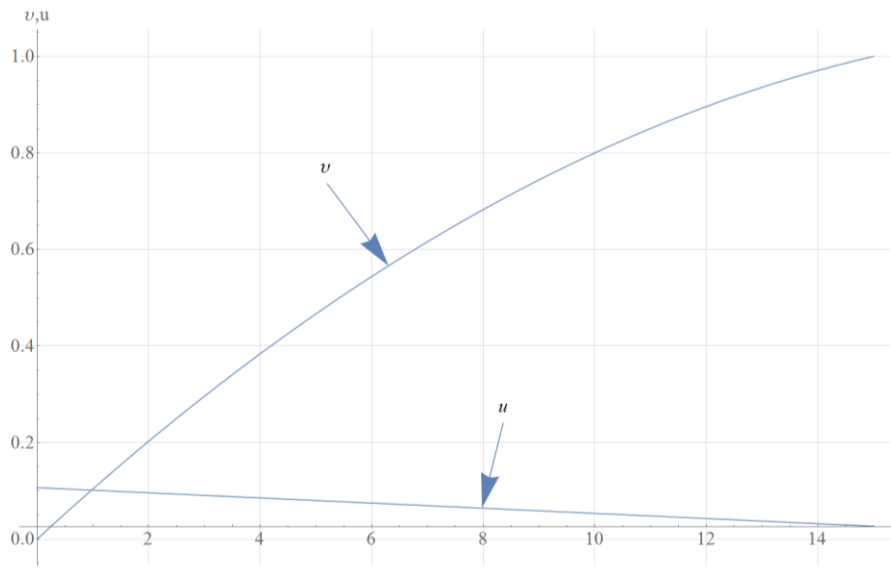


Рис. 3. Графики управляющего сигнала и скорости

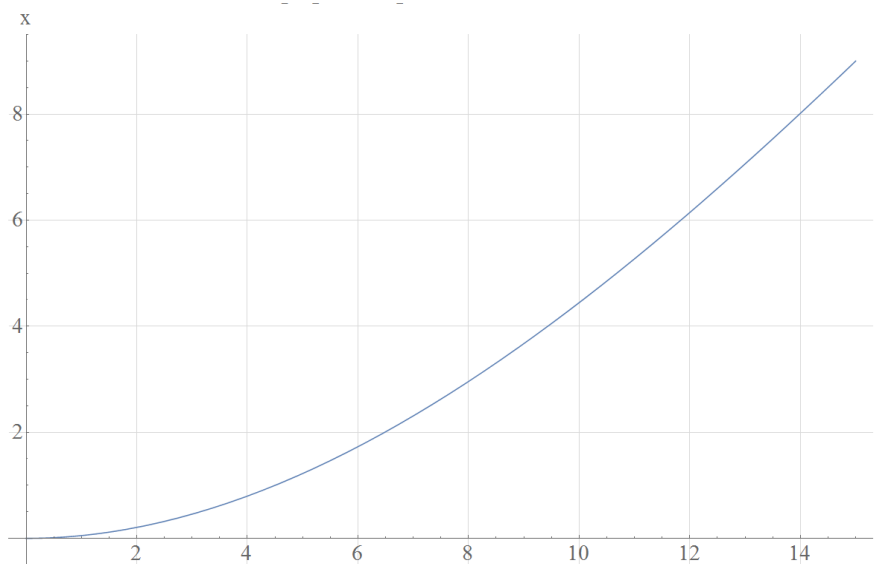


Рис. 4. График изменения координаты

Далее была исследована реакция системы на возмущающие воздействия. При этом использовалось как численное моделирование, так и символическое решение дифференциальных уравнений. В качестве возмущающих воздействий были рассмотрены:

1. Сопротивление воздуха (параметр  $k > 0$ ):

$$\begin{cases} \dot{x} = v, \\ \dot{v} = u - kx. \end{cases}$$

2. Снижение мощности управления  $u$  (параметр  $m < 1$ ):

$$\begin{cases} \dot{x} = v, \\ \dot{v} = mu. \end{cases}$$

3. Наличие встречного ветра (параметр  $n > 0$ ):

$$\begin{cases} \dot{x} = v, \\ \dot{v} = u - n. \end{cases}$$

Решив данные системы дифференциальных уравнений при  $k = 0,1, m = 0,9, n = 0,05$  получаем следующие выражения для  $x$  и  $v$ :

1. Наличие сопротивления воздуха:

$$\begin{cases} x = -\frac{(-60+t)t^2}{1125}, \\ v = -\frac{1}{375}(-40+t)t. \end{cases}$$

2. Снижение мощности управления  $u$ :

$$\begin{cases} x = -\frac{(-60+t)t^2}{1250}, \\ v = -\frac{3(-40+t)t}{1250}. \end{cases}$$

3. Наличие встречного ветра:

$$\begin{cases} x = -\frac{(-60+t)t^2}{1250}, \\ v = -\frac{3(-40+t)t}{1250}. \end{cases}$$

На рис. 5 представлены траектории на фазовой плоскости, как для идеального случая, так и при наличии возмущающих воздействий.

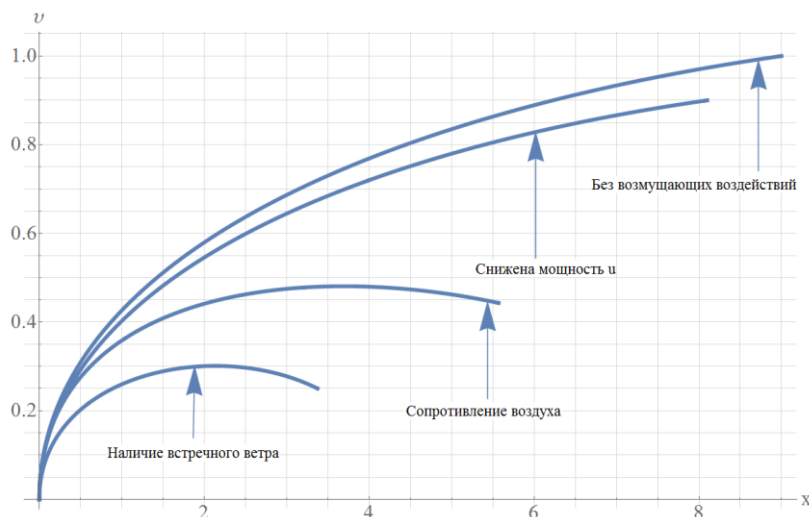


Рис. 5. График на фазовой плоскости

Для оценки ошибки, возникающей при наличии возмущающих воздействий, была использована следующая формула:

$$\varepsilon(t) = \sqrt{(x(t) - x^*(t))^2 + (v(t) - v^*(t))^2},$$

где символом \* помечены возмущенные переменные.

Так, например, формула для оценки ошибки, возникающей при снижении мощности  $u$  примет следующий вид:

$$\varepsilon(t) = \sqrt{\left(-\frac{4t}{375} + \frac{t^2}{3750}\right)^2 + \left(-\frac{2t^2}{375} + \frac{t^3}{11250}\right)^2}.$$

График зависимости ошибки от времени для каждого из возмущающих воздействий представлен на рис. 6.

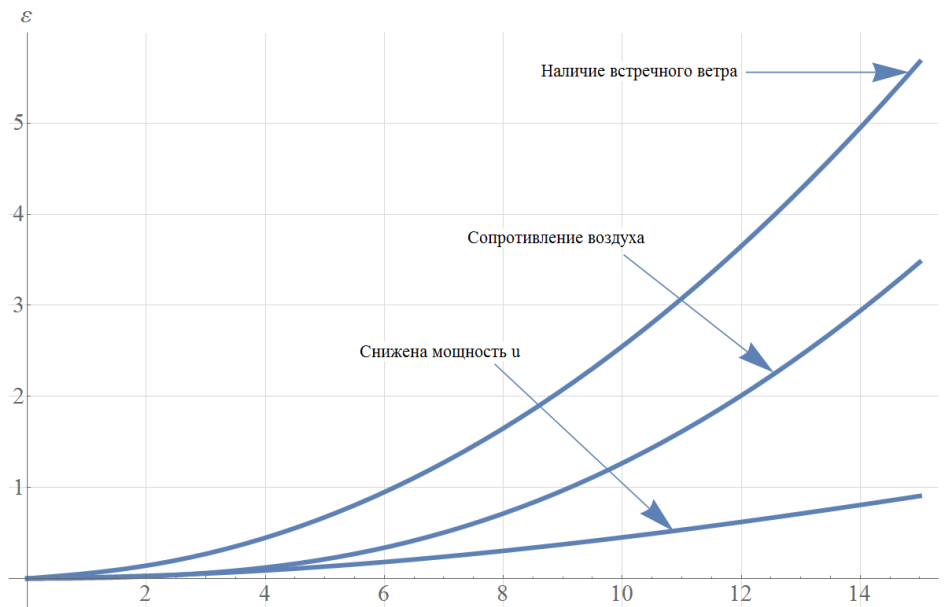


Рис. 6. График изменения ошибки во времени

Как видно из рис. 6, наибольшую опасность представляет ошибка, вызванная наличием встречного ветра. Устранить каждую из ошибок возможно, рассчитав сигнал управления с учётом действующих возмущений.

### Библиографический список

1. Мироновский Л.А. Моделирование линейных систем: учеб. пособие. СПб.: ГУАП, 2009. 244 с.
2. Красовский Н.Н. Теория управления движением. М.: «Наука», 1968. 476 с.

УДК 004.45

**А. А. Пивоваров**

студент кафедры компьютерных технологий и программной инженерии

**М. Д. Поляк**

ст. преп. – научный руководитель

### КОНТЕЙНИРИЗАЦИЯ И ВИРТУАЛИЗАЦИЯ, ПЕРСПЕКТИВЫ, СРАВНИТЕЛЬНЫЙ АНАЛИЗ

Всю свою историю программная инженерия борется со сложностью. Так появилось ООП, так появились компоненты, так появилось разделение системы на слои, различные паттерны и методологии. Эта тенденция легко объяснима. Сложной системой трудно оперировать. Ее сложно чинить, модифицировать, поддерживать, тестировать, добавлять новый функционал,

разворачивать в различных окружениях, одним словом, с такой системой чрезвычайно сложно работать.

Безусловный тренд последних лет, микросервисный подход – еще одна попытка контролировать растущую сложность, но эта попытка, в свою очередь, обернулась многократным увеличением количества операций над приложением. Если мы имеем одно большое монолитное приложение, то его необходимо всего лишь один раз развернуть на тестовом окружении для автоматизированного тестирования, развернуть на другом тестовом окружении для нагрузочного или стрессового тестирования, и, наконец, развернуть у заказчика. При микросервисной архитектуре мы имеем многократное возрастание количества операций над приложением, где множителем будет количество сервисов в системе. Мало того, немалых усилий потребует конфигурирование, настройка сервисов и их окружения.

Второй аспект, так же относящийся к архитектурным предпосылкам – возможность и эффективность горизонтального масштабирования, или, другими словами, эластичность. Предположим некий абстрактный сценарий. У высоконагруженного сервиса  $N$  есть мобильная версия  $M$ , развернутая на отдельной вэб-ферме. Конец рабочего дня, люди едут в транспорте, активно используя мобильные клиенты. Нагрузка на вэб-серверы, обслуживающие мобильную версию сайта, сразу возрастает многократно, становится возможным израсходование всей оперативной памяти потоками, обрабатывающими запросы. Балансировщик вэб-фермы выключает из обслуживания зависшие серверы, нагрузка на оставшиеся серверы возрастает в разы, и в итоге вся система выходит из строя. Выход из ситуации очевидный: нарастить мощность сервера. Но пик нагрузки составляет всего 2–3 часа, остальное время суток большая часть вычислительных мощностей будет простаивать, т. е. вырисовывается неприятная дилемма: терять деньги на простое оборудования или терять пользователей из-за некомфортного времени отклика системы.

Современные методологии разработки предполагают очень быстрые итерации жизненного цикла программного обеспечения. То, что раньше могло занимать месяцы и даже годы, теперь может выполняться в течение двух-трех недель, если говорить, например, о сравнительно новых гибких методологиях. Это означает, во-первых, бурное развитие различных процессов автоматизации в управлении разработкой, а во-вторых, в запросе на быструю оперативную работу с различными окружениями. К примеру, перенесение приложения на окружение для автоматического регрессионного тестирования должно быть автоматизировано и занимать минуты, вместо ручного копирования, правки конфигурационных файлов, создания баз данных с соответствующим набором тестовых данных.

Экономические предпосылки во многом перекликаются с аспектом, связанным с масштабированием системы, только в данном случае это взгляд со стороны не разработчика, а со стороны владельца или арендатора неких вычислительных ресурсов. Современные промышленные компьютеры имеют очень значительную вычислительную мощность, и существует сравнительно небольшое количество задач, способных в пределах одной операционной системы использовать эти ресурсы эффективно. Опять-таки вырисовывается некий запрос на дополнительный слой абстракции, позволяющий эффективно разделять ресурсы, оптимизируя их под потребности конкретного приложения.

Совокупность этих причин и вызвала к жизни появление двух главных стратегий абстрагирования вычислительных ресурсов от программных – так называемую контейнеризацию и виртуализацию. Если рассматривать виртуализацию как общее понятие, а не применительно конкретно к виртуализации ОС, то история данного вопроса насчитывает уже более полувека. Был у виртуализации период исключительно научный, точнее научно-исследовательский, был период разнообразных «песочниц», когда с их помощью различные компании пытались повысить привлекательность своих ОС, позволяя запускать на них наиболее удачные продукты фирм конкурентов. Но настоящий бум начался в начале нулевых с выходом в 2001 г. продуктов ESX Server и GSX Server компании VMware. ESX Server и GSX Server позволяли запускать несколько

ОС на специальной платформе, гипервизоре. Разница между двумя типами серверов была в том, что ESX позволял запускать гипервизор без наличия домашней ОС, непосредственно на оборудовании. Несмотря на то, что вскоре вдогонку ринулись конкуренты: Microsoft со своим Microsoft Virtual Server и компания Citrix с Citrix XenServer, очень скоро VMware покорил корпоративный рынок, захватив, по данным, к примеру, на 2013 г. около 64% рынка виртуальных серверов [1].

В вэб-сегменте индустрии шли не менее интересные процессы, чем в сегменте корпоративном. В 2005 г. Google занялась исследованием задачи представления своих сервисов, таких как, к примеру, Gmail и Google Docs – массовой аудитории потребителей, и, соответственно сразу же начинается поиск технологий и инфраструктурных решений, способных обеспечить эластичность масштабирования вычислительных ресурсов компании. Данные вопросы были для Google весьма критичны. В условиях пользовательской аудитории, потенциально исчисляющейся десятками миллионов клиентов, вполне естественно существенное неравномерное распределение потока пользовательских запросов по времени и, чтобы избежать простоя или наоборот перегрузки оборудования, возможность эластичного регулирования количества узлов системы становится абсолютно необходимой.

Начались эксперименты с традиционной виртуализацией, в результате которых стало очевидно, что:

1. Развертывание традиционной виртуальной машины на основе гипервизора имеет недопустимые временные параметры, что соответственно приведет к большому отклику системы и к крайне негативному пользовательскому опыту.

2. Виртуализация операционной системы целиком приводит к чрезвычайному перерасходу ресурсов (порядка 15–20%), т. е. к снижению так называемой вычислительной плотности, другими словами, к снижению отношения виртуализованных ресурсов, непосредственно участвующих в обработке клиентский вызовов к номинальным вычислительным ресурсам сервера [2].

Главная причина низкой вычислительной плотности – в архитектуре традиционных продуктов серверной виртуализации, в которой для функционирования программного кода необходимо наличие виртуального оборудования и гостевой операционной системы. Довольно скоро взгляды разработчиков обратились к Linux и механизму cgroups (англ. control groups), который разработанному двумя инженерами Google: Полом Менэждем (Paul Menage) и Рохитом Сетом (Rohit Seth) в 2006 г. и с версии 2.6.24 ядра Linux технология включена в официальные версии ядра [3]. Control groups – это довольно-таки широкий программный интерфейс к целому набору средств управления процессами, который в числе прочего ограничивает и изолирует вычислительные ресурсы (процессорные, сетевые, ресурсы памяти, ресурсы ввода-вывода). Google начинает разработку контейнеризации своих серверов на основе cgroups, и этот проект получает название Linux Containers или LXC. Но Google оказывается не одинок в своем внимании к Linux и cgroups, компания Parallels выпускает свой продукт под названием OpenVZ.

Вполне закономерно, что вскоре после Google с теми же проблемами эластичности и вычислительной плотности начали сталкиваться все больше количество топовых компаний, работающих в вэб-сегменте рынка. При этом разные компании работали с разными дистрибутивами Linux, поэтому естественным образом назрела необходимость в некой стандартизации контейнеров. В марте 2013 г. стартапом Docker был опубликован одноименный проект с открытым исходным кодом. Уже в августе число разработчиков, пользующихся Docker, достигло 10 000 [4]. Практически сразу же продукт привлек внимание к себе крупных игроков: в течение года Red Hat и Amazon начали поддерживать Docker, хотя сами разработчики контейнера в то время предостерегали от использования Docker в реальных проектах. 13 марта 2014 г. Docker в сотрудничестве с Canonical, Google, Red Hat, и Parallels в релизе 0,9 заменяет прежде использовавшийся LXC на собственный компонент libcontainer. libcontainer – это написанный на языке Go программный интерфейс к функциям ядра основных дистрибутивов

Linux. Docker и libcontainer произвели самую настоящую революцию в индустрии – все крупнейшие компании провайдеры Интернет-услуг спешно перешли на использование этой связки в своих процессах, плюс к этому, корпоративный рынок был также взломан контейнерной виртуализацией, а это казалось абсолютно невозможным еще некоторое время назад.

Да, безусловно, Docker решил две основные проблемы традиционных виртуальных машин; малую вычислительную плотность и недостаточную эластичность развертывания, но так ли все безоблачно и стоит ли считать контейнеризацию панацеей от всех проблем? Существует, как минимум, пять особенностей этого типа виртуализации, которые, при желании, можно считать недостатками.

libcontainer использует для взаимодействия с ядром Linux пять пространств имен – Process, Network, Mount, Hostname, и Shared Memory. Это означает, что критично важный системный код исполняется не в контейнере. А так как все контейнеры используют общее ядро ОС, некий код, выполняющийся в одном из контейнеров с привилегиями суперпользователя, может легко получить доступ к функциям ядра и обрушить всю систему. Разумеется, есть способы предотвратить это, но они не используются в Docker по умолчанию и требуют дополнительного времени и довольно серьезных навыков администрирования Linux.

В концепции развертывания системы, основанной на контейнерах, система складывается из набора отдельных, независимо поставляемых компонентов. Вэб-сервер здесь будет одним компонентом, база данных – другим, каждый из программных сервисов – третьим. Соответственно встает вопрос версионирования, полноты и качества компонентов – корректная ли версия сервиса, все ли обновления установлены и так далее.

Предыдущий пункт перекликается с трудностью администрирования контейнеров. В системе, состоящей из не одного десятка компонентов, это тоже становится серьезной проблемой. Существует некая точка невозврата, после которой парадигма компонентного разделения ответственностей превращает развертывание системы в крайне тяжело контролируемый процесс.

Одна из ключевых особенностей контейнерной виртуализации, напрямую проистекает из особенностей архитектуры: нельзя иметь в одном контейнере код, работающий под Windows, а во втором – под Debian. И это является серьезным ограничением во многих сценариях использования. К примеру некий кроссплатформенный продукт. Разумеется, его тестирование должно происходить под все целевые платформы, следовательно, тестовое окружение в данном случае должно быть построено на основе традиционных виртуальных машин, при помощи VMware, KVM или каких-либо аналогичных продуктов.

Контейнер конфигурируется раз и навсегда перед запуском. Это не имеет большой роли в вэб-приложениях, которые изначально пишутся с учетом возможности горизонтального масштабирования, но имеет колоссальное значение для корпоративных систем. Большинство программ и программных комплексов для этого сектора рынка разрабатывалось и продолжает разрабатываться до сих пор без учета требований горизонтального масштабирования, в частности 1С – безусловно, одна из самых распространенных, а возможно и самая распространенная корпоративная система в РФ. Теоретически, начиная с версии 8.1 1С может быть горизонтально масштабирована, но на практике все обстоит несколько иначе. Одним из центральных понятий в архитектуре является понятие сеанса и соединения: каждому активному пользователю присваивается объект сеанса, в котором хранятся все необходимые для работы данные. Сеанс хранится до конца работы, или может быть удален по тайм-ауту. Сеанс осуществляет свои запросы благодаря объекту соединения. Наличие подобных объектов в архитектуре является классическим признаком архитектуры, поддерживающей состояние, которая по своей природе горизонтальному масштабированию не подлежит. В общем, подводя итоги: стратегия масштабирования напрямую проистекает из архитектуры программы и, исходя из этого, некоторое программное обеспечение масштабированию с помощью контейнеров не подлежит.

Можно с уверенностью говорить, что в ближайшие годы мы будем видеть интенсивный рост контейнерной виртуализации, поддерживаемой и направляемой интернет-гигантами и крупнейшими хостинговыми компаниями. Вероятнее всего, в ближайшем будущем компания Microsoft объявит о портировании Docker под свой технологический стек и последующей тесной интеграцией с облачной платформой Azure. Самыми перспективными направлениями развития данного типа виртуализации можно обозначить улучшения в области безопасности и создание средств по более эффективному управлению большим количеством контейнеров в сложной компонентной системе. Если рассматривать вопрос с маркетинговой точки зрения, безусловно будет происходить дальнейшая экспансия контейнеров на корпоративный рынок и постепенное вытеснение виртуализации на уровне операционных систем. В плане операционном – наверняка мы увидим тесное взаимодействие и интеграцию различных стратегий виртуализации. Мы будем наблюдать упрощение, а значит и ускорение процессов поставки и развертывания приложений в различных окружениях, что позволит осуществлять более короткие циклы разработки и более тесную коммуникацию с заказчиком и/или с клиентами. В среднесрочной и долгосрочной перспективе, вероятно, будет происходить некая унификация принципов и практик разработки корпоративных и веб-приложений, границы между которыми будут постепенно размываться.

#### **Библиографический список**

1. <https://habrahabr.ru/company/1cloud/blog/237005/>
2. <http://www.datamation.com/data-center/virtualization-vs.-containers-what-you-need-to-know.html>
3. <https://ru.wikipedia.org/wiki/Cgroups>
4. <http://searchservervirtualization.techtarget.com/feature/A-brief-history-of-Docker-Containers-overnight-success>

**УДК 629.735.33**

***К. Д. Пожарова***

студентка кафедры компьютерных технологий и программной инженерии

***А. В. Фомин***

к-т техн. наук., доц. – научный руководитель

### **СБОР, СОХРАНЕНИЕ И АНАЛИЗ ДАННЫХ О ТРЕНИРОВКАХ И ПИТАНИИ ЧЕЛОВЕКА**

Каждый человек сталкивается с тем, что хочет упорядочить некую информацию о себе, о том, что он делает, какой результат от этих действий он получает.

Раньше это было возможно путем записи данных в дневник или тетрадь, где человек мог анализировать эти данные. Однако, это имело ряд неудобств, таких как громоздкость записей и неудобство их чтения. Сейчас, в век цифровых технологий, появились компьютеры, которые могут хранить, обрабатывать и анализировать информацию.

А что если пользователю необходим доступ к его «базе данных» на расстоянии, когда он занимается тем, что потом занесет в дневник, например занимается в тренажерном зале и ему нужно заносить каждое упражнение, каждый подход и количество повторений в дневник? На помощь приходят веб-сайты и приложения для смартфонов. Зайдя на сайт, человек может занести данные в свою «базу данных», где потом с помощью функций сайта, может проанализировать то, что он делал. Также это возможно сделать с помощью приложения на смартфоне.

В самом начале проекта был разработан веб-сайт, который реализует вышеперечисленные функции, в добавок те, которые описаны ниже. В дальнейшем, будет реализовано приложение для смартфонов. Планируется сделать приложение для смартфонов



фирмы Apple, в которое пользователь сможет заполнять данные не только с сайта, но и с приложения у себя на телефоне.

Данный веб-сайт реализован как для простых пользователей, так и для тренеров. Он состоит из календаря (рис. 1), нажимая на дату которого, пользователь попадает на страницу формы заполнения данных о своих тренировках и питании (рис. 2).

Пользователь может ввести данные о тренировке, такие как: время начала тренировки, время ее окончания, средний пульс (очень важно для тех, кто хочет сбросить лишнюю жировую массу), тип тренировки (какую часть тела он тренировал), также, он может обновить данные, удалить и добавить упражнение. Нажимая на «Exercises» пользователь попадает на страницу, на которой может добавить те упражнения, которые он делал в тренажерном зале. Также он может добавить количество подходов и количество повторений данного упражнения (рис. 3).

На странице заполнения данных о себе пользователь может добавить информацию о питании. Для этого есть еще одна форма, куда пользователь вносит такие данные, как: время приема пищи, вид и характер принятой пищи, количество грамм соответственно и сколько калорий это составило.

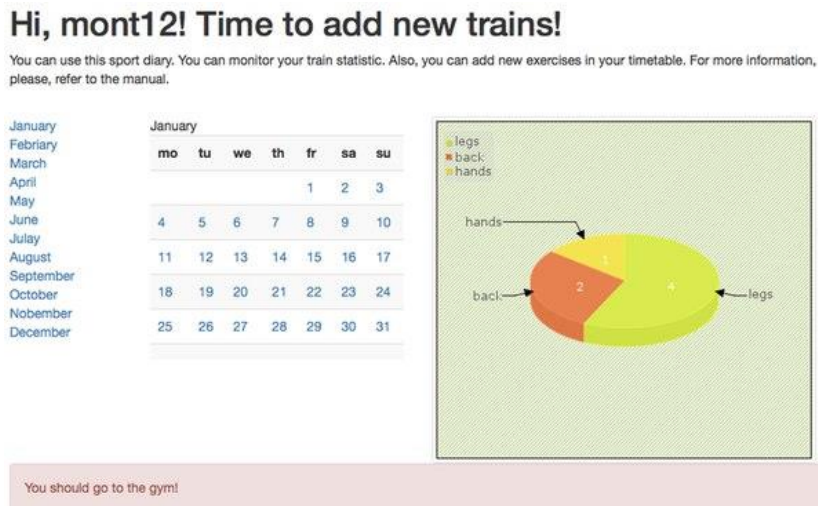


Рис. 1. Главная страница



Рис. 2. Страница заполнения/просмотра данных о питании и о тренировках

name	sets	reps	weight
squat	1	15	17
squat	2	16	22
squat	3	20	25
deadlift	1	15	20
deadlift	2	15	22
deadlift	3	17	25

squat Choose exercise

Enter set

Enter reps

Enter weight

send

Рис. 3. Форма заполнения данных для упражнений

Приложение рассчитывает, сколько калорий рекомендовано для употребления конкретного пользователя. При регистрации он вводит такие данные, как рост, вес, возраст и пол. И по данной формуле происходит расчет суточной нормы потребления калорий [1]:

$$OO = a + b - v,$$

где  $a$  – вес в килограммах, умноженный на 9.99;  $b$  – рост в сантиметрах, умноженный на 6.25;  $v$  – возраст в годах, умноженный на 492.

Когда пользователь вводит данные о питании и калориях, ему всплывают подсказки о том, какую пищу он может есть (тяжелую легкую, комбинированную). Также если пользователь не заполняет данные о тренировках, то всплывает флаг о том, что пора тренироваться.

Также рядом с календарем находится график, который показывает, сколько тренировок и какой группы мышц было сделано пользователем.

Для реализации серверной части веб-сайта применен скриптовый язык общего назначения php[2], и система управления базами данных MySQL. Данный веб-сайт функционирует стабильно, за время тестового использования данного веб-сайта критических ошибок выявлено не было. Поскольку серверная часть приложения была написана без использования библиотек php, то имеет небольшие минусы, такие как громоздкость кода и трудность доработки кода. В дальнейшем планируется переделать веб-сайт используя фреймворк Codeigniter.

В целом веб-сайт позволяет пользователю контролировать свои тренировки и питание, дисциплинировать себя. Как следствие этого, он мотивирует себя на дальнейший прогресс, а записи легко просматриваются и их анализ не вызывает никаких проблем.

#### Библиографический список:

1. Кузнецов М. В., Симдянов И. В. РНР. Практика создания Web-сайтов. М.: БХВ-Петербург, 2015. 1264 с.
2. Крестовников А. Б., Коссовская Э. В. Физиология человека. М.: Физкультура и спорт, 1954. 528 с.

УДК 004.415.28

**О. В. Полех**

студентка кафедры компьютерных технологий и программной инженерии

**А. В. Бржезовский**

канд. техн. наук, доц.) – научный руководитель

### ФОРМАЛЬНЫЕ СРЕДСТВА СПЕЦИФИКАЦИИ

В современном мире все чаще используются формальные методы спецификации при проектировании программного обеспечения. Ведь в условиях конкуренции важно выпустить продукт в срок, с высоким качеством и лучшим соотношением цена/производительность. Использование формальных методов спецификации является одним из важных условий достижения этих целей, как как они позволяют повысить точность требований спецификации и сводят к минимуму риски при разработке программного обеспечения.

Формальные средства спецификации используют математическую нотацию для точного описания свойств, которыми должна обладать система, без чрезмерного ограничения метода их реализации. Формальные спецификации отражают, что система должна делать, без указания того, как она должна это делать. Существует целый ряд формальных средств спецификации, рассмотрим некоторые из них: Z-нотацию, OCL (Object Constraint Language) и онтологию.

Z-нотация

Разработана Исследовательской группой по программированию Оксфордского университета в конце 70-х гг. XX в. и используется промышленными предприятиями Америки и Европы при проектировании высокоточных систем [1].

Z нотация используется для описания и моделирования вычислительных систем. Она основана на теории множеств и предикативной логике первого порядка. Она представляет данные в виде множеств, связей, кортежей, отображений, последовательностей и декартовых произведений, позволяет определять функции и операции для обработки данных этих типов. Эти объекты не ориентированы на машинное представление, но они подчиняются многообразным математическим законам, которые позволяют эффективно рассуждать о поведении указанной системы. Логика предикатов используется для абстрактного описания результатов каждого действия в системе.

Другая важная составная часть в Z-нотации – декомпозиция спецификации на небольшие части, называемые схемами. Каждая схема может быть связана с комментарием. Он, в свою очередь, помогает объяснить неформально значение формальной математики. Такое представление от общего к более детальному позволяет обосновать корректность спецификации, которая содержит больше деталей конкретного проекта [2].

Схемы являются описаниями сущностей и связей между ними. Они предоставляют среду для разработки спецификаций системы и их постепенного развития и уточнения. Схема состоит из двух частей: объявления переменных и предиката (рис. 1, 2).

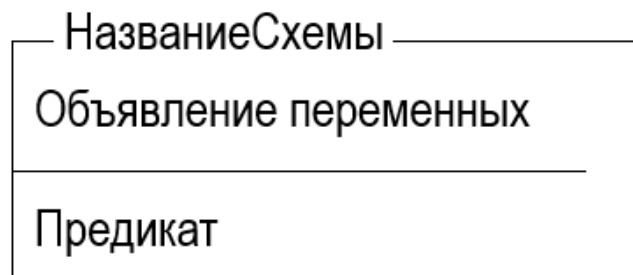


Рис. 1. Структура схемы Z-нотации

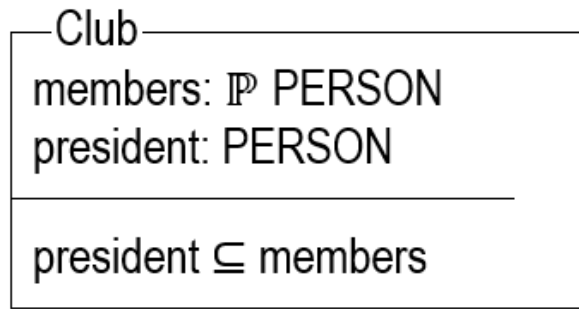


Рис. 2. Пример схемы Z-нотации.

Пусть имеется множество всех персон [PERSON]. 'Club' имеет множество членов клуба и одного президента, который является членом клуба.

Переменные схемы определяют сущности, отображающие состояние схемы. Предикатами являются ограничения булевого типа, которые должны быть всегда истинны для всех сущностей. Предикаты могут представлять пред- и постусловия для операций, если они включены в схему.

### OCL

Диаграмма классов UML (Unified Modeling Language) недостаточно уточнена, чтобы обеспечить все соответствующие аспекты спецификации. Существует необходимость описать дополнительные ограничения, налагаемые на объекты модели. Такие ограничения часто представляют на естественном языке, но это приводит к неоднозначности. Для того чтобы описать однозначные ограничения, придумали язык OCL, который разработан как язык бизнес-моделирования в IBM [3].

Язык OCL основан на выражениях и является типизированным языком. Каждое выражение возвращает некоторое значение. Средствами OCL нельзя изменить диаграмму классов UML и значения атрибутов классов, следовательно, состояние системы не измениться (рис. 3).

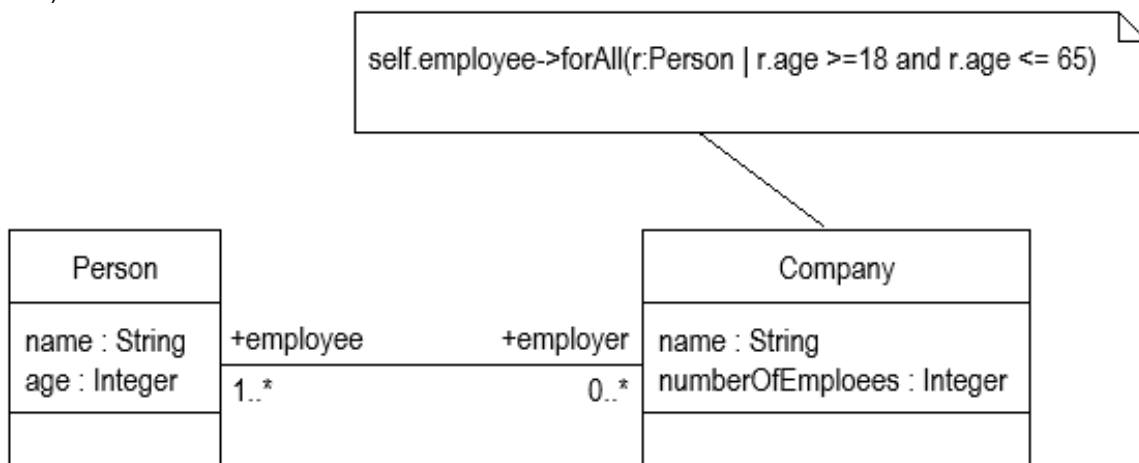


Рис.3. Пример ограничения OCL.

Возраст всех сотрудников должен быть от 18 до 65 лет.

Язык OCL может использоваться в качестве навигации по элементам модели или задавать условия и ограничения на элементы модели. В OCL могут определяться ограничения, описывающие пред- и постусловия для операций классов, и ограничения, представляющие собой инварианты классов. Инвариантом класса является логическое выражение, которое должно быть истинным для всех экземпляров этого класса в любое время.

ОСЛ содержит необходимую строгость высказывания и достаточно легко интерпретируются как человеком, так и вычислительной системой.

#### Онтология

Онтология позволяет представить понятия в таком виде, в котором они становятся пригодными для машинной обработки. Онтологией является модель описания предметной области, формальное представление которой включает словарь указателей на термины предметной области и логические выражения. Логические выражения описывают значения терминов, их соотношение и их связь друг с другом. Таким образом, онтология обеспечивает словарь для представления и обмена знаниями о некоторой предметной области и множество связей, установленных между терминами в этом словаре [4].

Формальная модель онтологии:

$$O = \langle T, R, F \rangle,$$

где  $T$  – конечное множество терминов предметной области, описываемая онтологией  $O$ ;  $R$  – конечное множество отношений между терминами предметной области;  $F$  – конечное множество функций интерпретации, задающиеся на терминах и отношениях онтологии [5].

Онтология на формальном уровне является системой, состоящей из наборов понятий и утверждений об этих понятиях. На их основе можно строить объекты, классы, функции и отношения. Большинство моделей онтологии содержат определенные концепты (классы), свойства концептов (атрибуты, роли), отношения между концептами (зависимости, функции) и дополнительные ограничения, определяющиеся аксиомами [6].

Концепт – это шаблон со множеством правил. Концепты определяют форму экземпляра, иначе говоря, каким образом может быть построен экземпляр. Описание действия, стратегии, функции или задачи может быть концептом. Экземпляр концепта представляет элемент предметной области, а концепт описывает группу элементов. Множество экземпляров и сама онтология составляют базу знаний.

Области искусственного интеллекта, поисковых систем, инженерии знаний, представления знаний, биомедицинской информатики и при решении задач семантической интеграции информационных ресурсов используют онтологии для ограничения сложности и организации информации.

Формальные спецификации полезны в процессе разработки программного обеспечения, так как позволяют уверенно отвечать на возможные вопросы о функциональности системы, без необходимости углубления в детали программного кода или размышлений о значении фраз, содержащихся в текстовом описании требований.

#### Библиографический список

1. С. Крылов Документирование требований к ПО (часть 2), URL: <http://www.comprice.ru/articles/detail.php?ID=246065>.
2. J. M. Spivey. The Z-Notation: A Reference Manual: Oriel College, Oxford, OX1 4EW, England, 1992, 168с.
3. Object Constraint Language, Version 2.2: OMG, 2010, 224p, URL: <http://www.omg.org/spec/OCL/2.2/PDF>.
4. BaseGroup Labs: Онтология анализа данных, URL: <https://basegroup.ru/community/articles/ontology>
5. А. Я. Гладун, Ю.В. Рогушина. Онтологии в корпоративных системах. Часть 2// Корпоративные системы, 2006, №1, URL: <http://www.management.com.ua/ims/ims116.html>.
6. А. Я. Гладун, Ю.В. Рогушина. Онтологии в корпоративных системах. Часть 1// Корпоративные системы, 2006, №1, URL: <http://www.management.com.ua/ims/ims115.html>.

УДК 004.51

**А. А. Розов**

студент кафедры компьютерных технологий и программной инженерии

**М. Д. Поляк**

ст. преподаватель – научный руководитель

### **СРАВНЕНИЕ КАРТОГРАФИЧЕСКИХ СЕРВИСОВ ДЛЯ РЕАЛИЗАЦИИ ГЕОИНФОРМАЦИОННОЙ СИСТЕМЫ ГИДРОБИОЛОГИЧЕСКОГО МОНИТОРИНГА (ПРЕДЛОЖЕН АНАЛИЗ СУЩЕСТВУЮЩИХ СЕРВИСОВ НА ОСНОВЕ ТРЕБОВАНИЙ К ГЕОИНФОРМАЦИОННОЙ СИСТЕМЕ – ГИС)**

При разработке веб-интерфейсов, реализующих работу с большим количеством точек и областей на карте, очень важно определиться с набором необходимых инструментов, присутствующих в API картографического сервиса. Для задачи визуализации данных гидробиологического мониторинга необходимы следующие инструменты:

- Вывод координат на карту.
- Группировка координат, находящихся рядом.
- Выбор координат из группы путем выделения области на карте.
- Выделение области карты цветами.
- Получение координат в формате JSON.

Критерием выбора наиболее подходящего картографического сервиса было наличие советующего API, позволяющего реализовать требуемый функционал ГИС. К серверной части предъявляются требования по работе с разработанной ранее базой данных и предоставление нужных данных картографическому сервису и пользователю.

Для сравнения были выбраны три картографических сервиса:

- Open Street Maps;
- Google Maps;
- Яндекс.Карты.

Первый сервис, который был рассмотрен, – Open Street Maps.

Данный проект не имеет никаких ограничений по использованию и распространяется свободно. Карту рисуют сами пользователи, поэтому местами она имеет высокую детализацию, а местами – большие пробелы. Подключение карты происходит путем загрузки видимой области карты с сервера, на котором расположен сайт. Сама карта храниться локально в месте со всем проектом в виде файлов с расширением *osm*.

Работа с картой происходит на стороне клиента с использованием языка JavaScript и библиотеки JQuery. Из минусов системы стоит отметить отсутствие единой документации, ответы на вопросы и документирование осуществляются волонтерами через форум проекта. Обновление карт можно производить вручную, загружая свежую версию только нужной части карты, что уменьшает нагрузку на сеть, или всей карты, что позволяет не изменять карту при расширении зоны визуализации данных.

Для работы с картами серверная часть может быть реализованная на любом языке программирования. Загрузка гео-данных на карту происходит при помощи занесения их в специальный XML-файл путем выполнения AJAX запроса. Вывод точек на карту производится из полученного XML-файла.

Карта поддерживает такую важную функцию при работе с гео-данными как кластеризация. Кластеризация – это объединение данных (точек) с совпадающими координатами или находящихся достаточно близко друг от друга, так что при уменьшении масштаба карты они могут быть объединены в один кластер. При этом имеется возможность просмотреть метки, попавшие в кластер, путем вывода их из общего количества или списка, если они имеют разные координаты.

Также важным аспектом для оценки картографического сервиса является наличие возможности для построения температурной карты, позволяющей отображать на карте концентрации различных гидробиологических параметров с помощью изменения цветового градиента. В рассматриваемой картографической системе Open Street Maps нет встроенной возможности для визуализации температурных карт [1].

В виду того, что система Open Street Maps разрабатывается сообществом и не имеет единой цели развития и поддержки перехода к новым версиям, для обозначенных целей разработки интерфейса ГИС она не подходит.

Второй рассматриваемой картографической системой являются карты от компании Google. Данный картографический сервис, в отличие от предыдущего, разрабатывается частной компанией, и для получения доступа к нему необходимо сгенерировать специальный ключ доступа. Бесплатный доступ предоставляется на условии, что число запросов к карте в месяц не будет превышать 25 000.

Карта хорошо документирована и частично переведена на русский язык. Подключение происходит путем загрузки видимой части карты с внешнего сервера. Данный способ позволяет уменьшить нагрузку как на клиентское приложение, так и на сервер ГИС. Еще одним достоинством является то, что поддержание карты в актуальном состоянии и ее хранение ложатся на поставщика карты (компанию Google) в отличие от подхода, используемого в Open Street Maps.

Для загрузки данных на карту применяются AJAX запросы, через которые идет обращение к специальному файлу. Данные гео-меток хранятся в файлах формата JSON или XML. Запись данных в файлы производится серверной частью приложения.

Вывод точек на карту может производиться двумя способами:

- указание их в коде подгружаемого скрипта на JavaScript;
- указание их в специальном файле в формате JSON.

В API карт компании Google встроен такой важный механизм как кластеризация, так же есть возможность создания активных областей. Активные зоны – это особо важная часть API-карт. Она позволяет нарезать готовые зоны на карте и проводить выборку только по ним, или самостоятельно создать зону путем построения геометрических фигур. Возможность выбора группы позволяет проводить сравнение между элементами зоны. Этот инструмент дает широкие возможности при работе с картой.

Google-карты имеют удобные инструменты для вывода пользовательских слоев, позволяющих изменить представление данных [2].

Сами карты имеют хорошую детализацию на территории США. На территории РФ детализация не такая точная, но все же достаточная для задач гидробиологического мониторинга. При масштабировании проекта особых усилий и изменения кода не потребуется.

Третьей рассмотренной системой являются карты от компании Яндекс.

Сервис имеет бесплатный доступ, ограниченный, как и в Google, 25000 запросов в день, но, в отличии от карт Google, весь API предоставляется бесплатно. Платными являются только дополнительные расширения, не входящие в базовый функционал.

Вся документация карты представлена на русском языке. Имеется служба поддержки разработчиков как через форму обратной связи, так и через специальный форум.

Карты встраиваются на сайт как внешний JavaScript файл или как статичное изображение, получаемое по HTTP запросу. Для разработки ГИС предпочтительнее подключение внешнего файла, так как при таком подключении карта является динамической и дает больший простор для визуализации данных.

Загрузка изображений и отображение гео-объектов производятся лишь в видимой части карты. При таком подходе нагрузка на клиентскую часть уменьшается и с точки зрения пользователя карты работают быстрее. Данные передаются для отображения на карте с использованием AJAX запроса, который возвращает файл в формате JSON или XML.

Карты компании Яндекс имеют в своем составе схожий с Google Maps набор инструментов. Они так же обладают инструментами кластеризации, но в отличии от Google Maps имеют 3 различных способа его реализации. Активные области могут задаваться как статично, так и по усмотрению пользователя, путем построения геометрических фигур поверх карты. Карта поддерживает работу со слоями, которые могут выводиться сразу при загрузке карты или по нажатию на заданный элемент.

Немаловажным элементом API является температурная карта, позволяющая визуально отобразить на карте концентрацию какого-либо параметра в заданной области [3].

Карты имеют высокую степень детализации для территории РФ. Не менее существенным плюсом является тот факт, что при выходе новой версии Яндекс описывает способы перехода на нее. Старые версии поддерживаются в рабочем состоянии в течении длительного времени, что обеспечивает плавный переход от версии к версии, незаметный для конечных пользователей. В таблице приведен краткий анализ упомянутых картографических систем.

«Сравнение характеристик картографических сервисов»

Характеристика	Яндекс.Карты	Google Maps	Open Street Maps
Хранение карты	На серверах Яндекса	На серверах Google	На сервере приложения
Кластеризация	Да	Да	Нет
Активные зоны	Да	Да	Нет
Способ хранения данных	JSON, XML	JSON, XML	XML
Изменение вида карты	Наложение новых слоев	Наложение новых слоев	Редактирование самой карты
Поддерживаемые форматы	JSON, XML	JSON, XML	XML
Формат координат	Широта /Долгота Долгота/Широта	Долгота/Широта	Долгота/Широта

В результате сравнительного анализа для реализации геоинформационной системы гидробиологического мониторинга наилучшим образом подходит картографический сервис, предоставляемый компанией Яндекс. Сервис обеспечивает наиболее широкий набор встроенных инструментов для разработчиков и, что немаловажно, дает возможность для построения пользовательских модулей, расширяющих функционал картографического сервиса.

### Библиографический список

1. WIKI-страница Open Street Map <http://www.openstreetmap.org/> (04.03.2016)
2. Сайт Google Maps API <https://developers.google.com/maps/web/> (03.03.2016)
3. Сайт Яндекс.Карты API <https://tech.yandex.ru/maps/doc/jsapi/2.1/dg/concepts/about-docpage/> (05.03.2016)



УДК 556.3.01

**А. А. Свиридова**

студентка кафедры компьютерных технологий и программной инженерии

**М. А. Нарбут**

д-р. физ.-мат. наук, профессор – научный руководитель

## МАТЕМАТИЧЕСКИЕ ОСНОВЫ ГЕОЭЛЕКТРИЧЕСКИХ МЕТОДОВ ОБНАРУЖЕНИЯ ПОДЗЕМНЫХ ИСТОЧНИКОВ ВОДЫ

Геофизические методы исследований предназначены для изучения верхних слоев Земли, инженерно-геологических и гидрологических исследований. Геофизическая разведка проводится при поисках нефти и газа, рудных полезных ископаемых и подземных вод [1]. Геофизические методы исследований основаны на изучении физических полей и физических свойств пород и минералов, их слагающих. Геофизические методы делятся на два типа. К первому относятся методы измерения естественных физических полей – гравитационного, магнитного и электрического, теплового, ко второму типу относятся исследования искусственно создаваемых физических полей.

Объектом разведки недр и геофизических исследований можно рассматривать скважину. Скважина является важным источником информации о строении недр и местонахождении полезных ископаемых. Нахождение параметров поля в скважине по заданным параметрам его источников и характеристикам среды называют прямой задачей геофизического исследования скважин (ГИС). На практике, напротив, по измеренным в скважине параметрам поля определяют характеристики среды, т.е. решают обратную задачу. Специфика обратных задач ГИС в том, что из-за недоступности исследуемого объекта о его параметрах судят по косвенным проявлениям. Так, о горной породе, расположенной вне скважины, судят, измеряя характеристики поля в скважине. Классификация методов ГИС может быть выполнена по виду изучаемых физических полей. Их делят на электрические, электромагнитные, ядерно-физические, сейсмоакустические, гравитационные, магнитные, термические, геохимические и некоторые другие.

Одним из методов, в реализации которого используют электрическое поле, является метод естественной поляризации (или самопроизвольной поляризации). Метод потенциалов самопроизвольной поляризации (ПС или СП) основан на измерении в разрезах скважин естественного постоянного электрического поля Земли. Это поле создается вследствие окислительно-восстановительных, фильтрационных и диффузионно-адсорбционных процессов. Окислительно-восстановительные процессы в основном возникают на контакте с рудными телами при обязательном присутствии водных растворов (природный гальванический элемент). Фильтрационные процессы связаны с динамикой подземных вод. Диффузионно-адсорбционные процессы имеют место при контакте водных растворов с различной концентрацией.

Фильтрацией называется движение жидкости в пористой среде. Движение происходит не по всем, а только связанным друг с другом достаточно крупным порам.

Скорость движения воды в пористых средах, заполняющих водоносные горизонты, определяется по закону Дарси:

$$w = -k \operatorname{grad} H \quad (1)$$

где  $H$  – гидравлический напор;  $k$  – коэффициент фильтрации.

Гидравлический напор, пропорциональный давлению  $p$ , в простейших случаях удовлетворяет уравнению фильтрации:

$$\Delta H = \frac{1}{a} \frac{\partial H}{\partial t} \quad (2)$$

где  $a$  – коэффициент пьезопроводности.

В стационарном случае имеем уравнение Лапласа:  $\Delta H = 0$ . (3)

В результате фильтрации образуется потенциал течения. Потенциал течения – это явление возникновения разности электрического потенциала в дисперсной системе при фильтрации сквозь нее жидкости относительно неподвижной твердой дисперсной фазы. Для возникновения электро-кинетического потенциала течения необходимо наличие развитых ДЭС вокруг частиц дисперсной системы. Вокруг частиц тонкодисперсных грунтов (глины, суглинки, почвы, торф, илы и др) могут существовать ДЭС разной толщины, так как они являются капиллярно-пористыми дисперсными системами.

Течение воды в водоносных слоях порождает слабый электрический ток с плотностью:

$$j = Qw , \quad (4)$$

которой отвечает распределению электрического потенциала  $U$ . Измерения потенциала производятся на земной поверхности.

Предполагаем, что распределение потенциала в среде следует уравнению:

$$\operatorname{div}(\sigma \operatorname{grad} U) = \operatorname{div} j , \quad (5)$$

где  $\sigma$  – коэффициент электропроводности среды.

При постоянном коэффициенте электропроводности получаем уравнение Пуассона:

$$\Delta U = \rho \operatorname{div} j , \quad (6)$$

где  $\rho$  – удельное сопротивление среды,  $\rho = \frac{1}{\sigma}$  (7)

Прямая задача электроразведки на воду заключается в определении потенциала  $U$  на земной поверхности при заданном гидравлическом напоре  $H$ . Обратная задача: по известным значениям потенциала  $U$  на поверхности оценить параметры водоносного пласта: коэффициента фильтрации  $k$  и коэффициента пьезопроводности  $a$ . Эти задачи представляют собой краевые задачи математической физики и могут быть решены с помощью метода конечных разностей или метода конечных элементов (2 – 4). Основная идея метода конечных элементов состоит в том, что любая непрерывная величина аппроксимируется дискретной моделью, построение которой выполняется на множестве кусочно-непрерывных функций, определенных на конечном числе подобластей. Процесс решения задачи методом МКЭ включает в себя следующие этапы:

1. Дискретизация области: построение сетки. Область, на которой решается задача, аппроксимируется (покрывается) непересекающимися подобластями простого типа, которые называются конечными элементами (КЭ).

2. Выбор аппроксимирующих (базисных) функций. Чаще всего базисные функции выбираются в виде полиномов. Поэтому пространство, на котором ищется решение, является пространством кусочно-полиномиальных функций. Базисные функции могут иметь различный порядок: линейный, квадратичный, кубичный и т.д.

3. Формирование СЛАУ с учетом вкладов от элементов и узлов, введение граничных условий в систему уравнений.

4. Решение системы уравнений.

5. Определение расчетных величин в элементах. Этими величинами обычно являются производные от неизвестной функции (например, деформации, напряжения, тепловые потоки, скорости).

Точное решение дифференциального уравнения при подстановке в это дифференциальное уравнение обращает его в тождество в каждой точке. Методом КЭ находится приближенное решение  $\bar{u}$  задачи [2,3]. Решение аналогичной задачи полуаналитическими методами представлено в работе [5].

Программным средством для решения поставленных задач был выбран комплекс COMSOL Multiphysics, позволяющий решать методом конечных элементов краевые задачи различной физической природы.

Пример двумерной осесимметричной области изображен на рис.1.

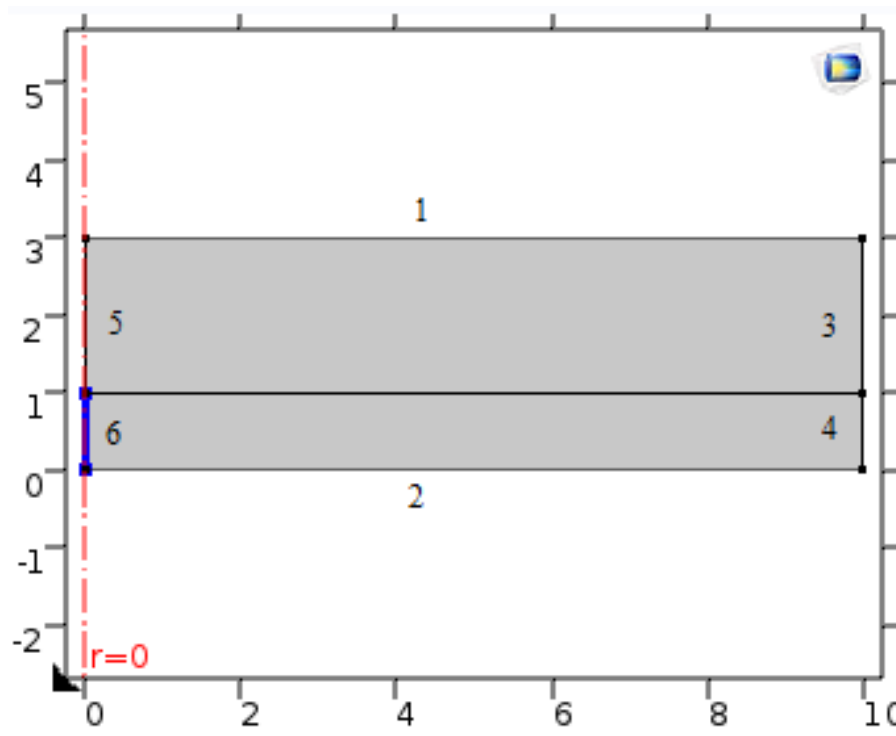


Рис. 1. Заданная область

Область состоит из двух слоев. Скважина расположена слева от области. Границы 1, 2 – приняты изолирующими. На границе 3,4 задано условие Дирихле ( $u = 0$ ) в предположении, что эти границы значительно удалены от скважины, и входящий в нее поток не влияет на результат исследования. На границе 5 и 6 заданы граничные условия Неймана. На границе 5 исходящий поток равен 0, на границе 6 – задано числовое значение 2.

Для вычисления значения гидравлического напора  $H$  в заданной области используется уравнение (2). Результаты расчета представлены на рис.2 и рис.3

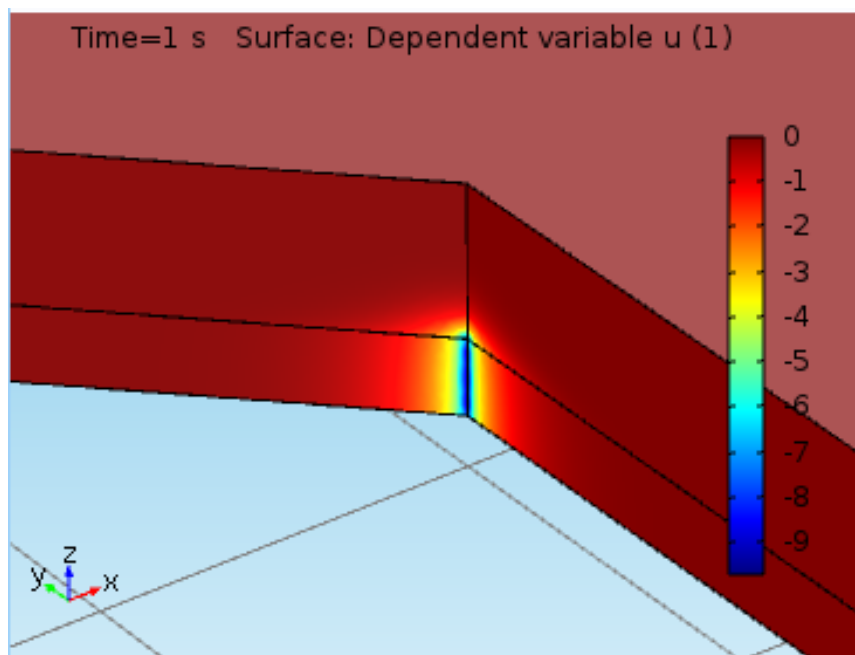


Рис. 2. Результаты моделирования 3d

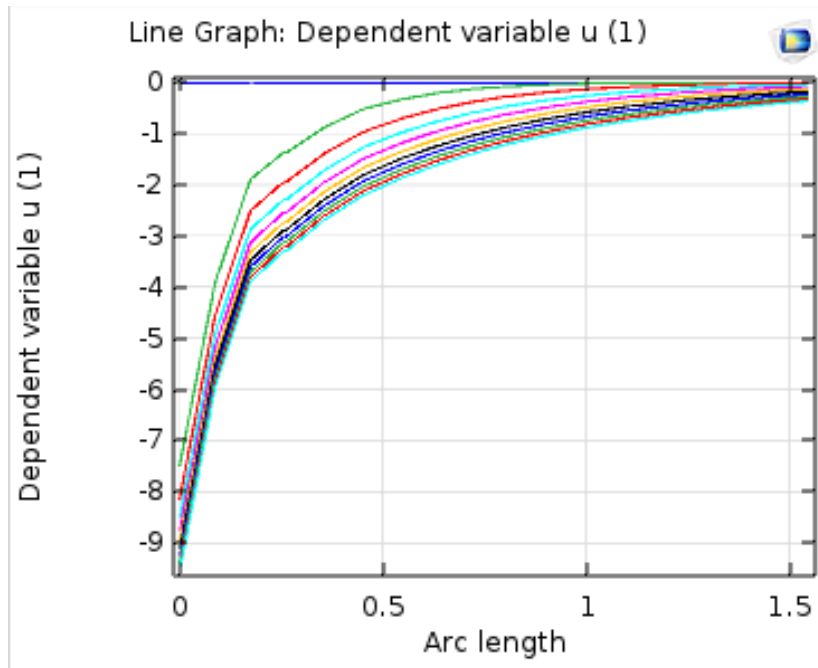


Рис. 3. Результаты моделирования 1d

Электрический потенциал находим как решение уравнения

$$\sigma \Delta U = \text{div}(Qw),$$

или

$$\sigma \Delta U = L \Delta H, ,$$

где  $L$  – электрокинетический коэффициент. Полученный в модели 1 результат – массив значений  $H$  был подан на вход модели 2. Пример расчета показан на рис. 4.

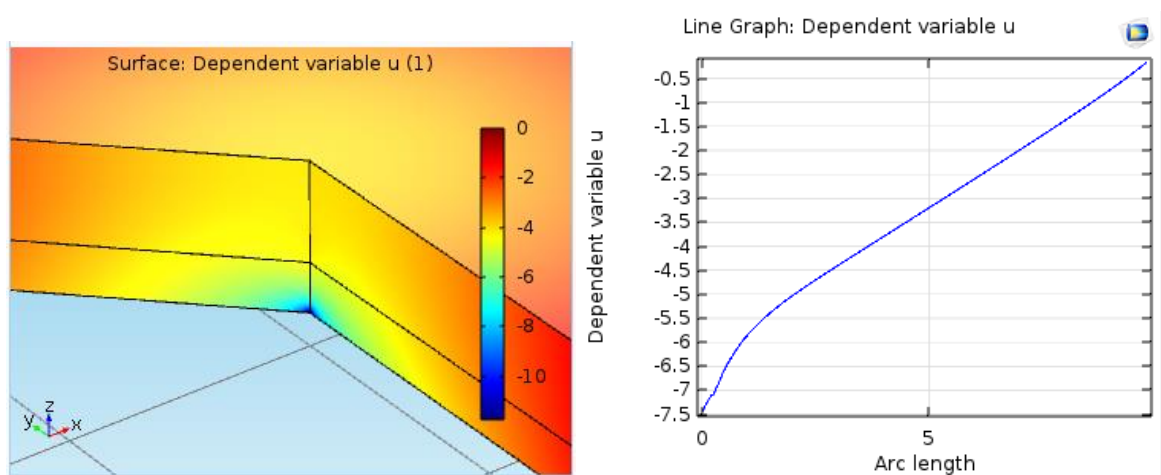


Рис. 4. Результаты моделирования электрического поля

### Библиографический список

1. Басниев К.С., Кочина И.Н., Максимов В.М. Подземная гидромеханика: учебник для вузов. М.: Недра, 1993. 416 с.
2. Андреев В.Б. Лекции по методу конечных элементов: учеб. пособие. М.: Издательский отдел факультета ВМиК МГУ им. М.В. Ломоносова, МАКС Пресс, 2010. 2-е изд. 264 с.
3. Нарбут М.А. Вычислительная геофизика: учеб. пособие. СПб.: С.-Петербург. ун-та, 2014. 200 с.

4. Рояк М.Э., Соловейчик Ю.Г., Шурина Э.П. Сеточные методы решения краевых задач математической физики: учеб. пособие. Новосибирск: Изд-во НГТУ, 1998. 120 с.
5. B. Malama, A. Revil, K. L. Kuhlman. A semi-analytical solution for transient streaming potentials associated with confined aquifer pumping tests – Geophys. J. Int. (2009) 176, 1007–1016

УДК 004.514

**А. В. Сергеев**

студент кафедры проблемно-ориентированных вычислительных комплексов

**Е. А. Бакин**

канд. техн. наук, доцент. – научный руководитель

### СРЕДСТВО ВИЗУАЛИЗАЦИИ СОСТОЯНИЯ СЕТИ МАССОВОГО ОБСЛУЖИВАНИЯ

Оценка и оптимизация параметров систем, предназначенных для многоразового выполнения идентичных задач, является достаточно нетривиальной задачей из-за сложности описания составляющих систем и происходящих в них процессов. Зачастую сложность связана либо с созданием математической модели, которая бы поддавалась численному анализу, либо с ограниченностью возможностей для проведения экспериментов на реальной системе [1,2]. Для описания и оценки параметров подобных систем используется имитационная модель сети массового обслуживания. Сеть представляет собой совокупность множества обслуживающих устройств, которые состоят из сервера, обслуживающего заявки, и буферов, хранящих заявки, которые ожидают обработки [3].

Существует задача создания компьютерного отображения сети, в которую включается разработка алгоритма автоматического расположения обслуживающих устройств и прокладки маршрутов между ними.

В решении задачи расчета и отображения маршрутов между устройствами используются графовые модели. Главным требованием к отображению графа является наглядность. Обеспечение данного требования осуществляется благодаря соблюдению эстетических критериев.

Основными эстетическими критериями являются:

- Пересечения  $\rightarrow \min$ ;
- Область построения  $\rightarrow \min$ ;
- Общая длина ребер  $\rightarrow \min$ ;
- Максимальная длина ребер  $\rightarrow \min$ ;
- Универсальная длина ребер  $\rightarrow \min$ ;
- Общее число изгибов  $\rightarrow \min$ ;
- Максимальное число изгибов  $\rightarrow \min$ ;
- Симметрия  $\rightarrow \max$ .

Существуют общие подходы к отображению графа, сочетающие в себе приведенные выше критерии. Это могут быть плоское и прямолинейное отображение, при котором ребра между вершинами проходят по кратчайшему пути без изгибов, или отображение с помощью ломаных линий, где ребра графа содержат в себе изгибы, или ортогональное отображение, при котором ребра могут содержать изгибы с углами  $90^\circ$  [4].

Наилучшую наглядность отображения обеспечивает именно подход ортогонального отображения графа. При применении данного подхода качество отображения прокладываемых маршрутов значительно превосходит результаты, получаемые при использовании других подходов.

Целью данного проекта являлось создание средства для визуализации сетей. Графическое представление сети, создаваемое программой, должно включать в себя области с изображениями обслуживающих устройств и области с рассчитываемыми маршрутами.

Обслуживающие устройства при отображении занимают фиксированное количество места, поэтому для резервирования области на экране и исключения пересечений устройств маршрутами было предложено создать сетку с четырьмя типами блоков. Блоки 1-го типа предназначены для размещения устройств, 2-й тип блоков используется для обозначения входных и выходных потоков, 3-й и 4-й типы являются диагональными и горизонтальными блоками соответственно (рис.1).

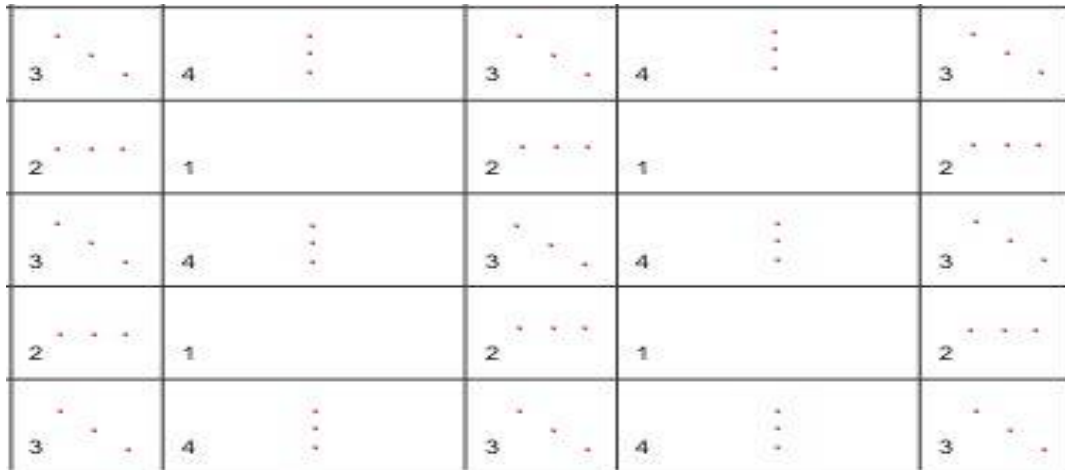


Рис. 1. Размещение элементов отображения сети на экране

Вершины графа представляют собой множество точек, через которые могут проходить маршруты. Каждая из вершин находится внутри блока для отображения маршрутов и связана с соседними вершинами таким образом, чтобы в результате не существовало маршрута из одной вершины блока в другую вершину того же блока.

Расстановка обслуживающих устройств на экране происходит следующим образом: начальные элементы маршрутов помещаются в левую часть экрана, конечные попадают в правую часть, а промежуточные располагаются по середине экрана. Расчет путей осуществляется с помощью алгоритма Дейкстры [5]. После выполнения расчета вес ребер, связанных с вершинами маршрута в диагональных блоках, увеличивается для предотвращения наложений путей и улучшения качества отображения (рис.2).

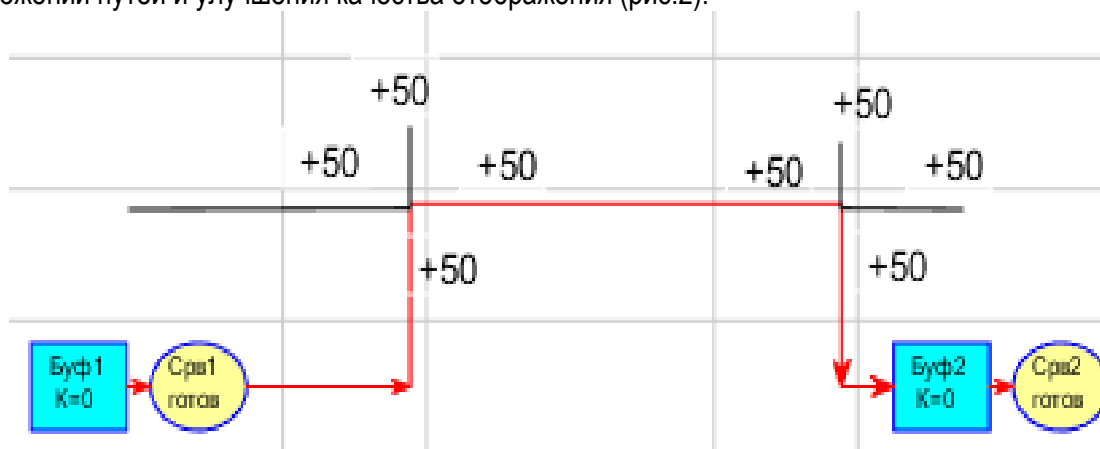


Рис. 2. Увеличение веса ребер, связанных с вершинами маршрута в диагональных блоках

Для оценки результата отображения сети был создан тестовый список маршрутов и построены ручное (рис.3) и компьютерное (рис.4) отображения сети.

Список маршрутов сети:

- 1→3→4→6→7→9;
- 2→3→5→8;
- 2→5→9→10;
- 5→7→9→11;
- 3→9→12;
- 13→15→16;
- 14→16;
- 15→17;

Сравнивая полученные отображения можно сделать вывод, что расположение обслуживающих устройств на экране схоже, а вот по количеству пересечений путей компьютерное отображение уступает ручному. Связано это с отличиями в начальном расположении обслуживающих устройств на экране.

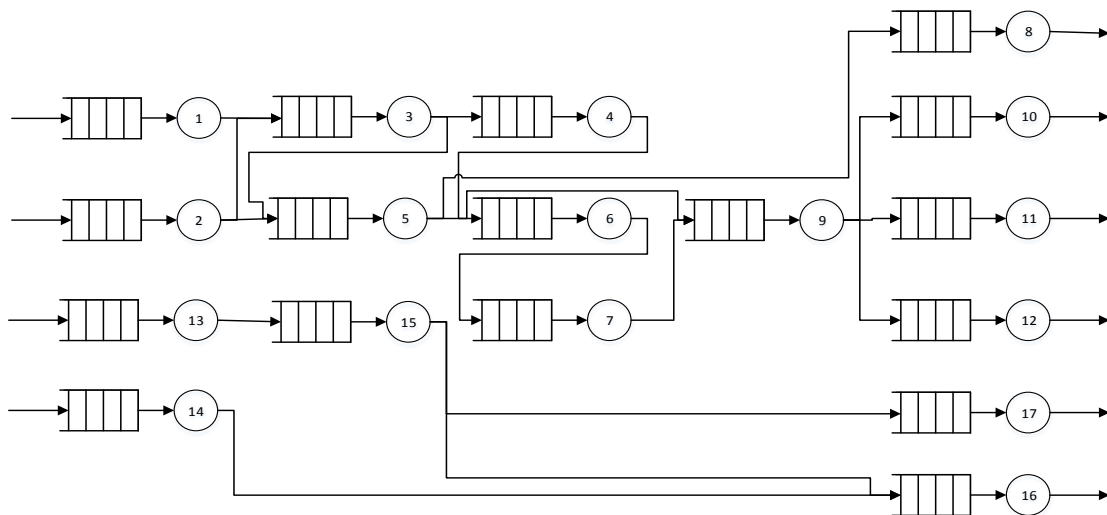


Рис. 3. Ручное и компьютерное отображение сети

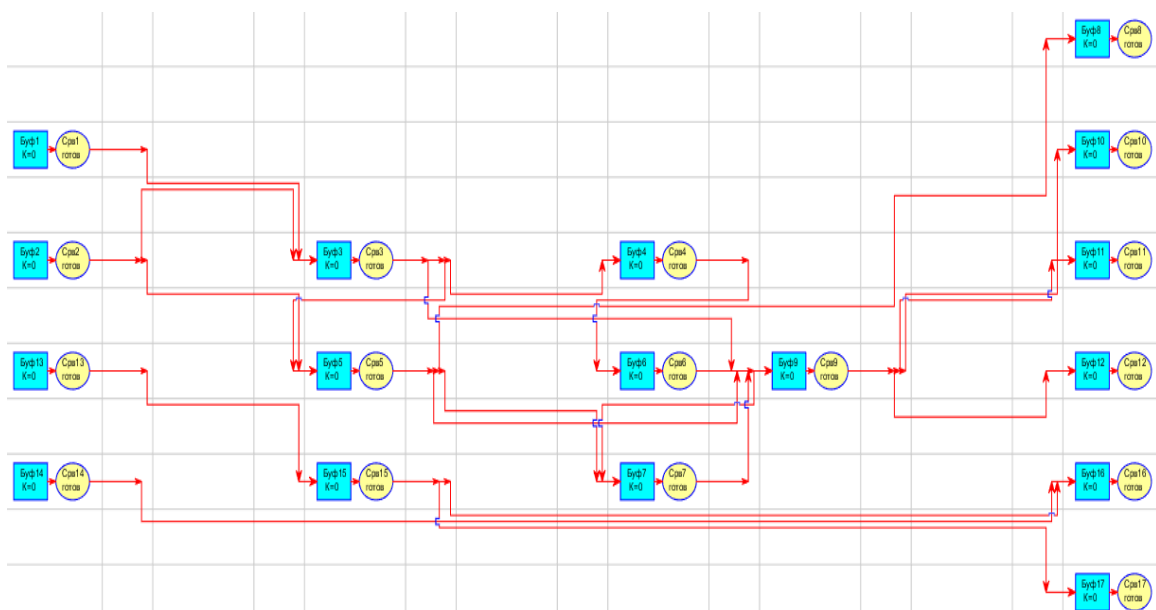


Рис. 4. Компьютерное отображение сети

Безусловно, алгоритмы отображения подлежат доработке, но уже сейчас программа позволяет существенно сократить время на создание графического представления сетей.

Результатами данной работы можно считать следующие положения:

1. Разработанное программное средство может быть использовано в качестве инструмента для автоматизации процесса визуализации сети массового обслуживания.
2. Начальная расстановка начинает сильно влиять на качество отображения при увеличении количества обслуживаемых устройств.
3. В дальнейших исследованиях планируется вывести формулу для оптимизации эстетичности отображения (например, графа) и расширить функционал программы улучшением алгоритма автоматического расчета расположения систем массового обслуживания на экране.

#### Библиографический список

1. Применение микропроцессорных средств в системах передачи информации : учеб. пособие для вузов / Советов Б.Я [и др.]. М. : Высшая школа, 1987. 255 с.
2. Клейнрок Л. Теория массового обслуживания / Л. Клейнрок. М.: Машиностроение, 1979. 432 с.
3. Бочаров П.П. Теория массового обслуживания / П.П. Бочаров, А.В. Печинкин. М. : РУДН, 1995. 530 с.
4. Di Battista G., *Graph Drawing: Algorithms for the Visualization of Graphs* / G. Di Battista, Prentice-Hall, 1999. 397 p.
5. Оре О. Теория графов / О. Оре. М. : Наука, 1968. – 382 с.

УДК 004.056.53

**А. И. Синяев**

студент кафедры компьютерных технологий и программной инженерии

**М. Д. Поляк**

ст. преп. – научный руководитель

### ВОЗВРАТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ И ОБХОД РАНДОМИЗАЦИИ АДРЕСНОГО ПРОСТРАНСТВА

ASLR, Address Space Layout Randomization, или так называемая рандомизация адресного пространства – механизм обеспечения безопасности, который включает в себя рандомизацию виртуальных адресов памяти различных структур данных, чувствительных к атакам [1, 2]. Расположение в памяти целевой структуры сложно предугадать, поэтому шансы атакующего на успех малы.

Зачем придумали ASLR? Чтобы ответить на этот вопрос начнем с простой уязвимости вида переполнения стека.

Переполнение буфера – это уязвимость, основанная на ошибке в функции, когда функция не проверяет (или неверно проверяет) границы при записи в буфер данных, полученных от пользователя. Излишние данные попадают в память где расположены другие переменные, к примеру адрес возврата. Если адрес возврата будет перезаписан, то при возврате из функции управление будет передано по вновь записанному адресу. Частным случаем является переполнение стека (буфер расположен в стеке). К сожалению, данная уязвимость до сих пор встречается в популярных библиотеках, примером может послужить уязвимость `glibc` в функции `getaddrinfo()` (CVE-2015-7547) [3].

Для начала введем определение “эксплоит”. Эксплоит (англ. exploit, эксплуатировать) – компьютерная программа, фрагмент программного кода или последовательность команд, использующие уязвимости в программном обеспечении и применяемые для проведения атаки на вычислительную систему.



Современные эксплойты основаны на возвратно-ориентированном программировании (Return-Oriented Programming, ROP) [4]. Задача ROP – построить последовательность инструкций, чтобы выполнить полезную нагрузку, основываясь на существующих фрагментах кода (гаджетах). Конечно, внедрение кода больше невозможно, поскольку память, в которую можно писать, не может исполняться (NX, NotExecutable).

Далее shell-кодом будем называть последовательность команд, с помощью которых уже запущенная программа может быть принуждена к выполнению чего-то альтернативного. Обычный shell-код представлен ниже:

```

– dup2(s, 0);
– dup2(s, 1);
– dup2(s, 2);
– execve("/bin/sh", 0, 0);

```

Его ROP-версия показана на рис. 1.

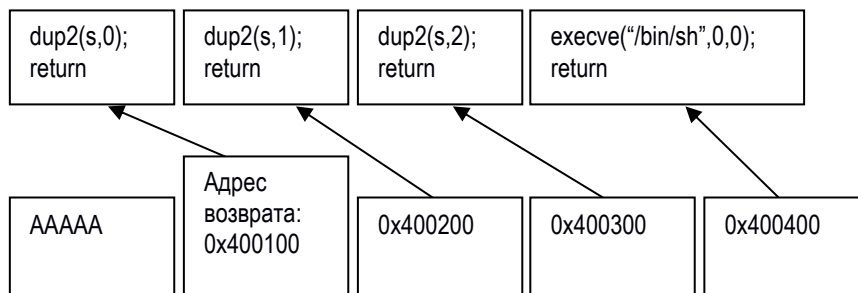


Рис. 1. ROP shell-код, который выполняет запуск /bin/sh на сокете s

На практике каждый гаджет – это короткая последовательность машинных инструкций, оканчивающихся возвратом. Чтобы исполнить простой системный вызов, такой как dup2(), потребуется несколько гаджетов, так как нужно еще занести аргументы в регистры. Пример ROP shell-кода для исполнения dup2() на рис. 2. Регистры rdi и rsi содержат первые 2 аргумента к системному вызову, rax содержит номер системного вызова. Регистры могут контролироваться rop-гаджетами при помещении загрузочных данных на стек. Присоединив достаточно гаджетов, можно получить готовый shell-код.

На большинстве современных операционных систем, где включены NX и ASLR, чтобы получить полный доступ к выполнению программы атакующему требуется:

- 1) обезвредить NX. Атакующий должен знать, где расположены гаджеты внутри исполняемого файла;
- 2) обезвредить ASLR. Атакующий должен распознать расположение сегмента кода исполняемого файла, который в данный момент загружен в память.

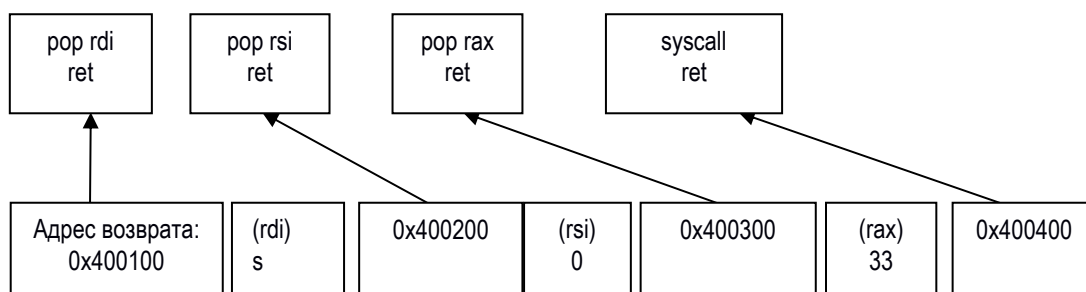


Рис. 2. ROP shell-код для выполнения dup2(s,0).  
Загрузка аргументов осуществляется rop-гаджетами

Оба эти требования решаются простым перебором на 32-битных системах методом опробования [5, 6]. Но это не относится к 64-битным системам. Как факт, большинство

публичных эксплоитов нацелены только на 32-битные системы. Задача техники «чтения» стека – обойти эти два ограничения для 64-битных систем.

Первое, что необходимо сделать для обхода ограничений в 64 битной ОС – атакующий должен получить копию уязвимого бинарного файла, дизассемблировать его и найти гаджеты.

Для обхода ASLR есть несколько известных стратегий. Во-первых, адрес расположения кода можно получить благодаря утечке информации. Во-вторых, иногда бывает так, что можно использовать какой-нибудь код, который не изменяет свое положение. Например, для Linux обычно, что исполняемый код располагается по строго установленному адресу, даже когда динамические библиотеки и секции данных грузятся по случайным адресам. В итоге исследователь может сделать ROP прямо на секцию кода. Вдобавок, в некоторых операционных системах, таких как Windows, несовместимые с ASLR библиотеки загружаются в память по фиксированным адресам.

На Linux можно применить ASLR к всему адресному пространству, включив PIE (опция -pie для gcc). Position Independent Executables – механизм защиты, который обеспечивает загрузку кода по случайным адресам. Нет известных техник обхода ASLR, когда включен PIE для 64-битной версии программы. Как быть в этом случае?

Некоторые атакующие часто встречаются с тем, что серверные приложения перезапускают процессы после сбоя автоматически. Таким образом, существует много ситуаций, когда атакующий имеет потенциально бесконечное число попыток (пока его не заметят).

Техника «чтения» стека использует описанный выше факт и задает условие: после сбоя серверное приложение не рандомизирует адресное пространство. Данная атака основывается на записи байт за байтом возможных значений, пока найдено правильное значение байта и не произошел сбой работы в сервисе. На рис. 3 показано типичное расположение стека. Canary – это известное значение, помещенное между буфером и управляющей информацией (адрес возврата) [7]. Canary служит для определения переполнения стека.

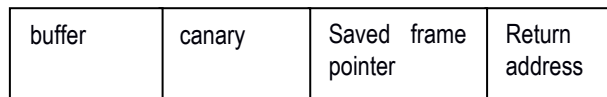


Рис. 3. Типичная раскладка стека

Техника «чтения» стека позволяет обезвредить ASLR для 64-битных систем. На рис. 4 по байту подбирается canary, в среднем получается 128 попыток на байт. Таким же образом мы можем найти в стеке адрес возврата. Есть шансы найти адреса на подгружаемые библиотеки.

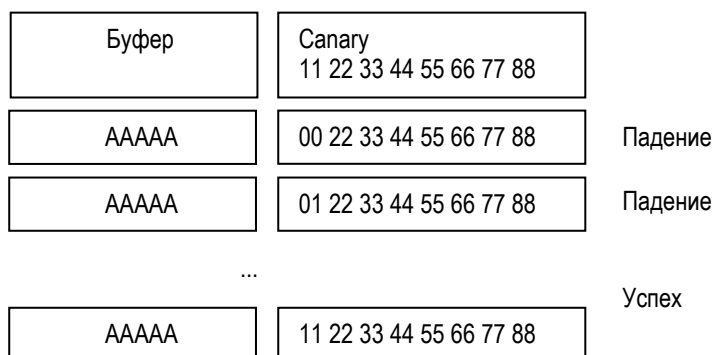


Рис. 4. Иллюстрация техники «чтения» стека

Получается, что и ASLR и NX были обезврежены, и эксплоит может быть создан на основе известных уязвимостей. Показанная техника гораздо эффективнее, чем прямой перебор.

Методом защиты от «чтения» стека может послужить рандомизация адресного пространства после сбоя работы процесса. Также после каждого падения сервиса можно

генерировать новое значение `sanary`, что не позволит прочитать управляющую информацию и сделает невозможным создание эксплоита.

В заключение можно сказать, что компьютерная безопасность – быстро развивающаяся отрасль и то, что кажется безопасным сейчас, потенциально уязвимо в будущем. В данной статье была рассмотрена стратегия создания эксплоитов в наши дни, которые основаны на возвратно-ориентированном программировании. Даже полная рандомизация адресного пространства (PIE) не всегда обеспечивает требуемый уровень надежности. И в некоторых случаях атакующий может успешно обезвредить ASLR для 64-битной системы.

#### **Библиографический список:**

1. Team P., "Pax address space layout randomization (aslr)", <http://pax.grsecurity.net/docs/aslr.txt>.
2. Bhatkar S., DuVarney D. C., Sekar R., "Address obfuscation: an efficient approach to combat a board range of memory error exploits", <http://home.eng.iastate.edu/~daji/seminar/papers/BVS03.UsenixSec.pdf>.
3. Serna F-J., Stadmeer K., "CVE-2015-7547: glibc getaddrinfo stack-based buffer overflow", 2016, <https://security.googleblog.com/2016/02/cve-2015-7547-glibc-getaddrinfo-stack.html>.
4. Roemer R., Buchanan E., Shacham H., Savage S., "Return-oriented programming: Systems, languages, and applications", 2012, <https://cseweb.ucsd.edu/~hovav/dist/rop.pdf>.
5. Shacham H., Page M., Pfaff B., Goh E.-J., Modadugu N., Boneh D., "On the effectiveness of address-space randomization", 2004, <http://benpfaff.org/papers/asrandom.pdf>.
6. Kingcope. "About a generic way to exploit linux targets", <http://www.exploit-db.com/wp-content/themes/exploit/docs/27074.pdf>.
7. Cowan C., Pu C., Maier D., Hintony H., Walpole J., Bakke P., Beattie S., Grier A., Wagle P., Zhang Q., "Stackguard: automatic adaptive detection and prevention of buffer-overflow attacks", 1998, [https://www.usenix.org/legacy/publications/library/proceedings/sec98/full\\_papers/cowan/cowan.pdf](https://www.usenix.org/legacy/publications/library/proceedings/sec98/full_papers/cowan/cowan.pdf).

**УДК 004.056.53**

**Д. А. Смолиенко**

студентка кафедры компьютерных технологий и программной инженерии.

**В. В. Бураков**

д-р. техн. наук, доц., профессор. – научный руководитель

### **ОРГАНИЗАЦИЯ ЗАЩИТЫ ОТ НСД ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА**

В небольших организациях с увеличением роста производства и увеличением штата встает вопрос о реорганизации документооборота на предприятии. Самые известные и дорогие решения это Microsoft, EMC, 1С и т.д. Но есть и альтернатива закрытым решениям – система управления документами с открытым исходным кодом Alfresco. Преимуществом открытой системы является возможность настройки и создания программы в зависимости от нужд компании. Именно на основе Alfresco была создана по заказу ГУ ГШ система электронного документооборота.

Хотя сам документооборот может использоваться как отдельная полноценная программа, он задумывался как один из модулей комплекса программ (далее Парсек). Еще одним модулем, отвечающим за централизованное создание пользователей и групп, учет ресурсов и пр., является «Центр безопасности». Если в Парсек необходимо добавить нового пользователя или группу, то эти объекты создаются вначале в «Центре безопасности» и потом переносятся в документооборот. Документы и пользователи представляют собой объекты и субъекты соответственно, задачей «Центра безопасности» является защита первых от вторых. Права пользователя не должны проверяться на каждое действие пользователя с документом, но необходимо дать возможность вмешиваться в процесс и запрещать то или иное действие или же

разрешать. Таким образом, этим двум модулям необходимо дать возможность обмениваться запросами и организовать защиту от НСД.

Обычно в качестве средств защиты данных от НСД используют шифрование, ЭЦП, мандатный и дискреционный разграничения доступа, средства доверенной загрузки, так как по требованиям регуляторов для шифрования необходимо использовать специальные платы (т.е. недостаточно шифровать только программными средствами), которые в несколько раз увеличивают стоимость компьютера. Защита данных от НСД в этом проекте основана на комбинации аппаратных средств доверенной загрузки, ЭЦП, программных средств мандатного и дискреционного разграничения доступа, которая гарантирует выполнение параметров защиты, требуемых соответствующими российскими регуляторами (ФСБ, ФСКН, МО РФ).

С учетом вышеперечисленных технологий был разработан межпрограммный интерфейс посредством Http запросов. На данный момент «Центр безопасности» способен отправлять следующие запросы:

- добавление, удаление, изменение пользователей;
- добавление и удаление групп;
- просмотр информации о пользователях.

Парсек способен отправлять запрос на регистрацию ресурсов.

При выполнении этих запросов каждый модуль должен авторизоваться и получить специальную метку – ticket. Эта метка действует во время некоторой сессии и затем необходимо получить новую. Метка помещается в качестве отдельного поля в каждый запрос. Рассмотрим механизм авторизации и получения метки более подробно.

При запросе пользователя какого-либо ресурса «Парсека», клиент запрашивает данные аутентификации у специальной службы, которая связана с блоком «Ключ-Л2» – сторонней программы, отвечающей за эти данные. Сюда входит и мандатная метка определенного уровня: «секретно», «совершенно секретно», которая сравнивается с меткой ресурсов и если они совпадают, значит пользователь может получить доступ к этому ресурсу. Затем, эти данные необходимо подписать с помощью электронно-цифровой подписи, для этого служба обращается в библиотеку ЭП для получения ключа и осуществляется подпись. Каждому пользователю выдается специальная флешка, с хранящемся на ней ключом, который используется для подписи. Далее эти данные возвращаются на клиент и уже подписанные отсылаются на сервер приложений. На сервере происходит обратная операция, то есть сервер отправляет полученную информацию на сервер распределения сертификатов и, если проверка прошла успешно, генерируется ключ сессии и отправляется *обратно* клиенту (рисунок).

Также интересен механизм входа пользователя в систему. На данный момент для входа пользователя в систему документооборота имеется два варианта действий: первый, когда он используется самостоятельно, в этом случае пользователю необходимы стандартные логин и пароль, которые он вводит на страницу входа в систему, а ключ на флешке используется лишь для подписания документов. Во втором случае, когда документооборот входит в состав комплекса программ, то страница входа не используется. Служба логина сама определяет пользователя, с помощью ключа подписывается логин через службу подписи, и эта подпись используется как пароль. Данный механизм является отличительной особенностью описываемой программы.

Таким образом, для организации связи между «Центром безопасности» и «Парсек» была поставлена задача доработки программного интерфейса каждого из модулей и реорганизации структуры данных. В дальнейшем перечень запросов можно расширить. Легкая модернизация, расширяемость, и механизмы защиты делают его мощным и доступным средством.



**Библиографический список**

1. Bhandari Amita, Majmudar Pallika. Share Alfrasco. Packt Publishing, 2012. 360 с.
2. Нечаев В. И. Элементы криптографии (Основы теории защиты информации). М.: Высшая школа, 1999. 109 с.

УДК 004.932

**А. С. Сысоева**

студентка кафедры компьютерной математики и программирования

**С. В. Щекин**

доц., канд. техн. наук, доц.– научный руководитель

**ПРИМЕНЕНИЕ МЕТОДОВ И АЛГОРИТМОВ КОМПЬЮТЕРНОЙ ГРАФИКИ  
ДЛЯ ОБРАБОТКИ И ВИЗУАЛИЗАЦИИ ДАННЫХ ДИСТАНЦИОННОГО  
ЗОНДИРОВАНИЯ ЗЕМЛИ**

Актуальность обработки пространственных снимков обусловлена возможностью их применения для решения задач, в которых невозможны наземные методы исследования или требуется обработка данных больших площадей наземной поверхности. В число этих задач входит принятие решений при чрезвычайных ситуациях, для территориального планирования, в

градостроительстве, для расчета навигационных маршрутов, для разработки новых методов поиска полезных ископаемых, для изучения земной коры, в том числе, составление карт.

Аэрокосмический снимок – это двумерное изображение реальных объектов, которое получено по определенным геометрическим и радиометрическим (фотометрическим) законам путем дистанционной регистрации яркости объектов и предназначено для исследования видимых и скрытых объектов, явлений и процессов окружающего мира, а также для определения их пространственного положения.

Космическая съемка различается по:

- масштабам;
- пространственному разрешению;
- обзорности;
- спектральным характеристикам.

Эти параметры определяют возможности дешифрирования на космических снимках различных объектов и решения тех геологических задач, которые целесообразно решать с их помощью.

Этапы обработки снимков: их дешифрирование, фотограмметрическая обработка, а также составление по снимкам карты, геоинформационной системы, разработка прогноза. Важным элементом исследований по снимкам является также оценка достоверности и точности полученных результатов.

Полученное аэрокосмическое изображение может отличаться в зависимости от положения солнца относительно снимаемого участка земной поверхности в процессе съемки. При обработке снимка могут потребоваться данные о рельефе, которые недостаточно ярко выражены на снимке при определенном азимуте солнца. Так же, может возникнуть необходимость получения данных уже отснятого участка поверхности при другом азимуте солнца, например в задачах, требующих сравнения данных, полученных с изображений разных участков земной поверхности. В подобных случаях возможно применения метода теневой отмывки рельефа.

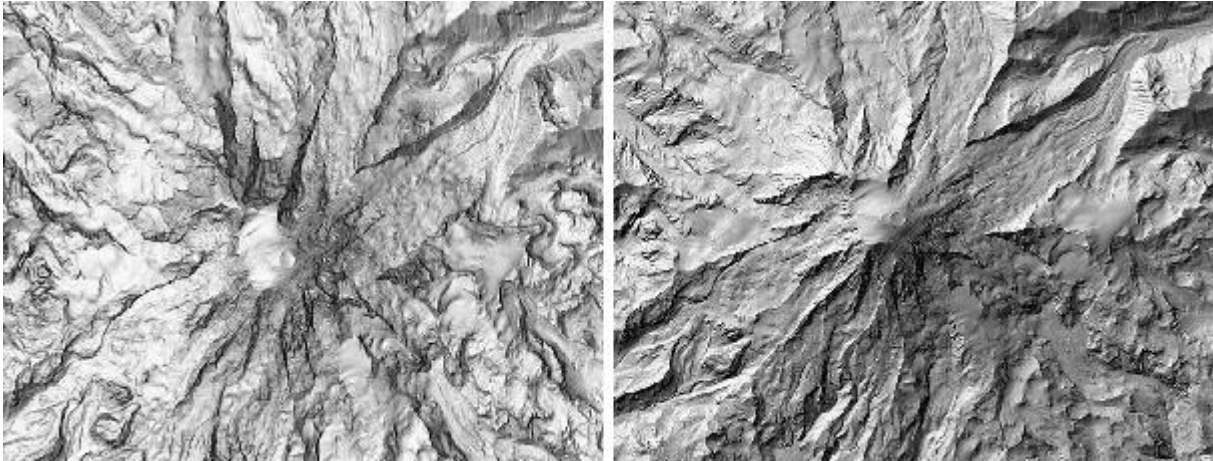
Отмывка рельефа – способ отображения рельефа местности методом теневой пластики при косом, реже – отвесном освещении. Также отмывкой называют пластическое полутонное изображение рельефа путем наложения теней. Отмывка создает иллюзию выпуклости изображенного рельефа (рисунок).

Обычно отмывку применяют при:

- боковом освещении, когда источник света находится в левом верхнем углу карты;
- отвесном освещении, когда свет падает сверху;
- комбинированном освещении, когда местность считается освещенной с разных сторон.

Отмывка рельефа позволяет получить гипотетическое освещение поверхности путем определения освещенности для каждой ячейки растра. Принцип его работы заключается в установке положения гипотетического источника света и вычислении значений освещенности каждой ячейки по отношению к соседним ячейкам. Применение функции может значительно улучшить визуальное представление поверхности для анализа и графического отображения, особенно при использовании опции прозрачности.

Модификация данного алгоритма позволит устранять ошибки на снимках и проводить их модификацию, например, улучшать качество снимков в местах, на которых при съемке не всегда акцентируют внимание (обширные лесные области, океаны). Так же возможно изменение некоторой метаинформации о снимке (под метаинформацией понимается информация о методах и способах обработки информации о/на снимке, а также местоположение данной информации). Например, в разрабатываемое ПО планируется встроить возможность изменения азимута солнца на снимке для удобства дальнейшей обработки.



Результат обработки изображения алгоритмом отмывки рельефа

Для обработки аэрокосмических изображений существует множество различных программ, обладающих многочисленными возможностями и инструментами обработки пространственных данных, например:

1. ArcGIS. Данная платформа предоставляет широкий спектр приложений, поддерживающих решение геоинформационных задач, в том числе картографию, построение моделей и приложений. Поскольку данное ПО специализируется на построении различных объектов на основе координатной геометрии (COGO) пространственных снимков, ему необходима высокая точность масштабирования объектов на снимке на всех его участках. Для реализации данной возможности в нем присутствует коррекция направления азимута и расстояния от съемочного аппарата до различных участков снимка. В данном алгоритме так же, как и в нашем случае используется азимут, другие параметры, также как источники информации о корректировке;

2. Программный комплекс ENVI. Отличительной особенностью ПК является его открытая архитектура и модульная структура, позволяющая приобретать только необходимые функциональные модули. В базовой комплектации присутствует возможность геометрического преобразования изображений (географическая привязка, создание бесшовной мозаики на основе серии снимков), классификации изображений, постклассификационной коррекции. Дополнительные модули включают в себя:

- 2.1. Модуль атмосферной коррекции, устраняющий влияние атмосферных явлений.
- 2.2. Модуль создания высокоточных моделей рельефа.
- 2.3. Модуль Выявления объектов на панхроматических и мультиспектральных изображениях.
- 2.4. Модуль чтения/записи изображений формата NITF.
- 2.5. Модуль ортотрансформирования изображений с использованием аппроксимационных RPC-коэффициентов.

3. ERDAS Imagine. Распространяется в трех комплектациях:

3.1. IMAGINE Essentials – базовая версия, содержащая базовые инструменты для визуализации, коррекции, составления карт.

3.2. IMAGINE Advantage имеет расширенные возможности спектральной обработки, анализа измерений, ортоторрекции, мозаики, анализа изображений;

3.3. IMAGINE Professional предлагает еще более широкий набор инструментов для обработки спектральных, гиперспектральных и радиолокационных данных, а также пространственного моделирования.

В данном программном продукте имеется возможность вычисления начального азимута между точками для устранения неточностей в построении моделей и карт на основе пространственных изображений.

Как видно, существующие программные комплексы, предназначенные для обработки пространственных снимков, выполняют крупные задачи, в особенности, в области построения карты моделей. Корректировка же изображений с использованием информации об освещении производится только как сопутствующая часть более крупных задач.

Исходя из вышесказанного, очевидна актуальность разработки более узкоспециализированного ПО, с целью обеспечения более широких, чем у аналогов, возможностей по обработке изображений, корректирования их качества на основе освещения, а так же изменения азимута.

**УДК 004.932, 004.8**

***А. Р. Сытин***

студент кафедры компьютерных технологий и программной инженерии

***С. А. Рогачев***

ассистент – научный руководитель

### **АНАЛИЗ КРИТЕРИЕВ КАЧЕСТВА РАБОТЫ НЕЙРОННОЙ СЕТИ В ЗАДАЧАХ КЛАССИФИКАЦИИ**

В рамках выпускной квалификационной работы бакалавра было разработано ПО, осуществляющее классификацию пространственного изображения на основе радиальной базисной нейронной сети [1]. Данное ПО позволяло успешно обучать нейронную сеть, и на её основе проводить классификацию.

Радиальная базисная нейронная сеть, как и другие нейронные сети, имеет ряд настроек, существенно влияющих на результат классификации. Применительно к данной сети настраиваемыми параметрами являются:

- Скорость обучения;
- Максимальное смещение;
- Связи между нейронами;
- Допустимый диапазон масштаба функции;
- Количество входов;
- Количество нейронов скрытого слоя;
- Количество классов;
- Количество эпох обучения.

Разработанное ПО имело существенный недостаток, сказывающийся на качестве классификации. В процессе обучения сети имел место ручной подбор перечисленных выше параметров, и для достижения удовлетворительного результата эти параметры приходилось подбирать эмпирически (рис. 1).

Корректировка параметров происходит на основе знаний об их влиянии на качество классификации. Например, плохо обученный алгоритм будет показывать одинаково плохие результаты как на тестовом, так и на обучающем множествах, а переобученный будет демонстрировать отличный результат на обучающем и плохой на тестовых множествах [2]. Однако, человеческий фактор не позволяет подобрать данные параметры достаточно точно, в то время как автоматизированная система теоретически может подобрать намного более точные и подходящие для качественной классификации параметры сети.



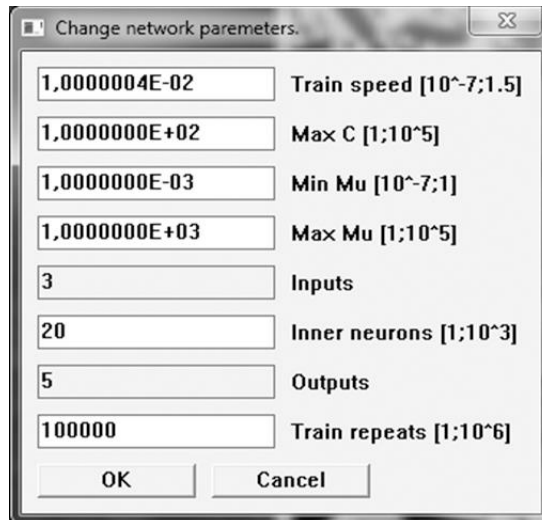


Рис. 1. Окно эмпирически подобранных параметров сети

Как следствие, нельзя утверждать, что полученное в результате классификации изображение является наилучшим (рис. 2).

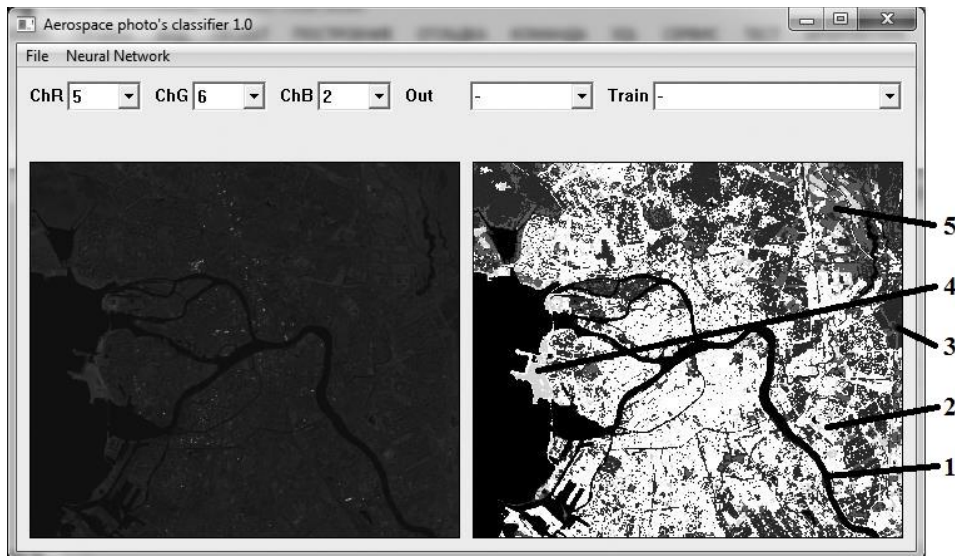


Рис. 2. Результат классификации снимка спутника Landsat 8 в комбинации 5,6,2:  
1 – вода, 2 – антропогенные объекты, 3 – густая растительность, 4 – песчаные поверхности, 5 – редкая растительность

Как видно из таблицы, результат классификации является неточным при некоторых значениях, вследствие чего появилась необходимость интеграции в ПО модуля, осуществляющего проверку результата и корректировку настроек сети.

Проверка качества классификации эталонной выборкой

Класс объекта	спешно классифицированные эталонные точки, %	Наименьшая точность успешной классификации. %
Водные объекты	100%	82.93%
Густая растительность	100%	76.42%
Антропогенные объекты	90%	60%
Редкая растительность	100%	45.3%
Песчаные поверхности	90%	62.91%

Если считать, что обучающие и тестовые выборки корректны и не содержат ошибок, схема модифицированного алгоритма классификации будет выглядеть следующим образом (Рис. 3).

Как видно из схемы, в новом алгоритме, пользователю не придется проверять результирующее изображение на корректность. По окончании работы алгоритма ему будет предоставлен наилучший, с точки зрения проверки на качество по определенному критерию.

Процесс выбора конечной сети может быть остановлен либо путем выбора наилучшего варианта, либо превышением определенного порога значения коэффициента качества работы сети. В обоих подходах наибольшее значение имеют критерии оценки качества классификации.

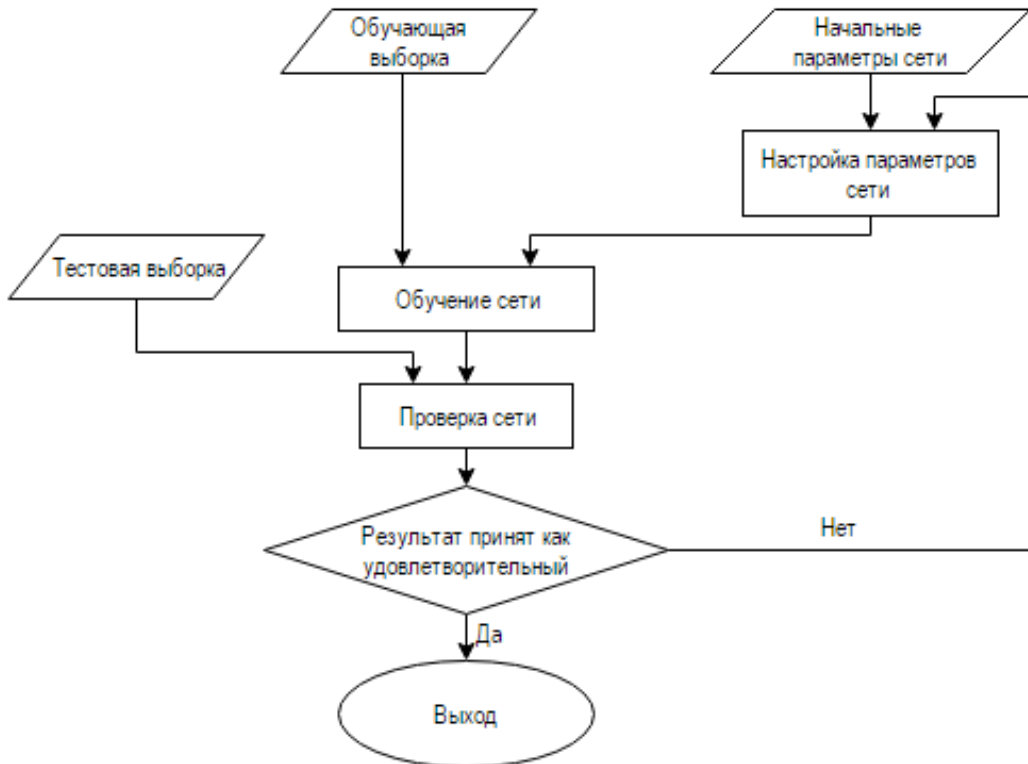


Рис. 3. Автоматизированный алгоритм обучения и классификации

Поскольку нейронная сеть является классификатором, к ней применимы меры расчета эффективности. Эти меры базируются на четырех метриках:

- истинный положительный результат (TP). Верный набор данных, классифицированный как верный;
- истинный отрицательный результат (TN). Неверный набор данных, классифицированный как неверный;
- ложный положительный результат (FP). Верный набор данных, классифицированный как неверный;
- ложный отрицательный результат (FN). Неверный набор данных, классифицированный как неверный.

Меры расчета эффективности классификатора [4]:

- чувствительность связана со способностью выявлять положительные результаты –

$$Sensitivity = \frac{TP}{TP + FN};$$

- специфичность связана с возможностью выявлять отрицательные результаты –

$$Specificity = \frac{TN}{TN + FP};$$

– идеальная классификация отражает способность правильно идентифицировать данные –

$$Perfect\ classification = \frac{TP + TN}{K},$$

где  $K$  – общее число входных данных.

– пропущенная классификация – показатель, обратный идеальной классификации –

$$Missed\ classification = \frac{TN + FP}{K};$$

– индекс эффективности является мерой производительности классификатора:

$$PI = \frac{PC - MC - FA}{PC} 100,$$

где  $PC$  – идеальная классификация,  $MC$  – пропущенная классификация,  $FA$  – ложное срабатывание.

Для классификатора можно выделить качественные метрики, показывающие достоверность классификатора как оптимизатора или мультиклассового классификатора[3].

Показатель качества:

$$Qv = \frac{C}{(Rfa + 0.2) * (Pdet + 6 * Pmsd)}$$

где  $C$  является константой масштабирования и равняется 10,  $Rfa$  – число ложных срабатываний на набор данных;  $Pdet$  – процент совершенной классификации;  $Pmsd$  – процент пропущенных классификаций.

Мера стоимости определяется как фактор потери  $L$  в результате ошибки:

$$L = (False\ alarm + 2 * Missed\ Classification) * D,$$

где  $D$  – константа масштабирования, которую можно приравнять к 1.

На основе этих мер получаем показатель общей стоимости отбора:

$$C = (Sensitivity + Selectivity) * D + L.$$

В идеальном случае мера стоимости  $L = 0$  и общая стоимость отбора  $C = 2D$ .

На основе показателя общей стоимости отбора и показателя качества планируется дальнейшая модернизация алгоритма и его тестирование на различных пространственных снимках и комбинаций их спектральных каналов. Конкретные значения данных показателей можно будет получить только в процессе разработки. В случае успеха будет получен алгоритм, позволяющий уменьшить объем временных затрат на подготовку и проведение классификации, а также повысить качество классификации. Помимо этого, данную процедуру автоматизации будет возможно применять для других классификаторов, а в перспективе, проводить классификацию несколькими классификаторами с целью получения наиболее качественного результата.

### Библиографический список

1. Сьтин А.Р. Классификация аэрокосмических изображений с использованием радиальных базисных нейронных сетей // Шестьдесят восьмая международная студенческая научная конференция ГУАП: сб. докладов. СПб: ГУАП, 2015, с. 279–282.
2. Мухамедиев Р. И., Мухамедиева Е.Л., Кучин Я. И. Таксономия методов машинного обучения и оценка качества классификации и обучаемости // Электронный журнал Cloud of science. 2015. Т. 2, № 3. 13 с.
3. Hari Kumar Rajaguru, Vinoth Kumar. Bojan Performance Analysis of EM, SVD, and SVM Classifiers in Classification of Carcinogenic Regions of Medical Images // Journal International Journal of Imaging Systems and Technology. 2014. Vol. 24. No.1. P. 16–22.
4. Marina Sokolova, Guy Lapalme. A systematic analysis of performance measures for classification tasks // Journal Information Processing and Management: an International Journal. 2009. Vol. 45. No.4. P. 427-437.

УДК 004.912

**А. С. Толчий**

студент кафедры вычислительных систем и сетей

**Е. Л. Гребенникова**

асс. – научный руководитель

## **ВОЗМОЖНОСТИ ПРОГРАММНОЙ ИНТЕРПРЕТАЦИИ И ГЕНЕРАЦИИ ИНДИВИДУАЛЬНЫХ ОСОБЕННОСТЕЙ РУКОПИСНОГО ШРИФТА**

Рукописный шрифт – это шрифт, литеры которого выглядят как буквы, написанные от руки. Рукописные шрифты получили большое распространение в различных поздравлениях, приглашениях и оформлении надписей праздничных мероприятий [1]. Кроме того, они используются в областях, где требуется написать от руки много текста, например для написания писем или размножения конспектов лекций среди студентов [2]. Цель этой работы – выяснить, что необходимо для правдоподобной имитации рукописного письма, включая индивидуальные особенности человеческого почерка и узнать, каким образом можно решить эту задачу, и какие готовые решения существуют на сегодня.

При создании индивидуального рукописного шрифта приходится учитывать следующие особенности письма от руки:

- одни и те же символы могут выглядеть по-разному; чаще всего в тексте вообще не встречаются абсолютно одинаково написанные символы;
- начертание символов может меняться в зависимости от того, какие символы стоят рядом;
- буквы, находящиеся рядом, могут быть написаны слитно (не отрывая ручку от бумаги) или отдельно;
- высота, ширина и наклон могут меняться как у букв, так и у строк;
- если при написании текста была допущена ошибка, её нельзя исправить незаметно; чем больше текст, тем больше вероятность допустить ошибку;
- всё вышеперечисленное зависит от огромного количества факторов, включая манеру письма, особенности письменной принадлежности, подложку, скорость письма, преобладающую руку, настроение и самочувствие писца и т. д.

Для правдоподобной имитации рукописного письма необходимо создать шрифт с поддержкой псевдослучайного выбора глифов и лигатур, умеющего менять высоту, наклон, ширину букв и строк, вносить различные искажения в сгенерированный текст. На сегодняшний день такого формата шрифта не существует.

Наиболее часто рукописное письмо имитируют с помощью рукописных шрифтов формата TrueType или OpenType. Рассмотрим последний, как обладающий расширенными возможностями [3]:

- есть поддержка unicode;
- есть поддержка множества глифов для одного и того же символа;
- есть поддержка лигатур.

Это делает формат OpenType пригодным для небольших надписей. Существует много сервисов, предлагающих создать свой собственный шрифт OpenType. На основе заполненного пользователем шаблона они генерируют файл шрифта, который в дальнейшем можно использовать в текстовом процессоре [4]. Но, поскольку в формате шрифта OpenType нет возможности выбирать глифы и лигатуры псевдослучайным образом, при большом объеме текста могут бросаться в глаза абсолютно одинаково «написанные» символы и их сочетания [2][5].

Другой способ решения поставленной задачи – написание программного приложения, учитывающего особенности рукописного письма. На сегодняшний день уже существуют некоторые готовые решения; к наиболее функциональным и распространённым относятся программы Синяк [6], HandWriter [7] и сервис Писец [8]. Они ориентированы на учащихся в учебных заведениях стран СНГ, т. е. на генерацию текстов большого объёма.

У программы Синяк большие возможности: она имитирует слитное написание букв, применяет разные эффекты для скрытия факта имитации, в экспериментальном режиме предоставляет возможность создания пользовательского почерка, предоставляет возможности по настройке формата вывода. Эта программа бесплатна, она до сих пор развивается и поддерживается. При этом Синяк не поддерживает разные глифы для одного и того же символа, что сразу же выдаёт факт имитации письма на тексте большого объёма. К недостаткам этой программы также относятся крайне низкое качество интерфейса, неполная поддержка Unicode, низкая скорость работы, возможность сохранения результата только в виде изображений.

HandWriter – платное ПО, доступное как онлайн, так и в виде исполняемого файла. По сравнению с программой Синяк у неё гораздо меньше настроек, в том числе маскирующих эффектов, более низкое качество соединительных линий, нет возможности настройки формата вывода, ограниченный даже по сравнению с CP1251 набор символов. Зато, в отличие от Синяка, она поддерживает использование нескольких глифов для одного символа, а её маскирующие эффекты намного лучше компенсируют использование небольшого количества глифов (в демо-версии их всего два на один символ). В данный момент программа не развивается.

Сервис Писец поддерживает использование множества глифов для одного символа, предлагает три стандартных почерка, умеет переносить слова по слогам. При этом отсутствует возможность загрузки собственного почерка, нет имитации слитного письма, вывод доступен только в виде изображения первого листа текста, настроек формата вывода нет, Unicode поддерживается не полностью. Два года назад разработчик сообщил о планах исправить некоторые из упомянутых недостатков и добавить больше возможностей, однако с тех пор функциональность программы существенно не изменилась. В данный момент сервис находится в стадии бета тестирования; ведётся ли разработка новых функций, неизвестно.

Таким образом, существующие на сегодня решения обладают чередой существенных недостатков, которые затрудняют их использование и/или ограничивают область использования.

Проблема имитации индивидуальных особенностей рукописного письма является актуальной и интересной для того, чтобы попытаться решить её самостоятельно. В рамках дипломной работы было решено разработать программное приложение для имитации рукописного письма, которое вобрало бы в себя лучшие черты его аналогов.

На данный момент реализована поддержка множества глифов для одного и того же символа, возможность создания индивидуального шрифта пользователя, слитное написание слов, сохранение вывода в формате PDF или в виде изображений, полная поддержка Unicode, возможность печати результата прямо из программы, множество настроек шрифта и формата вывода [9]. Однако, здесь всё ещё нет модификаторов, скрывающих факт имитации при недостаточном количестве глифов, отсутствует поддержка лигатур. Кроме того, существуют проблемы с качеством имитации слитного письма и отображением пунктуационных знаков. Поскольку мои ресурсы не позволяют создать действительно универсальное, лишённое недостатков решение, программа и её исходные коды распространяются по лицензии MIT с тем, чтобы любой желающий мог внести свой вклад в этот проект.

#### Библиографический список

1. Рукописные русские шрифты – <http://www.xfont.ru/rusrukopisnye>
2. Создаём шрифт, имитирующий Ваш рукописный почерк / Хабрахабр – <https://habrahabr.ru/post/127088/>
3. OpenType - Wikipedia, the free encyclopedia – <https://en.wikipedia.org/wiki/OpenType>
4. Handwriting font creator – <http://www.myscriptfont.com>

5. Handwriting font that at random picks characters? - Graphic Design Stack Exchange – <https://graphicdesign.stackexchange.com/questions/26833/handwriting-font-that-at-random-picks-characters>
6. Dimlight Studio | Синяк – <http://dimlight.ru/projects/programs/sinyak.html>
7. Из печатного текста в рукописный – программа HandWriter – [http://dmtsoft.ru/un/hw\\_help\\_1](http://dmtsoft.ru/un/hw_help_1)
8. Писец 1.2a – генератор конспектов для ленивых студентов и не только – [http://coo.by/writer\\_dev](http://coo.by/writer_dev)
9. Руководство пользователя · aizenbit/Scribbler Wiki – <https://github.com/aizenbit/Scribbler/wiki/Руководство-пользователя>

**УДК 681.3.06**

**А.Ю. Фролова, Е.С. Шлюев**

студенты кафедры проблемно-ориентированных вычислительных комплексов

**Г.С. Евсеев**

канд. техн. наук, доцент – научный руководитель

**ОСОБЕННОСТИ МЕХАНИЗМА БЛОКИРОВОК В СЕТЕВОЙ ВЕРСИИ ACCESS**

Работа на изолированном компьютере с небольшой базой данных в настоящий момент становится нехарактерной для большинства приложений. БД (база данных) отражает информационную модель реальной предметной области, хранит большие объемы информации. В современных системах, как правило, должен быть обеспечен параллельный доступа пользователей к БД [1].

Сетевое использование СУБД Access

В современной версии приложение MS Access может использоваться как SQL – сервер в локальной сети. Отсутствие некоторых обычных для серверов функций (ведение журнала аудита и т.д.) компенсируется тем, что такой сервер проще использовать и легче администрировать.

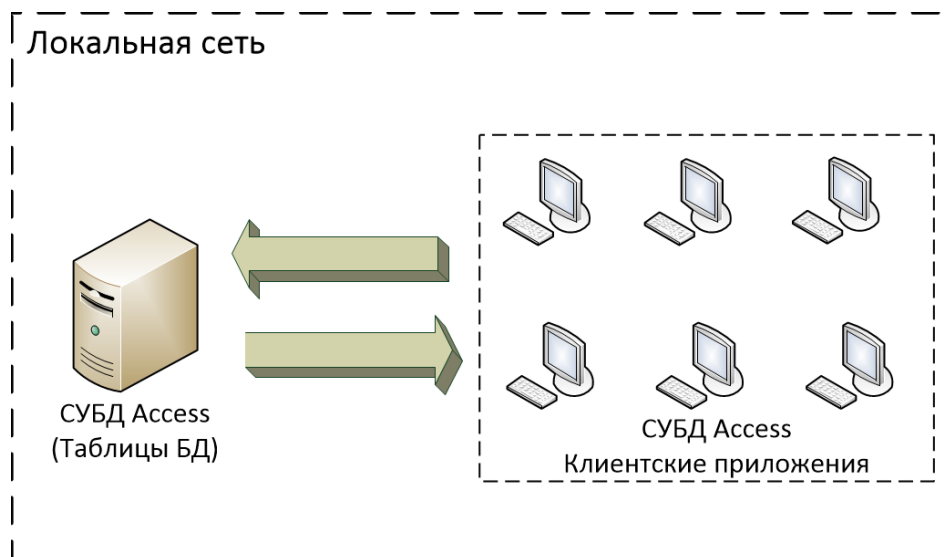


Рис. 1. Сетевое использование СУБД Access

На рис. 1 представлена структурная схема клиент-серверной системы на базе СУБД Access. В этой системе допускается одновременная работа нескольких пользователей БД. Известно, что при этом требуется использовать некоторый механизм блокировок, который исключает возможность нарушения целостности данных при параллельной работе пользователей. В рассматриваемой системе реализация механизма блокировок является специфической и отличается от механизма блокировок, описанного в стандарте языка SQL и

применяемого в популярных SQL-серверах. В этой работе представлено описание механизма блокировок и даны рекомендации по его использованию.

Установка режима блокировок

Существует три режима использования блокировок клиентами:

1. Блокировка отсутствует. Если два пользователя одновременно вносили изменения в запись, то тот, кто сохраняет изменения первым, может это сделать. Когда второй пользователь попытается сохранить свои изменения, то появляется диалоговое окно «Конфликт записи».
2. Блокировка изменяемой записи. Access блокирует изменяемую в данный момент запись, не позволяя изменять ее другим пользователям. Если другой пользователь попытается изменить заблокированную запись, у него появится маркер заблокированной записи.
3. Блокировка всех записей. Access блокирует все записи таблицы, поэтому другие пользователи не могут её открыть. Этот режим накладывает жесткие ограничения на параллельную работу и явно снижает производительность [2].

Использование блокировок по умолчанию

Каждый пользователь может выполнять разные по типу и по важности операции с таблицами БД. При этом некоторые операции будут требовать блокировки изменяемой записи, другие будут требовать блокировки таблицы или вообще не требовать блокировок. Следует учитывать, что даже блокировка записи может затормозить работу других пользователей. Еще больше тормозит работу пользователей блокировка таблицы. Поэтому блокировки желательно устанавливать на минимально возможное время, а их количество желательно минимизировать с целью увеличения общей производительности.

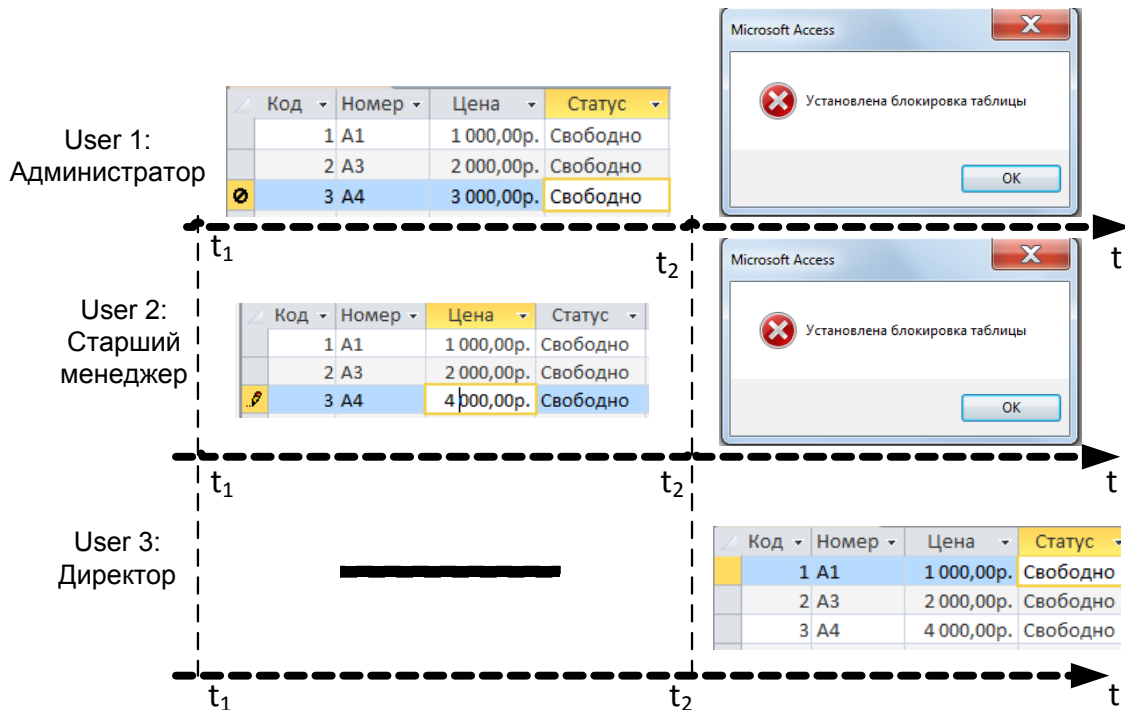


Рис. 2. Пример сценария работы с БД

Рис.2 иллюстрирует работу нескольких сотрудников в отеле. В момент времени  $t_1$  администратор выполняет продажу номера клиенту, в это время к БД подключился старший менеджер и повышает цены на определённые номера. Администратор не сможет продать номер A4, так как соответствующую запись заблокировал старший менеджер, у которого установлена

блокировка изменяемой записи. В момент времени  $t_2 < t_1$  таблицу открыл директор для того, чтобы сделать запрос на свободные номера в отеле. Открыть таблицу администратору и старшему менеджеру не удастся, потому что у директора установлена блокировка всей таблицы.

Программное управление блокировками

При программном управлении блокировками к запросу клиента предшествует инструкция, переключающая режим блокировок для этого запроса. По умолчанию (оптимистическая блокировка) используется режим блокировки изменяемой записи только в момент ее обновления, в остальное время она доступна для просмотра и изменения. Это обеспечивает высокую доступность данных при параллельной работе пользователей.

При использовании пессимистической блокировки редактируемая запись становится недоступной для других пользователей до завершения операции редактирования, т.е. её можно только просматривать, но не редактировать [3].

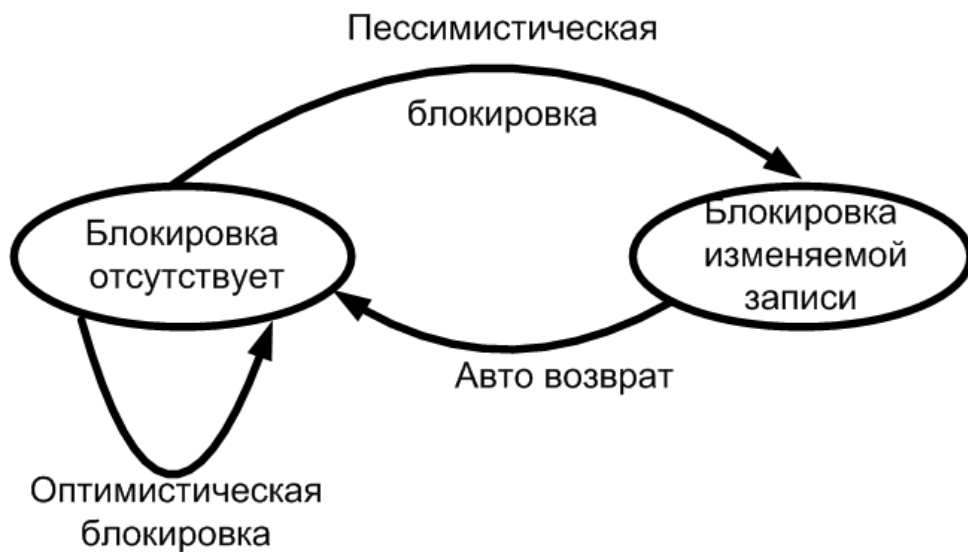


Рис. 3. Схема программных блокировок

На рис. 3 изображена схема работы программных блокировок. Изначально блокировка отсутствует, далее запускается хранимая процедура с пессимистической блокировкой, и происходит блокирование изменяемой записи. После выполнения хранимой процедуры происходит возврат в начальное состояние. При запуске хранимой процедуры с использованием оптимистической блокировки, блокируется запись лишь в момент ее фактического обновления, то есть, по сути, блокировка отсутствует.

Использование пессимистической блокировки

Рис. 4 иллюстрирует работу кассиров в параллельных кассах. Например, продажа ЖД билетов.

В момент времени  $t_1$ , предлагая клиенту билет, кассир 1 должен предварительно заблокировать первую запись от изменений, а разблокировать её либо в момент отказа от билета, либо после полного расчета с клиентом. Поэтому у кассира должен быть установлен режим блокировок «изменяемой записи». В момент времени  $t_2$  кассиры продают разные билеты и тем самым не мешают друг другу.

У любого пользователя могут быть операции, при которых заблокировать записи не нужно. Поэтому предусмотрена возможность для одних запросов использовать блокировки записей, а для других запросов не использовать, т. е. блокировки изменяемой записи включаются только на время выполнения определенной операции, но не всех.



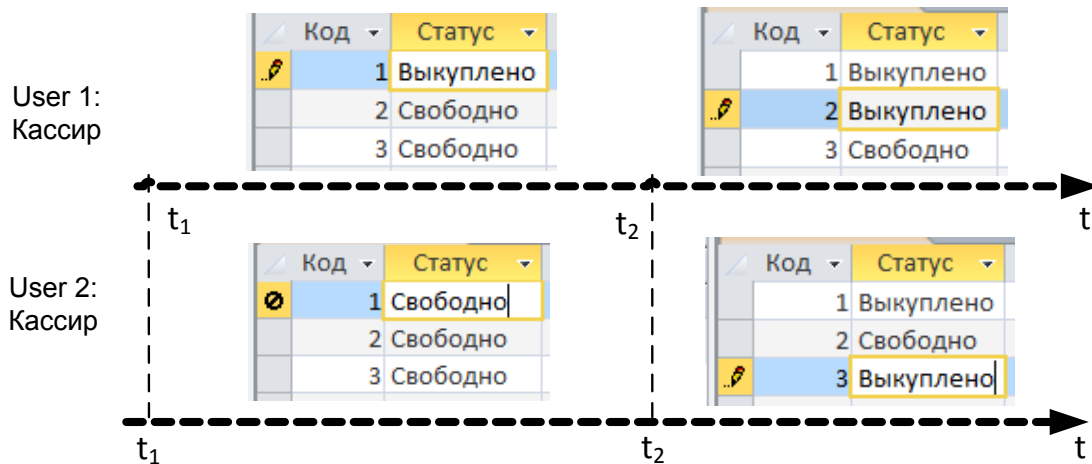


Рис. 4. Пример сценария работы с БД

**Заключение**

Механизм блокировок в Access хоть и упрощен по сравнению с MySQL, но позволяет совмещать параллельное выполнение запросов от разных пользователей, контролируя сохранение целостности данных. Если в БД имеется много разнородных транзакций, то этого механизма недостаточно, чтобы обеспечить качественный контроль целостности данных, т. е. СУБД Access желательно использовать при небольшом числе параллельных транзакций. Таким образом, сетевая версия Access может быть использована в качестве сервера во многих случаях, когда требуется быстро создать небольшую дешёвую информационную клиент-серверную систему на базе локальной вычислительной сети.

**Библиографический список:**

1. Слепцова Л.Д. Программирование на VBA. И.: Вильямс, 2004. 384 с.
2. Гандерлой М., Харкинз С. Автоматизация Microsoft Access с помощью VBA. И.: Вильямс, 2006. 416 с.
3. Аблязов В.И. Проектирование баз данных в среде Microsoft Access. И.: Вильямс, 2014. 104 с.

УДК 004.82

**И. А. Цветков**

магистрант кафедры компьютерной математики и программирования

**М. Ю. Охтилев**

д-р. техн. наук, проф. – научный руководитель

**МОДЕЛИ ПРЕДСТАВЛЕНИЯ ЗНАНИЙ, АЛГОРИТМЫ МАНИПУЛЯЦИИ  
ЗНАНИЯМИ И ИХ СПОЛЬЗОВАНИЕ В ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ  
ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ**

В настоящее время в связи с активным развитием техники, повышается сложность задач, возлагаемых на технические объекты, такие как космические аппараты, ракеты космического назначения, атомные электростанции, гидроэлектростанции и другие. При эксплуатации таких объектов неизбежно возникает вопрос о контроле их технического состояния, возрастает ответственность за принятые решения, при одновременном сокращении сроков на принятие этих решений [1-4]. Это обусловлено возрастанием потоков информации, поступающей в соответствующие ситуационные центры, усилением жесткости различных ограничений на принятие решений, а также сокращением расходов на разработку, эксплуатацию технических или

иных объектов. Таким образом, при эксплуатации сложных технических объектов возникает проблема оценивания его состояния, анализа и прогнозирования возможных штатных и внештатных ситуаций.

Для решения задачи контроля технического состояния используются автоматизированные системы мониторинга надежности и безопасности сложных технических объектов. Они предназначены для сбора, анализа и накопления данных от разнотипных и распределенных датчиков с целью определения технического состояния оборудования, обнаружения отклонений и неисправностей в его работе, обеспечения наблюдения за развитием ситуации и своевременного предупреждения о необходимости технического обслуживания и принятия решений, обеспечивающих надежную и безопасную эксплуатацию [1–5].

Такие системы являются одной из разновидностей экспертных систем, изучение которых проводится в рамках научного направления, которое называется инженерия знаний.

Цель исследований по экспертным системам состоит в разработке программ, которые при решении задач, трудных для эксперта-человека, получают результаты, не уступающие по качеству и эффективности решениям, получаемым экспертом [2]. Типичная экспертная система состоит из следующих основных компонентов [6–8]: рабочая память, база знаний, решатель, компонент приобретения знаний, объяснительный компонент, диалоговый компонент. Рабочая память предназначена для хранения исходных и промежуточных данных решаемой в текущий момент задачи. База знаний (БЗ) в ЭС предназначена для хранения долгосрочных данных, описывающих рассматриваемую область и правил, описывающих целесообразные преобразования данных этой области. Решатель, используя исходные данные из рабочей памяти и знания из БЗ, формирует такую последовательность правил, которые, будучи примененными к исходным данным, приводят к решению задачи.

Компонент приобретения знаний автоматизирует процесс наполнения ЭС знаниями, осуществляемый пользователем-экспертом. Объяснительный компонент объясняет, как система получила решение задачи (или почему она не получила решение) и какие знания она при этом использовала, что облегчает эксперту тестирование системы и повышает доверие пользователя к полученному результату. Диалоговый компонент ориентирован на организацию дружественного общения с пользователем, как в ходе решения задач, так и в процессе приобретения знаний и объяснения результатов работы.

Характерная особенность ЭС, отличающая их от традиционных систем обработки информации, заключается в оперировании знаниями. Формализм описания такого рода информации определяется как представление знаний. В системах с базами знаний, в том числе и экспертных системах, представление знаний является фундаментальным понятием. Решение о выборе способа представления знаний оказывает существенное влияние на любую их составную часть.

Существует несколько типов моделей представления знаний [1]:

- 1) декларативные (продукционные, редукционные, предикативные);
- 2) процедуральные (Planner, Conniver, ПРИЗ);
- 3) комбинированные (семантические сети, сети фреймов).

Декларативная модель состоит из двух частей: статических описательных структур знаний и механизма вывода, оперирующего этими структурами и практически независимого от их содержательного наполнения. В них не содержатся в явном виде описания выполняемых процедур. Эти модели представляют собой обычно множество утверждений. Предметная область представляется в виде синтаксического описания ее состояния. Вывод решений основывается в основном на процедурах поиска в пространстве состояний.

В процедуральном представлении знания содержатся в процедурах – небольших программах, которые определяют, как выполнять специфические действия (как поступать в специфических ситуациях). При этом можно не описывать все возможные состояния среды или

объекта для реализации вывода. Достаточно хранить некоторые начальные состояния и процедуры, генерирующие необходимые описания ситуаций и действий.

Наиболее распространенными моделями представления знаний в экспертных системах являются [9-12]:

1. Логическая модель представления знаний (предикативная).

Основная идея при построении логических моделей знаний заключается в следующем: вся информация, необходимая для решения прикладных задач, рассматривается как совокупность фактов и утверждений, которые представляются как формулы в некоторой логике [13]. Знания отображаются совокупностью таких формул, а получение новых знаний сводится к реализации процедур логического вывода. В основе логических моделей знаний лежит понятие формальной теории.

2. Продукционная модель.

Продукционная модель, или модель, основанная на правилах, позволяет представить знания в виде предложений типа «Если (условие), то (действие)». Под «условием» понимается некоторое предложение-образец, по которому осуществляется поиск в базе знаний, а под «действием» – действия, выполняемые при успешном исходе поиска. Чаще всего вывод на такой базе знаний бывает прямой (от данных к поиску цели) или обратный (от цели для ее подтверждения – к данным).

3. Фреймовая модель.

Фрейм – это абстрактный образ для представления некоего стереотипа восприятия. Во фреймовой системе единицей представления знания является объект, называемый фреймом. Он является формой представления некоторой ситуации, которую можно (или целесообразно) описывать некоторой совокупностью понятий и сущностей. Фрейм имеет определенную внутреннюю структуру, состоящую из множества элементов, называемых слотами, которым также присваиваются имена. Каждый слот, в свою очередь, представляется определенной структурой данных.

Совокупность фреймов, моделирующая какую-либо предметную область, представляет собой иерархическую структуру, в которую фреймы соединяются с помощью родовидовых связей. На верхнем уровне иерархии находится фрейм, содержащий наиболее общую информацию, истинную для всех остальных фреймов.

Фреймы обладают способностью наследовать значения характеристик своих родителей, находящихся на более высоком уровне иерархии. Значения характеристик фреймов могут передаваться по умолчанию фреймам, находящимся ниже их в иерархии, но если последние содержат собственные значения данных характеристик, то в качестве истинных применяются именно они.

4. Модель представления знаний в виде семантической сети.

Семантическая сеть представляет собой знания в виде графа, узлы которого соответствуют фактам или понятиям, а дуги – отношениям между понятиями. Как узлы, так и дуги обычно имеют метки. Проблема поиска решения в базе знаний типа семантической сети сводится к задаче поиска фрагмента сети, соответствующего поставленному вопросу [14].

Одним из примеров интеллектуальных систем поддержки принятия решения является «Комплекс программного обеспечения (КПО) автоматизированной системы анализа (АСА) при проведении испытаний», разработанный СКБ «Орион» [1, 2]. Это программный комплекс, обеспечивающий проведение автоматизированного анализа телеметрической информации при проведении испытаний путем реализации функций интеллектуальной обработки данных и знаний, получения и визуализации обобщенных оценок автоматизированного анализа состояния объектов испытаний [15].

Одной из составляющих КПО АСА является программа подготовки исходных данных и знаний о состоянии контролируемых объектов - автоматизированной подготовки исходных

данных (АПИД). Основной функцией программы АПИД является подготовка исходных данных для проведения автоматизированного анализа измерительной информации.

Создание и редактирование базы знаний в программе АПИД осуществляется с помощью редактора базы знаний. База знаний представляет собой набор данных внутри проекта для описания логики функционирования объекта анализа. На основе алгоритмов, описанных в базе знаний, происходит автоматическая генерация схем анализа. Создание и редактирование базы знаний выполняется в «Редакторе базы знаний об объектах анализа».

База знаний структурно подразделяется на следующие составляющие:

- классы, содержащие описание совокупности объектов, которые имеют общий набор свойств и обладают одинаковым поведением, с учётом принципа наследования, позволяющего создавать новые классы на основе существующих с возможностью добавления или переопределения данных и методов;

- непосредственно базы знаний, описывающие поведение объекта анализа на основе экземпляров объектов соответствующего класса.

Можно выделить следующие этапы разработки базы знаний:

1. Создание классов однотипных объектов, описывающих логику функционирования объекта анализа.

2. Создание базы знаний, детализация и создание иерархической структуры объекта анализа в соответствии с его структурой.

3. Создание экземпляров объектов определенного класса или объектов БЗ непосредственно описывающих логику анализа.

4. Задание исходных данных для объектов, привязка соответствующих атрибутов диагностики.

Основной целью магистерской диссертации является освоение и анализ моделей представления знаний и методов манипуляции знаниями в системе АПИД, а также разработка и внедрение новых.

На первом этапе необходимо ознакомиться с используемым в АПИД средством для разработки графического интерфейса Qt4. В целях обзора возможностей данной библиотеки был реализован перечень изменений в интерфейсе редактора базы знаний, таких как поиск с позиционированием по иерархической структуре базы знаний, а также по атрибутам для конкретного элемента базы знаний.

На втором этапе разработки в целях освоения программного проекта редактора базы знаний был реализован механизм отмены и возврата действий, который позволяет отменять и возвращать изменения, внесенные в структуру базы знаний, а также изменения, затрагивающие атрибуты элементов базы знаний. Для каждого действия отмены или возврата была предусмотрена система оповещения пользователя об изменении внесенных в структуру базы знаний или в атрибуты. Данная система оповещений была встроена в существующую систему текстового оповещения пользователя.

На следующих этапах работы будет производиться дальнейшее расширение возможностей существующей модели представления знаний.

В некоторых ситуациях свойства объекта-экземпляра могут представлять собой набор свойств, которыми обладают сразу несколько объектов. Для проектирования подобных объектов необходима реализация множественного наследования. Однако при множественном наследовании может возникнуть ситуация конфликта значений одноименных свойств объекта. В таком случае становится непонятно, какое значение свойства должно быть у объекта-наследника. Для того чтобы избежать подобных ситуаций, можно использовать интерфейсы. Интерфейс – это класс, который не содержит в себе никаких конкретных значений свойств, а только задает структуру объекта. При множественном наследовании от двух и более интерфейсов вопрос конфликта значений свойств будет снят, и в объекте-потомке множество свойств будет являться объединением всех свойств объектов-предков.

Система интерфейсов и множественного наследования внесет дополнительную гибкость для разработки базы знаний, а также в отдельных случаях позволит задавать определенную архитектурную жесткость. Например, наследование от некоторого интерфейса подразумевает необходимость назначения свойствам порожденного объекта конкретных значений в соответствии с архитектурой интерфейса.

Подводя итог, на первоначальном этапе работы была изучена теоретическая основа построения экспертных систем, роль базы знаний в экспертных системах, произведен обзор основных методов представления знаний и их сравнительная характеристика. Произведено изучение программного проекта для автоматизированной подготовки исходных данных в составе комплекса программного обеспечения автоматизированной системы анализа при проведении испытаний. Также были получены практические навыки использования библиотеки Qt и реализован перечень функций, расширяющих возможности редактора базы знаний, которые делают процесс проектирования базы знаний более технологичным.

### Библиографический список

1. *Охтилев М.Ю., Б.В.Соколов, Р.М.Юсупов* Интеллектуальные технологии мониторинга и управления структурной динамикой сложных технических объектов. М.: Наука, 2006. 411с.
2. *Ахметов Р.Н., Васильев И.Е., Капитонов В.А., Охтилев М.Ю., Соколов Б.В.*, Концепция создания и применения перспективной АСУ подготовки и пуска ракеты космического назначения «Союз-2»: новые подходы к интеграции, интеллектуализации, управлению, // *Авиакосмическое приборостроение*. 2015. № 4.С. 3.
3. *Майданович О.В., Каргин В.А., Мышко В.В., Охтилев М.Ю., Соколов Б.В.* Теория и практика построения автоматизированных систем мониторинга технического состояния космических средств : монография,; под ред. О.В.Майдановича. СПб.: ВКА имени А.Ф.Можайского, 2011. 219 с.
4. *Майданович О.В., Охтилев М.Ю., Соколов Б.В., Юсупов Р.М.* Комплексная автоматизация мониторинга состояния космических средств на основе интеллектуальных информационных технологий // Приложение к журналу «Информационные технологии». 2011. № 10. С. 2.
5. Интеллектуальные системы мониторинга и контроля состояния технически сложных объектов [Электронный ресурс] // URL: <http://www.nicetu.ru/reshenija-i-produkty/intellektualnye-sistemy-monitoringa-i-kontrolja-sostojaniya-tehnicheskikh-slozhnykh-obektov/> (дата обращения: 10.03.2016).
6. *Коробова И.Л.* Структура и состав экспертной системы // *Методы представления знаний*: Тамбов: ТГТУ, 2003. С. 5–6.
7. *Хабаров С.П.* Введение в экспертные системы [Электронный ресурс] // URL: [http://www.habarov.spb.ru/new\\_es/exp\\_sys/es01/es1.htm](http://www.habarov.spb.ru/new_es/exp_sys/es01/es1.htm) (дата обращения: 10.03.2016).
8. Экспертные системы [Электронный ресурс] // URL: [http://www.mari-el.ru/mmlab/home/AI/7\\_8/index.html](http://www.mari-el.ru/mmlab/home/AI/7_8/index.html) (дата обращения: 10.03.2016).
9. Представление знаний в интеллектуальных системах [Электронный ресурс] // URL: <http://www.mari-el.ru/mmlab/home/AI/4/index.html> (дата обращения: 10.03.2016).
10. *Спицын В.Г., Цой Ю.Р.* Модели представления знаний // *Представление знаний в информационных системах*: Томск: Издательство ТПУ, 2006. С. 29–45.
11. Модели и методы представления и организации знаний [Электронный ресурс] // URL: [http://it-claim.ru/Library/Books/Semantics\\_IT/gl1\\_1/glava1\\_1.htm](http://it-claim.ru/Library/Books/Semantics_IT/gl1_1/glava1_1.htm) (дата обращения: 10.03.2016).
12. Представление знаний в интеллектуальных системах [Электронный ресурс] // URL: <http://nrsu.bstu.ru/chap13.html> (дата обращения: 10.03.2016).
13. Логическая модель знаний [Электронный ресурс] // URL: <http://www.aiportal.ru/articles/knowledge-models/logical-model.html> (дата обращения: 20.03.2016).
14. Сетевая модель представления знаний [Электронный ресурс] // URL: <http://itteach.ru/predstavlenie-znaniy/setevaya-model-predstavleniya-znaniy> (дата обращения: 20.03.2016).
15. Комплекс программного обеспечения автоматизированной системы анализа при проведении испытаний. Руководство оператора, 2013. 149 с.

УДК 629.735.33

**В. А. Шведчиков**

студент кафедры компьютерных технологий и программной инженерии

**М. В. Фаттахова**

доц., канд. Физ.-мат. наук – научный руководитель

## **МАРШРУТИЗАЦИЯ ТРАНСПОРТНОГО ПОТОКА С ОГРАНИЧЕНИЕМ ПО ГРУЗОПОДЪЕМНОСТИ**

Успех любого предприятия, занятого транспортировкой грузов, всегда связан с эффективным принятием решений в области транспортной логистики. По разным оценкам от 30 до 50 процентов всех затрат на логистику связано с транспортными издержками. Формирование рациональных маршрутов помогает добиться сокращения не только общего расстояния и, как следствие, пробега транспортных средств, но и временных, и материальных затрат. Кроме того, оптимизация процесса доставки ведет к снижению товарно-производственных запасов на складах, что приводит к оптимизации всей совокупности логистических мероприятий.

Предметом исследования проведенной научной работы стала наиболее важная задача в данной области – формирование маршрутов доставки груза клиентам заданным количеством транспортных средств с обязательным возвращением в исходный пункт после окончания поездки.

Математическая формулировка данной задачи известна как задача маршрутизации транспорта (ЗМТ) или Vehicle Routing Problem (VRP). Цель решения ЗМТ состоит в минимизации общей длины маршрутов транспортных средств, обслуживающих клиентов сети. Данный класс задач является обобщением задачи коммивояжера на случай построения нескольких замкнутых маршрутов, имеющих начало и конец в одной общей вершине.

Задачи маршрутизации транспорта представляют собой довольно крупный класс задач, включающий в себя ряд разновидностей – подклассов, каждый из которых предназначен для более полного представления деталей реальной действительности. Переход от ЗМТ общего вида к конкретному подклассу осуществляется с помощью включения дополнительных ограничений в условия задачи.

В рамках настоящей работы исследуется задача маршрутизации транспорта с ограничением, очень часто имеющим место в реальной ситуации, а именно с ограничением на количество транспортных средств, находящихся в распоряжении компании, с учетом их грузоподъемности.

В связи с тем, что задачи маршрутизации являются NP-трудными, применение точных методов решения является затруднительным: из-за чрезмерно быстрого роста времени вычислений их невозможно применять для задач с более чем 25 вершинами. Реальные же транспортные сети могут включать в себя гораздо большее количество узлов, поэтому наибольшее внимание уделяется метаэвристическим методам. В настоящее время ученые и специалисты всего мира занимаются усовершенствованием имеющихся и поиском новых методов, которые позволят получить еще большую производительность алгоритмов и будут в состоянии находить новые лучшие решения для задачи маршрутизации транспортных средств.

Для решения поставленной задачи, был выбран алгоритм муравьиной колонии.

Идея метода заключается в моделировании поведения колонии муравьев в процессе поиска пути к источнику пищи. Несмотря на то, что отдельный муравей (агент системы) достаточно примитивен и в одиночку не способен к принятию оптимальных решений, поведение всей колонии в своей совокупности оказывается разумным. Интеллектуальность такой системы обоснована низкоуровневым взаимодействием между агентами, которое происходит через химическое вещество – феромон, откладываемое муравьями на своем пути. Выбирая

направление движения, каждый агент системы основывается на опыте предыдущих поколений (количестве феромона, отложенного колонией ранее). Таким образом, чем чаще муравьи предпочитают данный маршрут, тем больше феромона будет на нем отложено, и, следовательно, вероятность того, что следующий муравей будет двигаться в этом направлении, увеличивается. Однако такая положительная связь все муравьи будут двигаться только по одному пути. Поэтому в качестве отрицательной обратной связи вводится процесс испарения феромона, который гарантирует нам исследование новых маршрутов.

Опишем поведение муравьев и их свойства при выборе пути (рис. 1, 2):

1) муравьи имеют собственную «память», которая представляется в виде списка вершин, в которых данный муравей уже побывал;

2) муравьи обладают «зрением». за зрение муравья принимается величина обратно пропорциональная длине ребра:  $n_{ij} = 1/D_{ij}$ ;

3) муравей способен улавливать след феромона, который будет определять желание муравья пройти по данному ребру. уровень феромона в момент времени  $t$  на ребре  $D_{ij}$  будет соответствовать  $\tau_{ij}(t)$ .

4) вероятность перехода муравья из вершины  $i$  в вершину  $j$  определяется следующим соотношением:

$$P_{ij,k}(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \frac{1}{[D_{ij}(t)]^\beta}}{\sum_{l \in J_{i,k}} [\tau_{il}(t)]^\alpha [n_{il}(t)]^\beta}, j \in J_{i,k}; \\ P_{ij,k}(t) = 0, j \notin J_{i,k}, \end{cases} \quad (1)$$

где:  $\alpha$  – эмпирический коэффициент «жадности» алгоритма,  $\beta$  – эмпирический коэффициент «стадности» алгоритма,  $J_{i,k}$  – список вершин, которые ещё не были посещены данным муравьем;

5) на своем пути каждый муравей откладывает феромон, количество которого определяется по следующей формуле

$$\Delta\tau_{ij,k}(t) = \begin{cases} \frac{Q}{L_k(t)}, (i,j) \in T_k(t); \\ 0, (i,j) \notin T_k(t), \end{cases} \quad (2)$$

где  $Q$  – параметр, имеющий значение порядка длины оптимального пути;  $L_k(t)$  – длина маршрута  $T_k(t)$ .

6) испарение феромона определяется следующим выражением:

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij,k}(t), \quad (3)$$

где  $m$  – количество муравьев,  $\rho$  – коэффициент испарения ( $0 \leq \rho \leq 1$ ),

7) максимальный вес груза, который способен перенести каждый муравей на одном пути. не превосходит значение  $W$ .

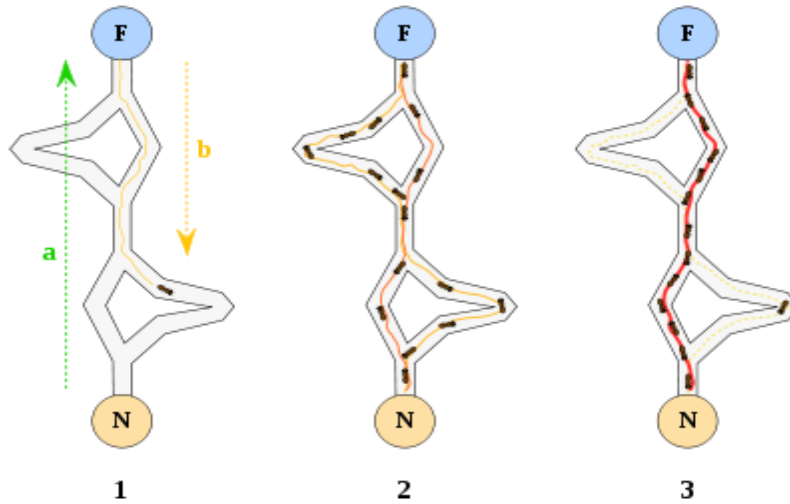


Рис. 1. Поведение муравьёв



Рис. 2. Алгоритм муравьиной колонии

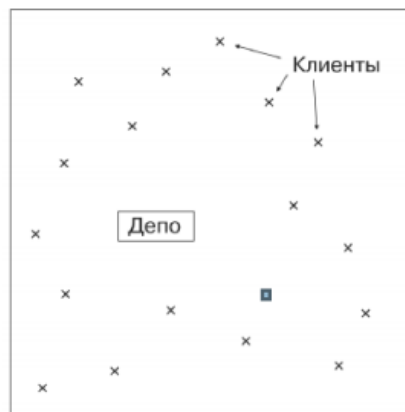


Рис. 3. Описание муравьиного алгоритма для задачи CVRP



Входными данными для данной задачи является множество вершин  $V = \{v_0, v_1, \dots, v_n\}$ , в которых заданы координаты и вес груза, который необходимо доставить в каждую вершину.

Для начала необходимо рассчитать матрицу расстояний  $D$  между всеми вершинами, после чего задать начальное количество феромона на всех дугах и определить коэффициенты  $\alpha$ ,  $\beta$ ,  $Q$ ,  $\rho$ . Выбор коэффициентов определяет эффективность работы алгоритма.

Работа алгоритма начинается с того, что каждый муравей начинает передвигаться из вершины в вершину. Для выбора следующей вершины оценивается вероятности перехода в каждую из ещё не посещенных муравьем вершин. Значения вероятностей зависят от величины, обратно пропорциональной длине дуги, соединяющей текущую и выбранную вершины, а также от количества феромона на данной дуге. Расчет вероятностей производится по формуле (1).

Следующим шагом генерируется случайное число. В зависимости от того, в какой вероятностный диапазон оно попало, выбирается следующая вершина.

Если грузоподъемность муравья  $W$  достигла максимально возможной грузоподъемности  $W_{\max}$ , муравей вынужден вернуться в стартовую вершину (депо), а затем снова продолжить обход оставшихся вершин. Этот процесс завершается, как только список  $J_{i,k}$  не посещенных муравьем вершин становится пуст. Когда будут пройдены все вершины, произойдет обновление количества феромона на пройденном маршруте согласно формуле (2) и испарение феромона на всех вершинах согласно формуле (3).

Длина всех путей  $L$ , пройденных  $k$ -ым муравьем, сравнивается с лучшей длиной маршрутов на текущий момент  $L^*$ . Если полученное значение меньше текущего лучшего, то происходит обновление информации о лучшем маршруте.

Работа алгоритма заканчивается в том случае, когда все муравьи пройдут по всем вершинам. Решение задачи будет множество маршрутов  $T^*$  и их общая длина  $L^*$ .

Для проверки работоспособности алгоритма была написана программа, находящая для множества вершин с заданным весом заказа, разбиение на  $m$  маршрутов, количество и суммарная длина которых минимальна.

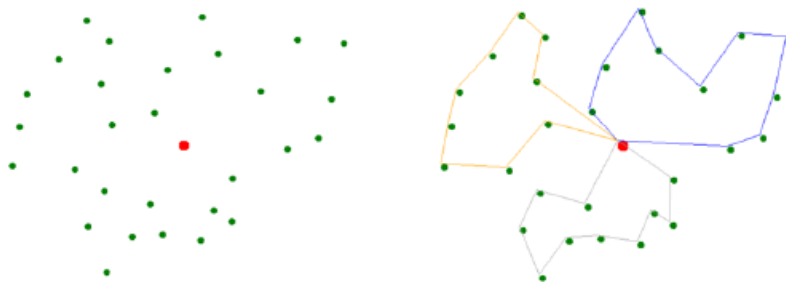


Рис. 4. Результат работы алгоритма в графическом виде

**Библиографический список:**

1. Кажаров А. А., Курейчик В. М. Муравьиные алгоритмы для решения транспортных задач. // Известия Российской академии наук. Теория и системы управления. 2010. № 1. С. 32–45.
2. Варламов О.О. Эволюционные базы данных и знаний для адаптивного синтеза интеллектуальных систем. Миварное информационное пространство. М: Радио и связь, 2002. 286 с.
3. Свами М., Тхуласираман К. Графы, сети и алгоритмы. М.: Мир, 1984. 454 с.

УДК 004.51

**И.В. Шульга**

студент кафедры компьютерных технологий и программной инженерии

**М.Д. Поляк**

ст. преподаватель – научный руководитель.

## **ЗАОЧНОЕ ГОЛОСОВАНИЕ СОБСТВЕННИКОВ ПОМЕЩЕНИЙ В МНОГOKВАРТИРНОМ ДОМЕ В ФОРМЕ ЭЛЕКТРОННОГО ГОЛОСОВАНИЯ**

Предложен способ проведения электронного голосования собственников помещений в многоквартирном доме с помощью Android – приложения.

В настоящее время всё больше в нашу жизнь входят электронно-цифровые устройства, которые призваны облегчать нам повседневные задачи. Они получили широкое применение и в управлении спутниками, и в управлении обычным телевизором. Тем самым с развитием технологий всё больше задач берут на себя технические устройства. В эпоху новых технологий, можно за пару секунду получить необходимые данные и сформировать документ, который может быть распечатан ещё через пару секунд. Поэтому во многих сферах всё чаще стараются внедрять электронно-цифровые технологии для автоматизации и облегчения управления потоками данных.

Основой управления многоквартирными домами являются собрания собственников, именно на этих собраниях путем голосования и принимаются все решения по управлению многоквартирным домом. Собрания собственников бывают двух видов [1]:

- Очное собрание собственников.
- Заочное собрание собственников.

Эти же собрания являются и слабым звеном, препятствующим эффективному и оперативному принятию решений, ведь для их проведения необходимо собрать большинство жильцов в одно время в одном месте, что крайне затруднительно в связи с невозможностью выбрать время, которое было бы одинаково удобно для всех. Ещё одну проблему составляет подсчёт результатов голосования. Для принятия решения по одному вопросу и в небольшом доме подойдёт простой подсчёт поднятых рук. Однако в повестку дня может быть включено и множество вопросов. В таком случае можно использовать бюллетени для голосования, которые необходимо раздать, заполнить, а затем обработать для подсчёта голосов. Для больших домов это будет создавать огромные трудности, как по времени, так и по ресурсам, и при этом повышается вероятность ошибок, связанных с человеческим фактором.

Для решения части этих задач может помочь Android приложение. Мобильные телефоны есть практически у каждого человека и самой распространённой операционной системой для них является система Android. Используя телефоны для проведения голосования, мы позволим всем собственникам участвовать в голосовании в удобное для них время. Также мы облегчим задачу подсчёта результатов голосования, ведь телефон будет отправлять на сервер результаты голосования конкретного человека уже в электронном виде, останется только записать их в базу данных и по окончании голосования выдать результаты, подсчитанные ЭВМ. Таким образом достигается колоссальный выигрыш и во времени, и в ресурсах, как для проведения голосований, так и для подсчёта результатов.

Операционная система Android была выбрана не случайно. Согласно исследованиям Gartner Analysts за четвёртый квартал 2014 года доля рынка устройств на базе этой системы составляла 76%, а за аналогичный период 2015 года уже 80,7%. Это говорит о широкой распространённости операционной системы Android и с каждым годом количество пользователей этой системы только растёт [2].

Приложение будет разрабатываться на языке программирования Java, который является основным для ОС Android. Сервер для данного приложения будет реализован на языке

программирования Python с использованием фреймворка Django, что позволит создать легко расширяемую структуру для дальнейшего совершенствования системы. Так же сервер будет соответствовать принципам REST для унификации способа общения между клиентом и сервером. Клиент–серверное взаимодействие будет осуществляться с помощью протокола HTTP и текстового формата обмена данными JSON.

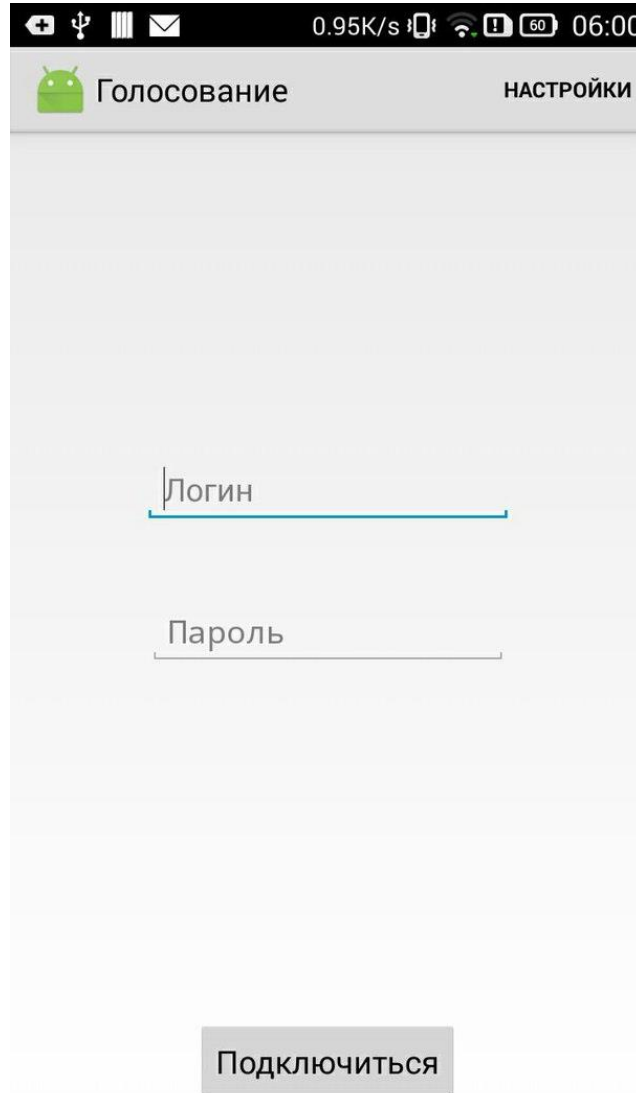


Рис. 1. Экран авторизации.

Для использования приложения пользователю понадобится смартфон, работающий на операционной системе Android с версией не ниже 4.0. Для установки приложения на смартфон необходимо загрузить на смартфон установочный файл приложения. После запуска приложения пользователю необходимо пройти авторизацию (рисунок 1) для получения доступа к списку активных собраний, в которых он может принять участие и проголосовать по поставленным вопросам (рис. 2).

В результате проведенной работы удалось реализовать Android приложение, которое в удобной форме обеспечивает участие пользователей в голосованиях для принятия решений по вопросам, вынесенным на собрание собственников. Пользователь сможет участвовать в голосовании в удобное для него время. Так же удалось реализовать сервер, отвечающий требованиям REST. Он позволяет без изменения кода создавать приложения и для других платформ, таких как iOS, Windows Phone и др. Разработанная система является достаточно

гибкой и предоставляет широкие возможности для дальнейшего усовершенствования, доработки и развития проекта.

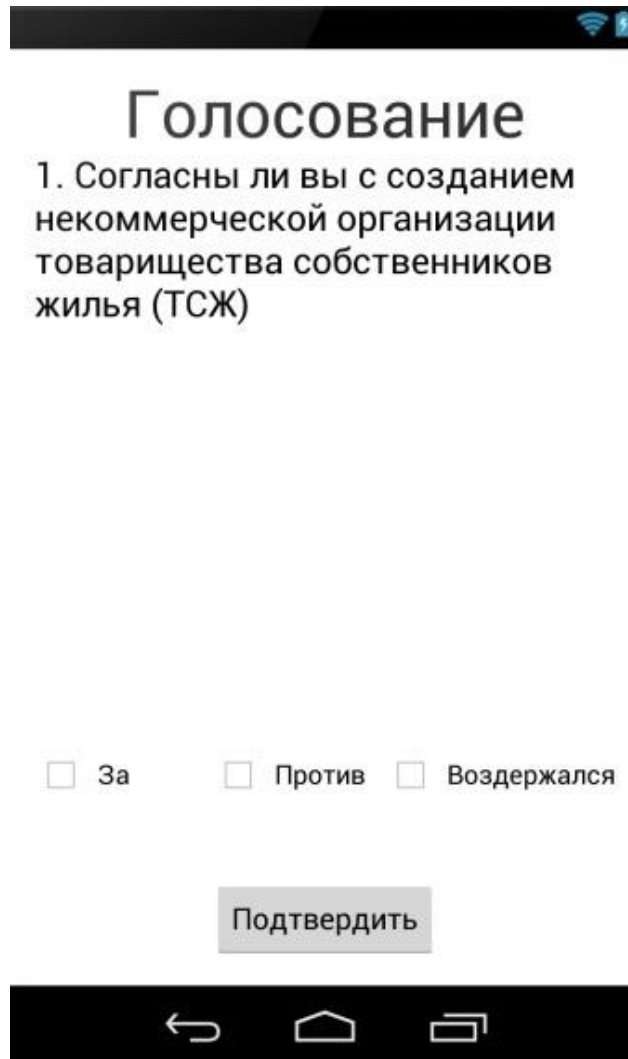


Рис. 2 Интерфейс экрана голосования на Android устройстве.

**Библиографический список:**

1. Жилищный кодекс Российской Федерации: федеральный закон РФ от 29.12.2004 г. № 188 – ФЗ.
2. Gartner Analysts <http://www.gartner.com/newsroom/id/3215217> (20.04.16)

# Информационные системы и защита информации

УДК 004.7

**Е. Н. Воропаева**

студент кафедры инфокоммуникационных систем

**А. М. Тюрликов**

зав. кафедры инфокоммуникационных систем – научный руководитель

## ИСПОЛЬЗОВАНИЕ СИСТЕМЫ МОДЕЛИРОВАНИЯ ДЛЯ ПРОВЕРКИ ЗНАНИЙ

Контроль знаний учащихся является неотъемлемой частью процесса обучения. От его правильной организации зависят эффективность управления учебно-воспитательным процессом и качество обучения учащихся. Проблема контроля учебной деятельности не нова, а педагогический опыт, накопленный в этой области, богат и разнообразен, однако наиболее распространенным контролем знаний является традиционная система тестов со скрытыми ответами.

В качестве примера существующих систем контроля знаний рассмотрим приведенные ниже системы.

1. Moodle (Modular Object-Oriented Dynamic Learning Environment – модульная объектно-ориентированная динамическая среда обучения)[1] – это свободная система с открытым исходным кодом.

В данной системе преподаватель самостоятельно формулирует задания теста и добавляет их в систему с помощью «перетаскивания» документов в окно системы. Система перетасовывает задания и выдает тесты студентам.

2. Mirapolis Testing [2]. Систему проверки знаний Mirapolis Testing разработал учебный центр «СТЕК». В данной системе хранятся списки вопросов и ответов к ним. Для тестирования студентов из общего списка выбирается нужное количество вопросов и соответствующие ответы на них.

В подобных традиционных системах заранее сформировано множество вопросов и множество скрытых ответов на них. Это наиболее простой и распространенный подход, однако, он имеет ряд недостатков:

1) вопросы составляет преподаватель и, как следствие, их будет небольшое количество, поэтому велика вероятность того, что многие студенты получат один и тот же перечень вопросов;

2) поскольку в системе хранятся и задания, и ответы, такая система занимает большой объем памяти;

3) при взломе такой системы к злоумышленнику попадает вся информация, в том числе все ответы на вопросы.

Поэтому актуальной задачей является разработка программного комплекса, автоматически формирующего задания для студентов. Это позволит увеличить количество вариантов и преодолеть второй и третий недостатки традиционной системы. Для решения этой задачи необходимо разработать алгоритм формирования исходных данных, который должен обладать следующими свойствами:

1. база данных не должна хранить задания и ответы на них;

2. вероятность совпадения заданий у двух студентов одной группы не должна превосходить заданный порог;

3. все исходные данные должны обладать приблизительно одинаковой сложностью.

Однако, минусом такой системы, по сравнению с традиционной, где для формирования заданий нужен только преподаватель, является тот факт, что, если преподаватель не может сам запрограммировать алгоритм, моделирующий исходные данные, то придется привлекать программиста.

Покажем, как можно достичь выполнения первого требования.

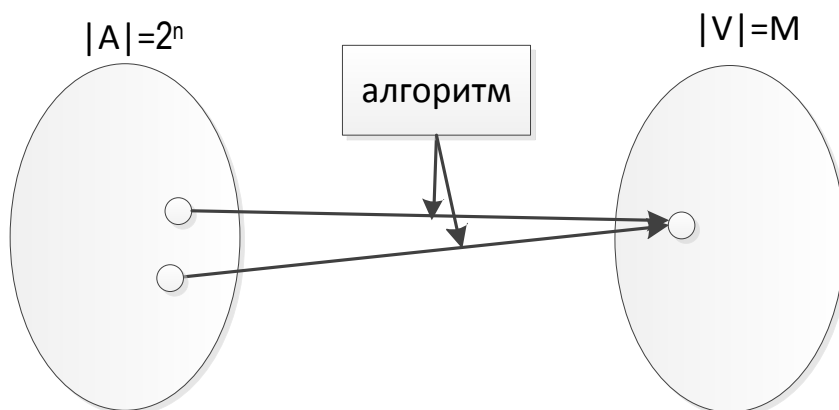
Исходные данные для каждого студента генерируются программой, при этом начальным параметром, определяющим генерацию исходных данных, является случайное значение, сгенерированное датчиком псевдослучайных чисел. При таком подходе информация о начальных данных и правильном ответе хранится в базе данных в виде одного числа, задающего параметр для запуска моделирующей программы. Преподаватель всегда сможет получить исходные и выходные данные программы, введя вначале это число, но при этом, в случае утечки информации из базы данных или ее взломе, злоумышленник сможет извлечь лишь одно число, которое не несет в себе никакой информации о самом задании или ответе на него.

При этом дополнительным преимуществом является существенное уменьшение информации, хранимой в базе данных, и, как следствие, она будет занимать значительно меньший объем памяти.

Рассмотрим более подробно то, как на основе начального числа могут быть сформированы индивидуальные задания.

Имеется множество вариантов начального значения датчика псевдослучайных чисел, составляющих множество  $A$ .  $|A| = 2^n$ ,  $n$  – машинная разрядность (в современных языках программирования используется разрядность не менее 32 бит). Случайным образом выбирается элемент этого множества, и далее с помощью алгоритма, разработанного для каждого вида лабораторных работ, путем последовательных обращений к датчику псевдослучайных чисел генерируется задание. Таким образом начальное значение однозначным образом определяет вариант задания.

Мы считаем, что если начальное число выбирается равновероятно и что мы используем обращение к датчику, который также равновероятно генерирует числа, то все сгенерированные варианты заданий также равновероятны. Это позволяет рассчитать вероятность совпадения заданий у двух студентов одной группы способом, описанным ниже.



Отображение множества  $A$  на множество  $V$

Как мы можем видеть из рисунка, отображение множества значений датчика на множество вариантов – не взаимно однозначное. Это объясняется тем, что мощность множества  $A$  фиксирована, и отображение определяется мощностью множества  $V$ , равного количеству заданий, которые могут быть сгенерированы при выбранных условиях алгоритма. Именно поэтому важную роль играет количество заданий, которые можно составить для каждого типа лабораторных работ: их количество должно быть большим, чтобы обеспечить множество  $V$  достаточным количеством элементов.

Учитывая то, что задания у разных студентов одной группы формируются независимо и равновероятно, вероятность того, что в группе из  $n$  человек и общем числе заданий  $M$  задания у всех студентов будут различные, можно рассчитать по формуле:

$$p(n) = \frac{M!}{M^n (M-n)!} \quad (1)$$

Отсюда следует, что вероятность совпадения заданий  $Pr$  рассчитывается с помощью формулы (2):

$$Pr(n) = 1 - p(n). \quad (2)$$

Обеспечить третье требование можно только применительно к конкретной области. Проиллюстрируем его на примере курса систем и сетей передачи данных на основе алгоритма «с возвратом назад» для передачи по каналу с обратной связью [3].

В англоязычной литературе алгоритм носит название GoBackN [4] (в случае ошибки повтор последних  $N$  пакетов), сокращенная его форма GBN. В GBN считается, что на приёмной стороне нет буфера для полученных сообщений, т. е. если сообщение успешно принято, то оно сразу передаётся получателю, в противном случае оно уничтожается из-за того, что в нём есть ошибки. Даже если сообщение передано верно, но предыдущее сообщение по какой-либо причине уничтожено (ошибка произошла в нем или в квитанции), полученное сообщение также уничтожается.

Для проверки знаний работы этого алгоритма студенту задается число сообщений в системе  $m'$ , количество ошибок  $er$ , ограничение на длину пакета ошибок  $pack$ . Студент входит в систему, которая запускает программу, моделирующую работу GBN. В этой программе случайным образом задается местоположение ошибок, после чего студенту выводится информация о количестве сообщений в системе и моментах времени, в которые появились ошибки. Задачей студента является составление временной диаграммы и получение моментов времени, в которое каждое сообщение покинет систему. После того, как студент получил задание, моделирующая программа так же подсчитывает моменты выхода сообщений, после того, как студент выполнил задание, он вводит полученную последовательность в программу, в которой сверяются данные, введенные студентом, с данными, которые были посчитаны в программе. Если результаты сошлись, студенту засчитывается тест.

Приблизительно одинаковая сложность заданий достигается за счет того, что всем студентам задается приблизительно одинаковое количество сообщений в системе  $m'$ , допускается примерно одинаковое количество возникающих ошибок  $er$ , а также вводится дополнительный параметр, накладывающий ограничение на длину пакета ошибок  $pack$  ( $pack \leq 2$ ).

С учетом этого, количество возможных заданий рассчитывается по следующим формулам:

$$M = C_{m'}^{er} - n \cdot pack(m', er); \quad (3)$$

$$C_{m'}^{er} = \frac{m'!}{(m' - er)! er!}, \quad (4)$$

где  $n \cdot pack(m', er)$  – количество заданий, не удовлетворяющих требованиям (те варианты, в которых длина пакета ошибок более двух).

Также посчитаем вероятности по формулам (1) и (2). Полученные данные внесем в таблицу, в которой первые 3 столбца обеспечивают выполнение третьего требования, а последние 3 – второго.

Рассчитанные параметры для различных вариантов начальных данных при  $n=20$

$m'$	$er$	$pack$	$C$	$M$	$Pr(n)$
10	4	15	210	195	64%
12	3	10	220	210	60%

m'	er	pack	C	M	Pr(n)
12	4	19	495	476	33%
12	5	27	792	765	21%
15	3	13	455	442	17%
15	4	25	1365	1340	11%
15	5	36	3003	2967	5%

Из таблицы следует, что если требуется обеспечить вероятность события, то в группе из двадцати студентов вероятность совпадения заданий хотя бы у двух человек не должен превышать 5%, достаточно сформировать задания, в которых будут 15 сообщений и 5 ошибок. При этом, из заданий надо будет исключить 36 вариантов расположения пакетов ошибок (пакеты ошибок длины 3,4,5).

Описанный выше подход также можно использовать для проверки знаний по другим алгоритмам передачи по каналам с обратной связью. При этом потребуется поменять алгоритм формирования исходных данных и алгоритм проверки результатов. Подход может быть применен и к другим областям знаний, в которых может быть формализован алгоритм формирования заданий, отвечающий выше сформулированным требованиям, и алгоритм проверки правильности ответов.

### Библиографический список

1. Moodle (Modular Object-Oriented Dynamic Learning Environment), <https://moodle.com>
2. Система онлайн тестирования «СТЕК», <http://ra-kurs.spb.ru/0/0/3/5/?id=34>
3. Бертсекас Д., Галлагер Р. Сети передачи данных/ пер. с англ./под ред. Б. С. Цыбакова М.: Мир, 1989 г., 544 с.
4. Krouk E.A. Modulation and coding techniques in wireless communication/ E.AKrouk, Semenov S. 2011, p 263-275.

УДК 004

**Н. А. Голос**

студент кафедры технологий защиты информации

**С. В. Беззатеев**

д-р техн. наук, доцент – научный руководитель

### АНАЛИЗ БЕЗОПАСНОСТИ ПРОТОКОЛА HTTPS

В данной статье рассмотрен прокол https, а так же TLS/SSL-уровень, отвечающий за безопасность в протоколе, были выявлены его уязвимости, а так же методы борьбы с ними. В заключение приведены примеры настроек, которые на данный момент обеспечивают максимальную безопасность при взаимодействии в Интернете.

Https – это защищенная версия http протокола. Он состоит из следующих уровней:

- HTTP
- SSL/TLS
- TCP
- IP

IP-протокол (Internet Protocol) отвечает за адресацию узлов в Интернете, а также обеспечивает доставку IP пакетов на нужные адреса. При этом он не гарантирует надежной доставки и того, что пакеты придут в надлежащем порядке. Так же пакеты в процессе передачи по протоколу могут продублироваться или быть повреждены и не дойти вовсе.

За гарантированную доставку отвечает протокол TCP (transmission control protocol). Он работает следующим образом. Сначала клиент с сервером устанавливает соединение, обмени-



ваясь специальными пакетами, далее по данному соединению в обе стороны посылаются пакеты, при этом механизм TCP в случае потери данных отправляет пакет повторно, а в случае их дублирования удаляет лишнее, гарантируя целостность передаваемых данных.

SSL/TLS-уровень отвечает за защищенность передаваемых данных в Интернете путем шифрования данных и аутентификации сервера и клиента, а так же за то, чтобы данные не были подменены в процессе передачи.

HTTP – протокол передачи гипертекста, разработанный для обмена «запросами-ответами» между сервером и клиентом.

Далее будем рассматривать именно SSL/TLS-уровень, так как именно он отвечает за безопасность https протокола. А точнее Протокол TLS отвечает за аутентификацию, шифрацию и целостность в https. Аутентификация нужна для проверки авторства передаваемой информации, шифрование для сокрытия информации при передаче и целостность для обнаружения подмены информации и нахождения ошибок при передаче.

Перед тем как начать обмениваться данными, сервер и клиент должны согласовать ряд параметров соединения таких как: сертификаты, способ шифрования и версии протоколов. Начало соединения называется TLS Handshake (рукопожатие). В конце рукопожатия клиент и сервер согласуют все параметры соединения и начнут шифровать сообщения выработанным сеансовым ключом с помощью синхронного шифрования.

Итак, чтобы злоумышленнику получить конфиденциальные данные из общения клиента и сервера ему нужно перехватить трафик и каким-то образом его расшифровать. Рассмотрим, как это можно сделать.

Для шифрования используются блочные и потоковые шифры.

Блочные шифры работают с блоками фиксированной длины и требуют разбиение потока байтов на данные блоки. При использовании данного метода шифрования каждый байт шифруется в составе блока байтов. Если исходный текст не кратен размеру блока, используется padding (дополнение) последнего блока недостающими байтами, что может повлечь за собой определенные уязвимости. На данный момент рекомендуется использовать GCM-режим шифрования, который так же поддерживает аутентификацию, однако, в интернете распространен CBC режим.

Уязвимость данного шифра заключается именно в способе дополнения блока. Основным способом дополнения блоков является PKCS7, в соответствии с которым значение каждого дополняемого байта устанавливается равным количеству дополняемых байт. Например, если размер блока 16 байт, а входной блок 13 байт, то к нему добавится [03, 03, 03]. Это значит, что в конце блока не может быть добавлено, например, следующее [05,09,02,13], так как не существует текстов, которые могли быть преобразованы в подобный шифротекст.

Суть атаки заключается в том, что, если злоумышленник перехватил зашифрованные данные, затем начал отправлять их на сервер, который возвращает информацию корректно ли дополнение или нет, то он сможет расшифровать любой зашифрованный текст.

Первая подобная атака была реализована профессором Швейцарского федерального технологического института в Лозанне Сержем Воденэ в 2003 году [2]. Данной атаке была подвергнута реализация SSL между почтовым сервером и клиентом.

Допустим, у нас есть запрос – <https://www.example.com/home.asp?UID=8A219A434525535FF324D4G56FC95348>, где содержимое UID, текст, который нужно расшифровать. UID состоит из четырех блоков 8A219A43| 4525535F| F324D4G5| 6FC95348.

Теперь злоумышленник может взять первый инициализирующий блок и начать изменять его значения, начиная с 00000000 и заканчивая 000000FF пока сервер в ответ на запрос выдаст не ошибку, а успех. Это произойдет в том случае, когда сервер расшифрует последний байт инициализирующего блока и получит 0x01.

В случае успеха можно будет вывести значение промежуточного байта, оно будет равно  $0x3C \oplus 0x01 = 0x3D$ . С помощью этого промежуточного байта теперь можно получить расшифро-

ванное значение данного «проксорив» его с исходным шифротекстом. Далее, выполняя данный процесс для каждого байта, можно получить весь инициализирующий вектор, а затем расшифровать и все значение целиком.

Так как CBC-режим блочного шифрования был скомпрометирован, то популярность стал набирать RC4 режим потового шифрования. По оценке экспертов, он используется в 50% всех TLS-соединений, текущая же оценка составляет 30% (на 2013г) [3].

Как и все потоковые шифры RC4 берет 128 или 256-битный ключ и растягивает в длинную строку, состоящую из псевдо случайных байтов. Эти байты XOR'ят с сообщением, и в результате получается достаточно случайно выглядящий шифротекст.

Проблема же заключается в том, что байты гаммы недостаточно случайны, они имеют небольшие уклоны. Недавно были открыты дополнительные существенные уклоны в первых 256 байтах RC4 вывода. В теории, если шифровать одно и тоже сообщение множеством различных RC4-ключей, должны получиться наборы совершенно случайных шифротекстов, однако, из-за данных уклонов, статистический анализ различных позиций шифротекста покажет, что некоторые значения появляются чаще остальных. Если же получить множество различных шифротекстов одного сообщения, можно просчитать отклонения и расшифровать текст.

С помощью данной уязвимости можно, например, расшифровать cookie, которые использует сайт для вашей аутентификации. Если соединение клиента с сайтом зашифровано с использованием RC4, то каждый раз, когда вы соединяетесь с сервером, вы отправляете новую копию ваших зашифрованных cookies. Если сессия между клиентом и браузером перезапускается, то соответственно генерируются новые ключи между данными соединениями, из-за этого злоумышленник может построить список шифротекстов, которые ему нужны для взлома.

Чтобы ускорить данный процесс, можно внедрить специальный код на стороне клиента, чтобы инициировать множество https-запросов, следовательно увеличить скорость взлома.

Исследования сотрудников из бельгийского университета Левена показали, что зашифрованный cookie можно расшифровать с вероятностью в 94% за 75 ч. При этом использовались  $9 \cdot 2^{27}$  шифротекстов. На 2016г. данный вид шифрования использовать крайне нежелательно и многие крупные компании (Mozilla, Google, Microsoft) собираются отказаться от поддержки RC4 из-за найденных в нем уязвимостей [4].

Помимо взлома шифров есть другой способ получения конфиденциальной информации – это Man in the middle attack. Атака Man in the middle (человек посередине) представляет собой случай, когда злоумышленник способен просматривать или изменять сообщения, которыми обмениваются абоненты, при этом ни один из них не догадывается о его присутствии.

Взаимодействие абонентов в проколе https сводится к тому, что сначала клиент посылает GET запрос серверу, тот в свою очередь отправляет клиенту свой публичный ключ и сертификат, затем, если сертификат прошел проверку, абоненты генерируют симметричный ключ, которым сервер шифрует страницу, отправленную клиенту, а браузер расшифровывает ее своим симметричным ключом и выводит ее содержимое.

Для реализации атаки «человек посередине» все запросы веб-браузера перенаправляются на специальный TLS/SSL прокси-сервер, чтобы это сделать не предпринимая никаких действий на стороне клиента, следует использовать ARP-спурфинг, в результате которого злоумышленник может представиться искомым маршрутизатором и контролировать сетевой трафик.

Затем TLS/SSL прокси-сервер перехватывает запрос клиента, направляет его серверу, в ответ получает открытый ключ и сертификат сервера, далее прокси сервер и сервер генерируют симметричный ключ и сервер шифрует им страницу и отправляет ее прокси серверу, который расшифровывает ее своим симметричным ключом и записывает в файл, затем прокси-сервер генерирует ложный сертификат и закрытый ключ и отправляет открытый ключ и сертификат клиенту. В случае успеха авторизации браузер генерирует симметричный ключ с прокси сервером, который потом отправляет клиенту страницу, зашифрованную данным ключом, браузер ее расшифровывает и отображает клиенту. Чтобы клиент принял поддельный сертификат следует

установить поддельный корневой сертификат CA сервера в хранилище доверенных центров сертификаций на стороне клиента.

Что же нужно делать для обеспечения безопасного взаимодействия с помощью протокола https? В первую очередь, следует использовать последнюю версию протокола TLS (1.2), а также при согласовании параметров соединения выбирать надежные шифры - AEAD (Authenticated Encryption with Associated Data) Ciphers, которые начинают внедрять с версии TLS 1.2. Примером является AES-GCM, который позволяет шифровать и расшифровывать 128-битные блоки данных с 128, 192 или 256-битным ключом. Так же он поддерживает GMAC для аутентификации сообщений.

Реализация https протокола увеличивает нагрузку на сервер, и, чтобы уменьшить сложность вычислений, при этом не пожертвовав безопасностью, начиная с TLS 1.2 вводят поддержку потокового шифра CHACHA20, основанного на Salsa20, который имеет более высокую криптостойкость при немного большей скорости. В качестве MAC кода начинают использовать Poly1305, который имеет кеш, короче чем у его аналогов, тем самым экономя ресурс сети. Его безопасность определяется выбранным режимом шифрования, это значит, чтобы его взломать, нужно, например, взломать CHACHA20, если используется TLS\_ECDHE\_ECDSA\_WITH\_CHACHA20\_POLY1305\_SHA256 [5].

Чтобы избежать атак «человек посередине» следует проверять IP адрес, а также DNS сервера. И тщательно анализировать сертификат сервера по следующим параметрам:

- является ли сертификат самоподписанным;
- кем подписан данный сертификат (есть ли на нем подпись центра сертификации);
- когда сертификат менялся в последний раз;
- имеют ли другие сервера подобный сертификат.

#### Библиографический список

1. Документация TLS/SSL - [http://www.cryptopro.ru/sites/default/files/docs/TLS\\_description.pdf](http://www.cryptopro.ru/sites/default/files/docs/TLS_description.pdf)
2. Applications to SSL, IPSEC, WTLS Serge Vaudenay
3. All Your Biases Belong To Us: Breaking RC4 in WPA-TKIP and TLS Mathy Vanhoef, Frank Piessens
4. Новостной портал dirty.ru - <https://net.dirty.ru/google-mozilla-i-microsoft-uprazdniat-rc4-v-nachale-2016-goda-830370/>
5. Документация TLS 1.3 - <https://tswg.github.io/tls13-spec/#rfc.appendix.D>

#### УДК 004

**М. А. Ключев, Т. А. Минаева**

студенты кафедры технологий защиты информации

**Н. В. Волошина**

канд. техн. наук, доцент – научный руководитель

### СТЕГАНОАНАЛИЗ СТАТИЧЕСКИХ ИЗОБРАЖЕНИЙ

В данной статье рассмотрены основные задачи и классификация стегосистем, классификация методов стеганоанализа, а также приведены результаты практической реализации одного из методов статистического стеганоанализа, а именно метода анализа распределения пар значений на основе критерия Хи-квадрат.

Стегосистема представляет собой совокупность методов и средств, которые используются для создания скрытого канала передачи информации [1]. К основным ее задачам можно отнести:

- 1) защиту авторских прав на некоторые виды интеллектуальной собственности;
- 2) защиту конфиденциальной информации от несанкционированного доступа;

- 3) преодоление системы мониторинга и управления сетевыми ресурсами;
- 4) камуфлирование программного обеспечения.

По уровню обеспечения стойкости к пассивным атакам стегосистемы можно разделить на теоретически стойкие, практически стойкие и нестойкие [2].

Теоретически стойкие стегосистемы осуществляют сокрытие информации в тех элементах контейнера, значения которых не превышают уровня шума или погрешности квантования. Невозможность различения пустых и заполненных контейнеров таких систем должны быть теоретически доказуема [2]. Под контейнером для сокрытия информации понимается любой файл, структура и размер которого позволяют скрыть необходимые данные.

Возможность выявления стеганоcontainers, созданных практически стойкими стегосистемами, не исключается, однако, на данный момент противник не располагает необходимыми для этого ресурсами.

Для нестойких стегосистем существуют стеганоаналитические методы, позволяющие выявить факт эксплуатации системы [2].

Самым распространенным, но наименее стойким к обнаружениям методом стеганографического сокрытия, является метод замены наименее значимых бит (НЗБ, LSB). Идея метода заключается в замене от одного до четырех битов в байтах цветового представления точек данного изображения битами скрываемого сообщения. Такой метод применяется к растровым изображениям, которые представлены в формате без компрессии – BMP [3]. Сокрытие в JPEG-файлах выполняется аналогичным способом, но вместо цветовых составляющих изменяются квантованные дискретные косинусные коэффициенты.

В статье [1] предлагается следующая классификация методов стеганоанализа:

- 1) В зависимости от количества информации, доступной аналитику:

- направленные;
- универсальные.

- 2) По критерию цели атаки:

- статические;
- динамические;
- вспомогательные.

- 3) В зависимости от объекта поиска в контейнерах:

- визуальные;
- сигнатурные;
- статистические.

Мы рассмотрим один из статистических методов - метод анализа распределения пар значений на основе критерия Хи-квадрат.

Для оценки эффективности статистических методов было реализовано приложение на языке C++, осуществляющее анализ изображений формата BMP на наличие скрытой информации тремя методами, описанными далее. Информация, представляющая собой псевдослучайный набор нулей и единиц, заносилась в изображение последовательно в последний бит каждой цветовой составляющей. Информация была занесена в восемь тестовых изображений: air-plane.bmp, earth\_from\_space.bmp, house.bmp, lena.bmp, mandrill.bmp, peppers.bmp, sail-boat\_on\_lake.bmp, san\_diego.bmp. Однако, перед тем, как применять статистические методы, необходимо оценить уровень шума, создаваемый таким встраиванием.

Первая оценка – абсолютная – среднеквадратичное отклонение реального сигнала, описывающего изображение, от полезного [4]:

$$Nrms = \sqrt{\frac{\sum_{i=0}^{k-1} (B_i - A_i)^2}{k}},$$

где  $N_{rms}$  – среднеквадратичное отклонение реального сигнала, описывающего изображение, от полезного;  $B_i$  – значение  $i$ -го элемента заполненного контейнера;  $A_i$  – значение  $i$ -го элемента пустого контейнера;  $k$  – количество элементов в контейнере.

На основании абсолютной оценки может быть вычислена относительная оценка - пиковое отношение полезного сигнала к шуму  $PSNR$  (peak-to-peak signal-to-noise ratio) [4], измеряемое в децибелах:

$$PSNR = 20 \log_{10} \left( \frac{A_{\max}}{N_{rms}} \right),$$

где  $PSNR$  – пиковое отношение полезного сигнала к шуму;  $A_{\max}$  – максимальное значение сигнала.

Максимальное значение сигнала в полноцветном BMP-изображении – 255. После встраивания информации описанным выше способом значение  $PSNR$  не превысило 52 дБ. Типичные значения  $PSNR$  для сжатия изображений лежат в пределах 30–40 дБ. Так как это отношение полезного сигнала к шуму, то, чем выше значение  $PSNR$ , тем меньше шума создает встраивание.

Метод анализа распределения пар значений на основе критерия Хи-квадрат [5] и основан на знании того, что младшие биты изображений не являются случайными и частоты двух соседних элементов контейнера должны находиться достаточно далеко от значения частоты среднего арифметического этих элементов. В «пустом» изображении ситуация, когда частоты элементов со значениями  $2N$  и  $2N + 1$  близки по значению, встречается достаточно редко. При встраивании информации данные частоты сближаются или становятся равными (рис. 1). Идея атаки хи-квадрат заключается в поиске этих близких значений и подсчете вероятности встраивания на основе того, как близко располагаются значения частот четных и нечетных элементов анализируемого контейнера к среднему значению пары.

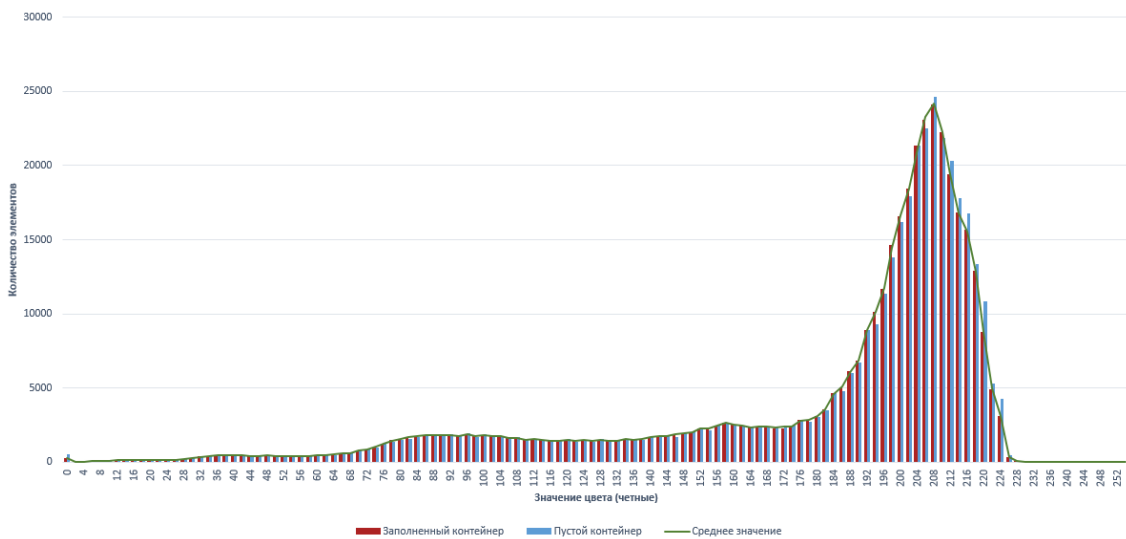


Рис.1. Гистограмма пустого и заполненного контейнеров изображения airplane.bmp

Из гистограммы видно, что частоты четных элементов в парах заполненного контейнера (левые столбцы) расположен ближе к линии среднего, чем аналогичные частоты четных элементов незаполненного контейнера (правые столбцы).

Теоретически ожидаемая частота элементов пары была рассчитана по следующей формуле:

$$n_i^* = \frac{|\{\text{colour} | \text{sortedIndex}(\text{colour}) \in \{2i, 2i + 1\}\}|}{2},$$

где  $n_i^*$  – теоретически ожидаемая частота появления элементов  $i$ -й пары.

Эмпирическая частота появления четных элементов контейнера была вычислена следующим образом:

$$n_i = |\{\text{colour} | \text{sortedIndexOf}(\text{colour}) = 2i\}|,$$

где  $n_i$  – эмпирическая частота появления четного элемента  $i$ -й пары.

На основании полученных значений была вычислена статистика хи-квадрат с  $k-1$  степенью свободы:

$$\chi^2_{k-1} = \sum_{i=1}^k \frac{(n_i - n_i^*)^2}{n_i^*},$$

где  $\chi^2_{k-1}$  – статистика хи-квадрат с  $k-1$  степенью свободы;  $k$  – количество пар значений элементов контейнера.

После получения статистики хи-квадрат была вычислена вероятность встраивания информации в изображение:

$$p = 1 - \frac{1}{2^{\frac{k-1}{2}} \Gamma\left(\frac{k-1}{2}\right)} \int_0^{\chi^2_{k-1}} e^{-\frac{x}{2}} x^{\frac{k-1}{2}-1} dx,$$

где  $p$  – вероятность встраивания информации в изображение.

Сравнение работы метода при применении к пустому контейнеру, контейнеру с полным заполнением и контейнеру, заполненному на 50%, представлено на графике (рис. 2).

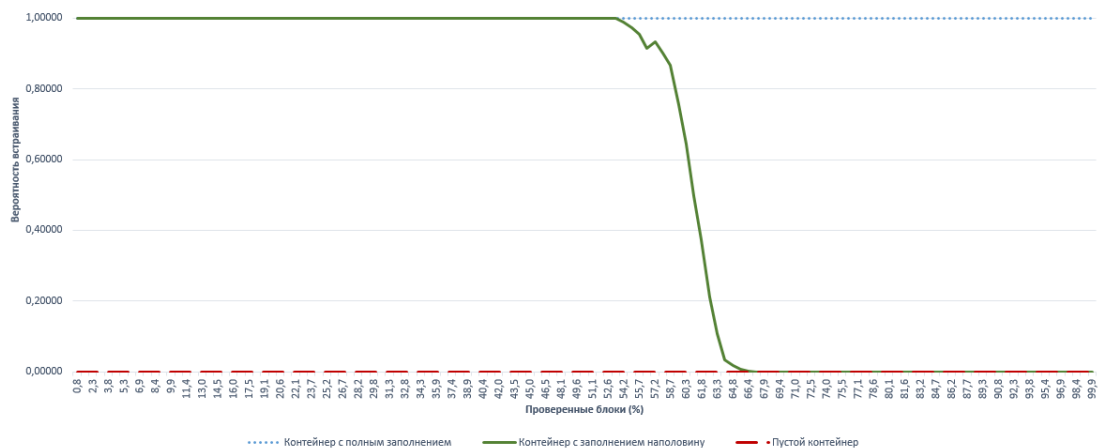


Рис. 2. График вероятность встраивания информации в изображение airplane.bmp

Изображение размером 512 на 512 пикселей анализировалось блоками по 2000 пикселей, после каждого блока вычислялась статистика по проанализированной части изображения и вычислялась вероятность встраивания. Как видно из графика, вероятность встраивания в пустом контейнере (штрих) лежит на нулевой отметке, вероятность в заполненном контейнере (пунктир) – на единичной отметке. При заполнении контейнера на 50% вероятность встраивания начинает убывать при проверке 53% изображения – после того, как в статистике накопится достаточное число неизмененных элементов.

Рассмотренный метод показал хорошие результаты при выявлении стеганографического скрывания информации в изображениях различных типов, таких как пейзажи, портреты, компьютерная графика и натюрморты. Для проверки действенности метода также были предприняты

попытки вызова ложного срабатывания путем создания искусственного шума, компрессии и декомпрессии изображений. Вызвать ложное срабатывание таким образом не удалось. Для более надежной оценки вероятности встраивания информации следует использовать данный метод в комплексе с другими методами как статистическими, так и методами других категорий. В качестве обхода данного метода может быть рассмотрено использованием адаптивных алгоритмов, не изменяющих распределение элементов изображения.

### Библиографический список

1. Генне О. В. . Основные положения стеганографии// Защита информации. Конфидент. 2000. №3
2. Кошкина Н. В. Обзор и классификация методов стеганоанализа// Фундаментальные и прикладные проблемы Computer Science. 2015.
3. Грибунин В. Г. Цифровая стеганография / В. Г. Грибунин. СПб.: СОЛОН-Пресс, 2002. 280 с.
4. Лапшенков Е. М. Незатонная оценка уровня шума цифрового изображения на основе гармонического анализа// Компьютерная оптика. 2012. Том 36. №3. С. 439–447.
5. Westfeld A. Attacks on Steganographic Systems: Breaking the Steganographic Utilities EzStego, Jsteg, Steganos and STools-and Some Lessons Learned / A. Westfeld, A. Pfitzmann // 3rd International Workshop on Information Hiding (2000).

УДК 004.852

**В. К. Козыревский**

студент кафедры информационной безопасности

**А. И. Веселов**

ассистент – научный руководитель

### СПОСОБ УМЕНЬШЕНИЯ СЛОЖНОСТИ ОБУЧЕНИЯ АЛГОРИТМА ВИАЛЫ–ДЖОНСА

Распознавание лиц является одной из самых изучаемых задач в таких областях как компьютерное зрение, биометрия, организация видеоконференций, создание интеллектуальных систем безопасности и контроля доступа и т. д. В основе этой задачи лежит обучение компьютера распознавать объекты. В свою очередь распознавание объектов – очень трудная задача как с практической, так и с теоретической точек зрения. В отличие от человеческого мозга, машина не способна отличить за доли секунды один объект от другого. 100% .

Метод детектирования лиц с использованием алгоритма Виолы-Джонса основан на следующих идеях [1]: интегральном представлении изображения, методе построения классификатора на основе алгоритма адаптивного бустинга (*Adaboost*) и методе комбинирования классификаторов в каскадную структуру (последний в рамках статьи описываться не будет). Эти идеи позволяют построить детектор способный работать в режиме реального времени. За счет своей высокой скорости и эффективности, алгоритм Виолы–Джонса является одним из популярных методов для поиска области лица на изображениях. Одним из существенных недостатков алгоритма является высокая сложность обучения. В данной статье будут рассмотрены основные аспекты алгоритма и предложен метод снижения сложности его обучения.

Перед описанием алгоритма *Adaboost* рассмотрим постановку задачи обучения в общем. Допустим, дано конечное множество прецедентов (объектов), по каждому из которых собраны некоторые данные – описание прецедента. В свою очередь, совокупность таких описаний называется обучающей выборкой [2]. Задача обучения заключается в поиске по обучающей выборке взаимосвязей (зависимостей), присущих для всех прецедентов. Это означает, что необходимо составить свод правил, указывающий, как действовать при том или ином событии, предварительно показав алгоритму на обучающей выборке, как вести себя для конкретных прецедентов. Для описания прецедентов чаще всего используется признаковое пространство – совокупность

из  $N$  показателей, присущая всем прецедентам. В рамках данной работы будут рассматриваться только числовые признаки.

Применительно к задаче классификации предположим, что имеется пространство, в котором находятся объекты двух классов с метками  $\{-1;1\}$ . Тогда задача обучения сводится к разделению этого пространства прямыми так, чтобы отделить объекты одного класса от объектов другого класса. Одна такая прямая будет называться слабым классификатором, а композиция из слабых классификаторов, в свою очередь, – сильным классификатором (рис. 1).

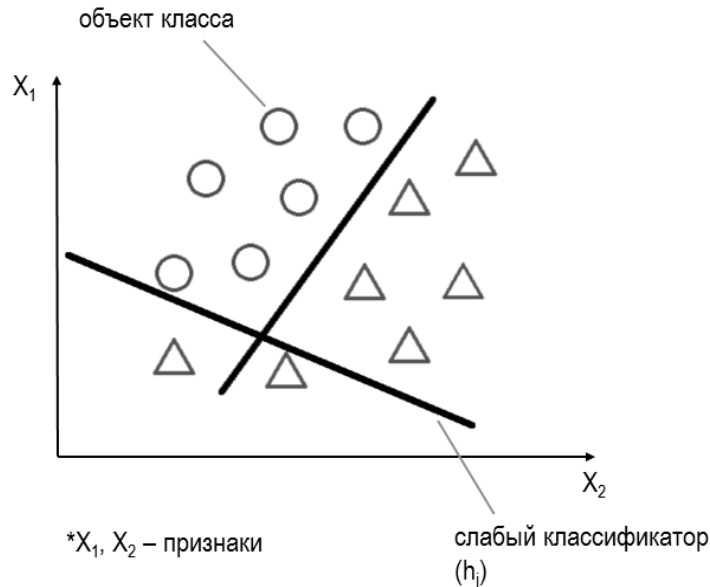


Рис. 1. Пример композиции слабых классификаторов

Рассмотрим подробнее алгоритм Adaboost, лежащий в основе детектора Виолы-Джонса. Adaboost является итеративным алгоритмом адаптивного бустинга: на каждой последующей итерации обновляется вектор весов (показателей степени важности того или иного объекта для классификатора) таким образом, чтобы добавить в композицию слабый классификатор, исправляющий ошибки, допущенные ранее выбранными слабыми классификаторами (рис. 2).

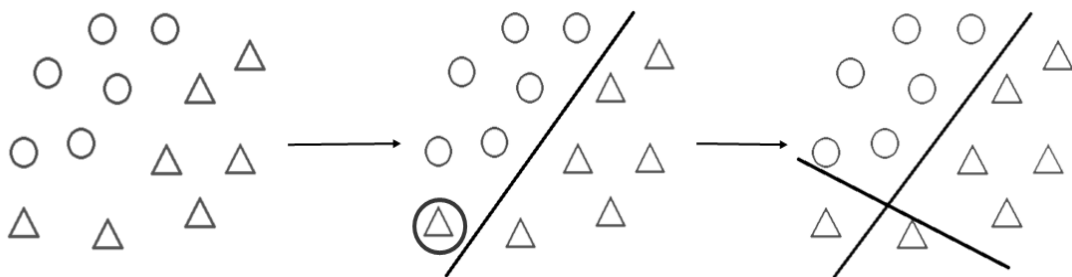


Рис. 2. Адаптивный бустинг

Для описания прецедентов, как уже было сказано, используется, так называемое, признаковое пространство. В алгоритме Виолы-Джонса используются признаки, известные, как признаки Хаара – прямоугольные маски, состоящие из черно-белых областей. Выбор пал на них с учетом наблюдения, что область вокруг глаз светлее области на носу, область рта темнее, чем область вокруг рта, и т. д. Области на масках выбираются из тех же соображений (рис. 3).



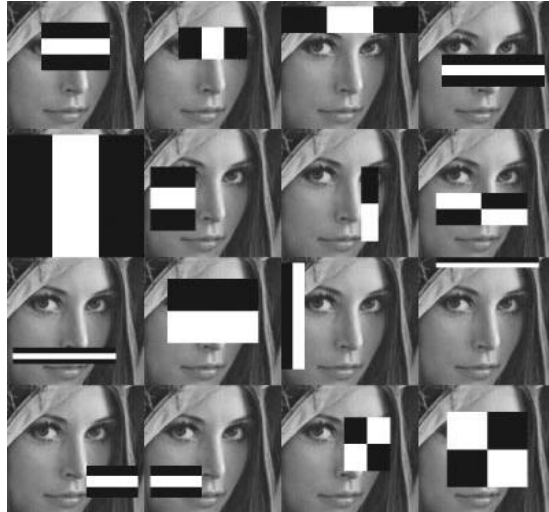


Рис. 3. Признаки Хаара

Каждый признак характеризуется размером светлой и темной областей, пропорциями, а также минимальным размером самой маски. Признаки Хаара дают точечные значения перепада яркости по оси. Отклик такого признака вычисляется следующим образом:

$$f = W - B, \quad (2)$$

где  $W$  – сумма значений яркостей точек, закрываемых светлой областью маски, а  $B$  – сумма значений яркостей точек, закрываемых черной областью маски.

Тогда слабым классификатором будет:

$$h_j(I) = \begin{cases} 1, & \text{если } \rho_j f_j(I) < \rho_j \theta_j \\ 0, & \text{иначе} \end{cases}, \quad (1)$$

где  $I$  – изображение,  $j$  – индекс признака,  $\theta_j$  – пороговое значение,  $\rho_j$  – значение паритета, позволяющее всегда использовать при сравнении отклика с порогом знак меньше,  $f_j(I)$  – отклик  $j$ -го признака на изображение  $I$ .

Для эффективного расчета признаков авторами алгоритма было предложено использовать интегральное представление изображения. Интегральное представление изображения можно описать как матрицу, размеры которой совпадают с размерами изображения, а значение каждого элемента которой является суммой пикселей, находящихся левее и выше соответствующего пикселя в изображении. Расчет каждого элемента матрицы интегрального представления изображения можно произвести по следующей формуле

$$I(x, y) = \sum_{i=0, j=0}^{i \leq x, j \leq y} I(r, c), \quad (3)$$

где  $I(r, c)$  – пиксель исходного изображения.

Вернемся к задаче обучения и поговорим о выборе слабого классификатора. Этот аспект является важнейшим в алгоритме Виолы-Джонса. Фактически, на каждой итерации алгоритма необходимо выбрать из всего множества слабых классификаторов один, обеспечивающий минимальную ошибку классификации. Это означает, что нужно перебрать все возможные признаки со всеми возможными порогами и выбрать из них наилучший с точки зрения взвешенной ошибки классификации. При размере изображения  $19 \times 19$  пикселей существует порядка 65 тыс. признаков Хаара для конкретной маски. Пусть размер тренировочной выборки составляет 10000 изображений, а число масок равно 10. В таком случае количество классификаторов, для которых

необходимо подобрать оптимальную границу равно  $65000 \cdot 10 = 6,5 \cdot 10^5$ . Подбор оптимальной границы для каждого слабого классификатора требует прохода по всем 10 тыс. изображений.

Для уменьшения сложности подбора границы для одного классификатора существует алгоритм *Fast Feature Select (FFS)* [3], который перебирает не все возможные границы, а только некоторое «допустимое» подмножество. Предположим, существует  $N$  тестовых изображений. Эти изображения имеют значения откликов  $v_1 \dots v_N$  для конкретного признака  $j$ . Для начала, отсортируем значения признаков  $v_{i_1} \dots v_{i_N}$ . Тогда, если верно  $v_{i_k} \leq \theta_1, \theta_2 \leq v_{i_{k+1}}$  для одного целого  $k$  и двух различных  $\theta_1$  и  $\theta_2$ , выбор границы признака для отклика, равного  $\theta_1$ , будет иметь то же самое влияние, что и выбор признака для отклика  $\theta_2$ . Таким образом, не нужно проверять все возможные значения признака. Достаточно проверить  $N + 1$  возможное значение, чтобы найти оптимальное значение границы  $\theta$ . Отметим, что данный алгоритм уменьшает сложность подбора границы, но никак не упрощает перебор всех возможных классификаторов.

Проведем оценку сложности обучения алгоритма Виолы–Джонса. Введем следующие обозначения:  $T$  – количество слабых классификаторов в итоговой композиции,  $M$  – количество всех возможных признаков на одном элементе выборки, а  $N$  – размер обучающей выборки. Тогда видно, что сложность выбора всех возможных признаков составляет  $O(MN)$ . Сложность алгоритма FFS за счет сортировки составляет  $O(N \log N)$ . Итоговая сложность обучения равна  $O(TM + N \log N + MN)$ .

Отметим, что, как правило, количество всех возможных признаков на одном элементе выборки превышает остальные параметры и значительно влияет на оценку сложности алгоритма. Следовательно, есть необходимость каким-то образом уменьшить это число и тем самым снизить сложность всего алгоритма обучения. С учетом всего вышесказанного можно сделать следующее наблюдение. Некоторые наборы признаков, имеющие разную форму, размеры и координаты, могут давать в конкретных областях значения откликов близкие по значению, а некоторые – обратные по значению. Продемонстрировать данное наблюдение можно построив графики зависимости откликов от элементов выборки (рис. 4, 5).

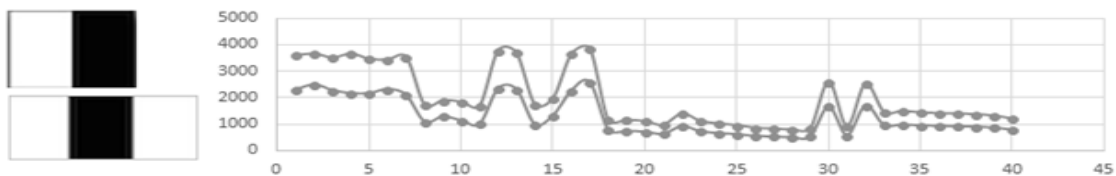


Рис. 4. Распределение значений признаков с похожими масками

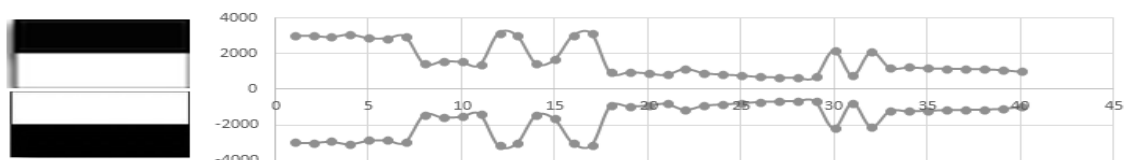


Рис. 5. Распределение значений признаков с разными масками

Пользуясь этим наблюдением, можно на одной итерации откинуть большую часть признаков и не перебирать те признаки, отклики которых коррелируют с откликами ранее выбранных признаков, так как их ошибки также будут коррелировать. Таким образом, перед обучением алгоритма необходимо составить матрицу признаков, по которой будет видно, какие признаки дают похожие по значению отклики, а от каких можно избавиться при дальнейшем обучении. Подбо-

ную матрицу достаточно составить единожды и использовать ее для изучения алгоритма, что снизит сложность обучения.

Взяв во внимание предложенное упрощение можно оценить сложность обучения как  $O(TM' + N\log N + M'N)$ , где  $M' \ll M$  – количество всех возможных признаков на одном элементе выборки с учетом использования предложенной матрицы.

Подведем итоги. В данной статье была рассмотрена задача обучения сильного классификатора в алгоритме Виолы-Джонса, проведена оценка сложности обучения и предложен способ ее снижения с  $O(TMN^2 + MN)$  до  $O(TM' + N\log N + M'N)$ ,  $M' \ll M$ .

В дальнейшем планируется реализация предложенного метода и проведение исследования эффективности распознавания в зависимости от значения  $M'$ . Также, необходимо предложить способ эффективного построения матрицы, показывающей корреляцию между откликами признаков.

#### Библиографическое описание:

1. Viola P., Jones M. Robust Real-time Object Detection // International Journal of Computer Vision. 2004. Vol. 57, № 2, pp. 137-154.
2. Вьюгин В. В. Математические основы теории машинного обучения. Москва, 2013, 387 с.
3. Wu. J, Brubaker S., Mullin M., Rehg. J. Fast Asymmetric Learning for Cascade Face Detection // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2008. Vol. 30, № 3, pp. 369–382.

УДК 004.057.4

**И. А. Миронова**

студент кафедры технологий защиты информации

**Н. В. Волошина**

канд. техн. наук, доц. – научный руководитель

### АНАЛИЗ ПРОТОКОЛА ПЕРЕДАЧИ FCoE

Сегодня сети используют различные протоколы для передачи данных между устройствами. Традиционно Ethernet представляет собой фреймовые компьютерные сетевые технологии для локальных вычислительных сетей (LAN), а оптоволокно (Fibre Channel) используется для сетей хранения данных (SAN). В 2009 году появился новый протокол - Fibre Channel over Ethernet (FCoE). Целью моей работы является рассмотреть данный протокол данных. Актуальность состоит в том, что FCoE один из новых этапов развития сетевой технологии и технологии хранения, поэтому необходимо выявить его преимущества и недостатки по сравнению с другими популярными протоколами.

Fibre Channel over Ethernet - протокол, определенный стандартом в комитете T11 (комитет в составе International Committee for Information Technology Standards— INCITS—отвечающий за Fibre Channel). FCoE инкапсулирует пакеты Fibre Channel в заголовки пакетов IP. Это позволяет пересылать их по традиционным сетям Ethernet, сохраняя при этом протокол Fibre Channel. Так как системы FCoE обычно делят полосу пропускания канала передачи данных с традиционным серверным трафиком, они подключаются к портам Ethernet с быстродействием 10 Гбит/с [1].

FCoE – это в определенном смысле объединение двух сетевых технологий. Коммутаторы FCoE должны поддерживать обе и иметь консолидированный ввод-вывод. В устройстве реализованы коммутационные функции как Ethernet, так и FC. [2]

Для определения преимуществ и недостатков данного протокола рассмотрим другие протоколы: iSCSI и Fibre Channel (таблица).

Сравнение iSCSI, Fibre Channel и FCoE

	iSCSI	Fibre Channel	FCoE
Описание	Протокол, который базируется на TCP/IP и разработан для установления взаимодействия и управления системами хранения данных, серверами и клиентами	Семейство протоколов для высокоскоростной передачи данных. FC используется как стандартный способ подключения к системам хранения данных уровня предприятия	Протокол переносит фреймы Fibre Channel через Ethernet, инкапсулируя кадры Fibre Channel в jumbo frames Ethernet-a
Максимальный размер устройства	64ТБ	64ТБ	64ТБ
Максимальное количество устройств	256	256	256
Производительность	iSCSI может работать поверх 1Gb или сети 10Gb TCP / IP. Несколько соединений могут быть объединены в один сеанс, установленный между инициатором и целью.	1Gb / 2Gb / 4Gb / 8Gb и 16Gb HBAs. Протокол влияет на центральный процессор хоста, потому что хост-адаптер шины управляет большей частью обработки	Протокол требует 10Gb Ethernet. Нет IP- инкапсуляции данных как iSCSI. Протокол требует jumbo кадров, так как FC нужна нагрузка 2.2K в размерах и это не может быть фрагментировано.
Безопасность	Реализует Challenge Handshake протокол аутентификации (CHAP), чтобы гарантировать, что initiator и target доверяют друг другу.	Некоторые FC-коммутаторы поддерживают концепцию в VSAN, чтобы изолировать части инфраструктуры хранения данных. Сети VSAN концептуально похожи на VLAN. Зонирование между хостами и FC target также предлагает степень изоляции	Некоторые FC-коммутаторы поддерживают концепцию в VSAN, чтобы изолировать части инфраструктуры хранения данных. Зонирование между хостами и FCoE target также предлагает степень изоляции

Итак, FCoE предлагает идентичный набор возможностей и абсолютно совместим с существующими сетями FC SAN. Отличие можно найти в том, что происходит замена физического уровня Fibre Channel на 10GbE. Преимущества в производительности FCoE над iSCSI все еще требуют подтверждений. Оба протокола используют 10GbE, но TCP/IP может увеличить задержки для iSCSI, поэтому у FCoE появляется небольшое преимущество в сходной среде.

#### Библиографический список

1. DeSanti C., Jiang J. Fcoe in perspective //Proceedings of the 2008 International Conference on Advanced Infocomm Technology. ACM, 2008. С. 138.
2. Jiang J., DeSanti C. The role of FCoE in I/O consolidation //Proceedings of the 2008 International Conference on Advanced Infocomm Technology. ACM, 2008. С. 87.

УДК 004.056.5

**С. В. Нестерук**

студент кафедры технологий защиты информации

**С. В. Беззатеев**

д-р техн. наук, доцент – научный руководитель

## ОБЗОР ПОДХОДОВ ЛИНГВИСТИЧЕСКОЙ СТЕГАНОГРАФИИ

### Введение

Стремительное развитие информационных технологий с каждым годом повышает требования информационной безопасности. Некоторые из задач, такие как скрытая передача данных и создание цифровых водяных знаков, решает стеганография. На сегодняшний день это развивающаяся наука, не имеющая единой теории.

Стеганография – наука и искусство написания скрытых сообщений таким образом, что никто, кроме предполагаемого получателя, не знает о существовании письма. Стеганография скрывает одно сообщение внутри другого (контейнера), например, обычного звука, графики или другого файла. Цель стеганографии – доставка сообщения, скрывая само существование обмена информацией [1].

В наше время исследователи сосредоточились на стегосистемах, прячущих сообщения в графических, звуковых и видеофайлах путем их незаметной модификации. Однако у таких систем есть недостатки: пользователям сложно придумать предлог, по которому бы они могли регулярно обмениваться фотографиями или иными стегоконтейнерами, а также у таких систем большая избыточность. Иное дело, если бы скрытые сообщения можно было встроить в сам текст. Именно этим и занимается лингвистическая стеганография [2].

Лингвистическая стеганография определяется как совокупность приемов и методов, позволяющих скрывать любую цифровую информацию в рамках текстов, основанных на некоторых лингвистических знаниях. Чтобы скрыть сам факт сокрытия конечный текст должен не только оставаться неприметным внешне, т. е. содержать обычный текст с орфографией, морфологией, синтаксисом, лексикой и порядком слов, соответствующим смыслу текста, но и сохранить смысловую сплоченность и грамматическую правильность [1].

### Классификации лингвистической стеганографии

На сегодняшний день существуют различные подходы к классификации видов лингвистической стеганографии и ее методов. В работе [3] представлены такие виды лингвистической стеганографии, как семаграммы и открытое кодирование. Текстовые семаграммы скрывают сообщение, изменяя тип или размер шрифта, добавляя дополнительные пробелы к тексту-контейнеру. А открытое кодирование скрывает сообщение в открыто читаемом тексте, который нужно читать согласно заранее заданным правилам.

Выделяются синтаксические и семантические методы лингвокриптографии [4]. К синтаксическим относится изменение пунктуации, изменение стиля и структуры текста, а также графематические методы. А семантические объясняются, как методы замены синонимами и синонимичными фразами.

В текстовой стеганографии выделяются три группы [5]:

- 1) стеганография, основанная на форматировании текста;
- 2) случайная и статистическая генерация;
- 3) лингвистические методы стеганографии.

Из них к лингвистической стеганографии можно отнести лишь последние два типа.

С точки зрения лингвистической стеганографии, в моделировании текста можно выделить следующие уровни: статистический, синтаксический, лексико-семантический и онтологический.

Тогда проблему лингвистической стеганографии можно переформулировать так: как создать стеготекст, характеризующийся на каждом из этих уровней свойствами, типичными для обычного, «естественного» текста? При этом текст, который не вызовет подозрений на предыдущем уровне, не обязательно выдержит проверку при оценке на последующем уровне.

Методы современной лингвистической стеганографии нацелены на автоматизацию создания стеготекста. В общем случае, в процедуре создания стеготекста в качестве входных данных используется собственно тайное сообщение  $M$  (традиционно представленное в двоичном коде) и опционально – текст-контейнер  $C$  (любой текст, созданный человеком, зачастую имеющийся в открытом доступе). Применяя знания из лингвистической базы знаний, сообщение  $M$  внедряется в контейнер  $C$ . Возможен также случай, когда текст-контейнер не задан. Тогда процедура создания стеготекста состоит в том, чтобы преобразовать тайное сообщение  $M$  на основе созданной лингвистической базы знаний и разработанных процедур в автоматически сгенерированный связный псевдотекст. Такой псевдотекст, как правило, может удовлетворять требованиям к «естественному» тексту на статистическом и синтаксическом уровне, но семантика, будучи наиболее сложным объектом компьютерного моделирования естественного языка и трудно поддающимся автоматизации, в таких текстах страдает. Если такой текст будет предъявлен человеку, он легко идентифицирует его «нерегулярность», установив наличие нетипичных для обычного текста, созданного человеком, свойств. Поэтому основная задача современной стеганографии заключается в решении вопроса, какова должна быть процедура автоматизированной генерации стеготекста, который в определенном смысле (при оценке на каждом из перечисленных уровней) был бы похож на «естественный» текст, созданный человеком. Именно такой стеготекст, предположительно, не вызовет подозрений у стегоанализатора (человека или машины), и цель стеганографии будет достигнута.

Методы стеганографии, связанные со статистическим уровнем, не используют собственно лингвистические свойства текстов. Результирующий текст должен подражать некоторому реальному тексту, написанному на естественном языке, с точки зрения статистического распределения лингвистических единиц в нем. В качестве лингвистических единиц могут выступать, например, символы или слова.

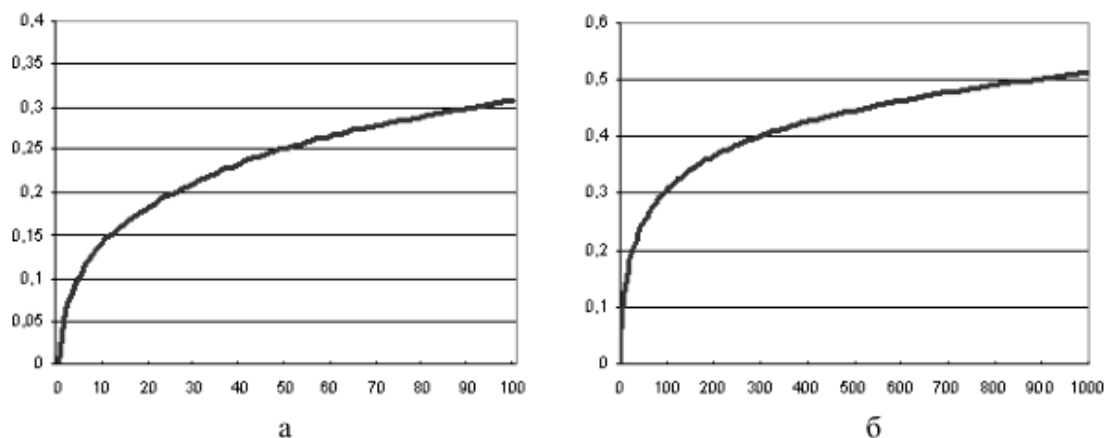
При генерации стеготекста для преодоления недостатков посимвольной генерации, в частности отсутствия приемлемой для естественного языка синтаксической структуры, а также наличия фиктивных слов, предлагается подход на основе применения контекстно-свободной грамматики – понятия, которое было введено Н. Хомским для объяснения функционирования языка. Основная идея контекстно-свободной грамматики Хомского заключается в замене слов исходного текста словами той же части речи из некоторого словаря по некоторым правилам.

При таком подходе каждая из синтаксических структур (тип предложения, порядок слов в предложении) соотнесена с двоичным кодом, и в зависимости от того, какая следующая последовательность бит сообщения  $M$  должна быть закодирована, для предложения текста-контейнера  $C$  используется трансформация с соответствующим кодом. Трансформированное предложение включается в стеготекст  $T$ . Ненадежность такого подхода с точки зрения стегоанализа заключается в том, что здесь не учитываются требования сохранения естественного для языка статистического распределения лингвистических единиц: некоторые синтаксические конструкции довольно редки в языке, и их частое появление в тексте может свидетельствовать о вмешательстве с целью передачи тайного сообщения [5].

Лингвокриптография на основе метода замены синонимов

В работе [6] базовым в лингвокриптографии является метод замены синонимов. Принцип его работы заключается в том, что мы определяем логический «0» или «1» по тому, была ли совершена в тексте замена определенного слова на его синоним или нет. Одна из сложностей метода заключается в том, что слово не всегда можно заменить синонимом, не искажая контекста и стиля. Чтобы избежать ошибок следует ограничивать словарь синонимов.

Словарь синонимов можно составить из наиболее распространенных слов языка. Анализ текстов (русского языка) показал, что словарь из 100 наиболее употребляемых слов обеспечивают покрытие более 30% любого осмысленного текста (рисунок, а), а словарь из наиболее употребляемых 1000 слов – более 50% (рисунок, б).



Поскольку для корректного использования слов (в русском языке) важно правильно согласовывать их окончания, в словаре синонимов следует хранить не только сами синонимы, но также их всевозможные окончания [6].

#### Классификация лингвистической стеганографии на основе входных параметров

На базе изученных материалов я решил составить свою классификацию, в которой методы делятся по обязательным входящим параметрам. Я выделил группы методов с заданным контейнером, с заданным шаблоном и с генерируемым контейнером.

В методы группы с заданным контейнером помимо самого тайного сообщения и контейнера в общем виде входит словарь либо некоторые правила. Например, в методе замены синонимов это будет словарь синонимов. В методе изменения порядка слов – собственно правила, по которым мы восстанавливаем исходное сообщение исходя из того какой порядок слов используется в предложении. А в методе добавления вводных конструкций необходимо использовать как словарь вводных слов и фраз, так и правила из подстановки.

В группе методов с заданным шаблоном вместо контейнера используется шаблон, по которому контейнер будет строиться. Помимо уже упомянутого синтаксического метода, я бы добавил в эту группу метод формирования визуальной модели прозы. Этот метод многократно применялся писателями 20 века, а первое упоминание о нем принадлежит Льюису Кэрроллу в 1870-х гг. Заключается данный метод в создании блоков текста (музыки или картинок), которые сочетаются между собой в любом порядке. При таком подходе можно шифровать сообщение, располагая текст в соответствующей комбинации [7].

К группе методов с генерируемым контейнером, кроме уже поясненных, вероятно, можно отнести и онтологический, который так или иначе фигурирует в разных источниках, но нигде не пояснен. Очевидно, сегодня использовать этот метод автоматизировано еще довольно сложно, поэтому пока оставим его без формализации.

#### Заключение

В настоящее время метод замены синонимов является самым изученным и отвечает многим требованиям, предъявляемым к стеготексту. Но он связывает нас заданным контейнером и пока не устойчив криптоанализу.

Наряду с ним наиболее интересными для дальнейшего изучения являются методы группы с генерируемым контейнером, а именно лексико-семантический и онтологический. Мы считаем, что именно с их помощью можно будет генерировать стеготекст, который был бы похож на

«естественный» текст, созданный человеком. Именно такой стеготекст, предположительно, не вызовет подозрений у стегоанализатора, и цель стеганографии будет достигнута.

### Список литературы:

1. Обзор различных видов лингвистической стеганографии и применение в защите данных. <http://masters.donntu.org/2014/fknt/sokolenko/library/article10.htm>
2. Лингвистическая стеганография становится более стойкой. <https://www.pgpru.com/novosti/2007/lingvisticheskajasteganografijastanovitsjastojjkojj>
3. Computer Forensics. Investigating data and image files. [http://greysec.ir/ebook/Computer.Forensics.Investigating.Data.and.Image.Files\\_%5Bwww.Graymind.ir%5D.pdf](http://greysec.ir/ebook/Computer.Forensics.Investigating.Data.and.Image.Files_%5Bwww.Graymind.ir%5D.pdf)
4. Лингвистическая стеганография. <http://cyberleninka.ru/article/n/lingvisticheskaya-steganografiya>
5. Лингвистическая стеганография: современные подходы. Часть 1. <http://vestnik.susu.ru/linguistics/article/view/4181/3757>
6. Лингвистическая стеганография на основе замены синонимов для текстов на русском языке. <http://cyberleninka.ru/article/n/lingvisticheskaya-steganografiya-na-osnove-zameny-sinonimov-dlya-tekstov-na-russkom-yazyke>
7. Концепция В. Розанова и ее роль в формировании визуальной модели прозы неклассического типа. [http://vestnik.tspu.edu.ru/files/vestnik/PDF/articles/sem-yan\\_t\\_f\\_85\\_90\\_2\\_76\\_2008.pdf](http://vestnik.tspu.edu.ru/files/vestnik/PDF/articles/sem-yan_t_f_85_90_2_76_2008.pdf)

УДК 004

**С.Ю. Прощенко**

студент кафедры технологий защиты информации

**С.В. Беззатеев**

д-р техн. наук, доцент – научный руководитель

### АНАЛИЗ АЛГОРИТМА ШИФРОВАНИЯ RSA

В данной статье рассмотрен алгоритм шифрования RSA, основные требования к нему, атаки на данный алгоритм и их ограничение.

Опубликованная в ноябре 1976 года статья Уитфилда Диффи и Мартина Хеллмана «Новые направления в криптографии» перевернула представление о криптографических системах, заложив основы криптографии с открытым ключом. Разработанный впоследствии алгоритм Диффи – Хеллмана позволял двум сторонам получить общий секретный ключ, используя защищенный канал связи. Однако этот алгоритм не решал проблему аутентификации. Без дополнительных средств пользователи не могли быть уверены, с кем именно они сгенерировали общий секретный ключ.

Изучив эту статью, трое учёных Рональд Ривест, Ади Шамир и Леонард Адлеман из Массачусетского технологического института (MIT) приступили к поискам математической функции, которая бы позволяла реализовать сформулированную Уитфилдом Диффи и Мартином Хеллманом модель криптографической системы с открытым ключом. После работы над более, чем 40 возможными вариантами им удалось найти алгоритм, основанный на различии в том, насколько легко находить большие простые числа и насколько сложно раскладывать на множители произведение двух больших простых чисел, получивший впоследствии название RSA. Система была названа по первым буквам фамилий её создателей.

В декабре 1997 года была обнародована информация, согласно которой британский математик Клиффорд Кокс, работавший в центре правительственной связи (GCHQ) Великобритании, описал криптосистему, аналогичную RSA, в 1973 г. Но из-за того, что он работал в секретной организации, опубликовать данную работу ему не позволили.



В основе RSA лежит задача факторизации произведения двух простых больших чисел. Для шифрования используется простая операция возведения в степень по модулю  $N$ . Для расшифрования же необходимо вычислить функцию Эйлера от числа  $N$ . Для этого необходимо знать разложение числа  $n$  на простые множители (в этом и состоит задача факторизации).

В RSA открытый и закрытый ключ состоит из пары целых чисел. Закрытый ключ хранится в секрете, а открытый ключ сообщается другому участнику, либо где-то публикуется.

Всё начинается с генерации ключевой пары (открытый, закрытый ключ). Генерация ключей в RSA осуществляется следующим образом:

- 1) Выбираются два простых числа  $p$  и  $q$  (такие, что  $p$  не равно  $q$ );
- 2) Вычисляется модуль  $N = pq$ ;
- 3) Вычисляется значение функции Эйлера от модуля  $N$ :  $\phi(N) = (p-1)(q-1)$ ;
- 4) Выбирается число  $e$ , называемое открытой экспонентой, оно должно быть взаимно простым со значением функции  $\phi(N)$ .

5) Вычисляется число  $d$ , называемое секретной экспонентой, такое, что  $de = 1 \pmod{\phi(N)}$ , т. е. является мультипликативным обратным к числу  $e$  по модулю  $\phi(N)$ .

Итак, мы получили пару ключей:

- пара  $(e, n)$  – открытый ключ;
- пара  $(d, n)$  – закрытый ключ.

От каких параметров зависит стойкость алгоритма RSA? Представим себе, что к Бобу и Алисе присоединяется Ева, которая хочет узнать, какое сообщение послал Боб. Допустим, что у Евы есть открытый ключ Алисы  $(e, N)$ . Для того, чтобы расшифровать сообщение  $c$ , необходимо знать закрытый ключ  $(d, N)$ . Мы знаем, что  $de = 1 \pmod{\phi(N)}$ , однако Ева не знает  $\phi(N) = (p-1)(q-1)$  т.е задача сводится к нахождению простых чисел  $p$  и  $q$  (хотя это не всегда так), которые связаны с известным  $N$  следующим образом  $N = pq$ .

Делаем выводы. Чтобы алгоритм был стойким, необходимо:

1. Выбрать два больших простых случайных числа  $p$  и  $q$  (к примеру,  $\geq 1024$  бита каждое), который должны быть не слишком различными и быть не слишком близкими.
2. Наибольший общий делитель  $(p-1)$  и  $(q-1)$  должен небольшим, в лучшем случае равен двум.
3. Выбрать большое значение открытой экспоненты  $e$ , как правило, выбирают простые числа Ферма: 17, 257, 65537
4. Сохранить в секрете закрытый ключ.

Существует 2 необходимых условия для работы алгоритма RSA:

1) Ассиметричный алгоритм шифрования является стойким, если атакующий имеет два открытых текста  $M_1$  и  $M_2$ , а так же зашифрованный текст  $C_i$ , не может с вероятностью большей, чем  $1/2$  определить к какому из сообщений  $M_1$  или  $M_2$  относится  $C_i$ .

Перейдем к RSA и проверим данное условие. Вспомним ситуацию с Алисой и Бобом, допустим, канал связи прослушивает Ева. Боб спрашивает у Алисы: «Алиса, мы идем сегодня в кино», причем сообщение не шифруется. Алиса отвечает Бобу, но не хочет, чтобы кто-то знал, поэтому шифрует свой ответ на открытом ключе Боба и отправляет шифротекст Бобу. (Предполагается, что Алиса отвечает Бобу монотонно). Ева перехватывает зашифрованное сообщение и знает, что Алиса ответила либо «Да», либо «Нет». Ева располагает открытым ключом Боба, поэтому последовательно шифрует сообщение «Да» и «Нет», соответственно одно из них совпадет с зашифрованным сообщением Алисы и Ева узнает, пойдет ли Алиса сегодня в кино или нет.

Из этого видно, что упрощенное описание алгоритма RSA не годится для практического использования. Как решается данная проблема на практике? Решается эта проблема достаточно

просто: к сообщению добавляется некоторая случайная величина, а затем полученный текст шифруется. Таким образом, если Ева перехватывает сообщение  $C1 = E(ДаA1B2)$ , то зашифровав «Да» и «Нет»:  $C2 = E(Нет)$ ,  $C3 = E(Да)$ , будет видно, что  $C1$ ,  $C2$  и  $C3$  не совпадают.

2) Допустим, у Евы есть две функции, одна  $F1$  шифрует сообщения, вторая  $F2$  расшифровывает шифротекст. Затем Ева генерирует два сообщения  $M1$  и  $M2$ . Затем наугад одно из сообщений шифруется функцией  $F1$ , на выходе функции – шифротекст  $Ci$ .  $Ci$  возвращается Еве, её задача угадать с вероятностью, большей, чем  $1/2$ , к какому из сообщений  $M1$  или  $M2$  принадлежит  $Ci$ . При этом Ева может расшифровать любое сообщение, кроме  $Ci$  (иначе задача лишена смысла). Считается, что криптосистема с открытым ключом стойкая, если злоумышленник не может с вероятностью, большей, чем  $1/2$ , сказать, какому из сообщений соответствует шифротекст.

Проверим второе условие. Пусть у Евы есть два открытых сообщения  $M1$  и  $M2$  и один шифротекст  $M1 \bmod(N)$ . Что делает Ева? Она создает сообщение, используя открытый ключ  $(e, N)$ :  $C^* = 2e C1 \bmod(N)$  затем, используя функцию  $F2$  расшифровывает это сообщение, таким образом:

$M^* = C^* \bmod(N) = 2ed * M1 \bmod(N) = 2M1 \bmod(N)$ , вычисляя  $M/2$ , Ева получит сообщение  $M1$ .

Вышесказанное еще раз показывает, что использовать на практике упрощенный алгоритм RSA нельзя. Решается это проблема почти, также, как и в случае с условием 1. Напомню, что в условии 1 мы добавляли к сообщению абсолютно случайную и непрогнозируемую информацию, а затем шифровали текст. Теперь накладываем еще одно требование: необходимо, чтобы дополнительные блоки помогали определить, был ли шифротекст получен в результате шифрующей функции или он смоделирован злоумышленником.

Рассмотрим наиболее известные атаки на алгоритм шифрования RSA:

1. Суть Атаки Шора заключается в сведении задачи факторизации к задаче поиска периода следующей функции:  $F(x) = a^x \bmod N$ .

Если известен период функции, то факторизация осуществляется при помощи алгоритма Евклида на классическом компьютере. И вот квантовая часть алгоритма факторизации как раз занимается поиском периода функции. А классическая часть алгоритма сначала специальным образом готовит оную функцию, а потом проверяет найденный квантовой частью период на достаточность для решения задачи. Если период найден правильно (алгоритм вероятностный, так что может найти не то, что хочется), то задача решена. Если же нет, то квантовая часть алгоритма прогоняется ещё раз. А, поскольку, проверка правильности решения для задачи факторизации очень проста, то эту часть алгоритма вообще можно не принимать во внимание с точки зрения подсчёта сложности.

Рассмотрим алгоритм факторизации Шора на простом примере. Для этого просто разложим на простые множители некоторое число, являющееся произведением двух и в точности двух простых чисел, ни одно из которых не является числом 2. Факторизуем число 21, главное в рассмотрении то, чтобы хватило вычислительных мощностей.

Для того чтобы факторизовать число при помощи алгоритма Шора, необходимо найти период периодической функции (рис.1), указанной выше, где  $a$  – некоторый параметр, выбираемый произвольно до старта алгоритма, а  $N$  – число, которое необходимо факторизовать. Самое главное ограничение заключается в том, чтобы у чисел  $a$  и  $N$  не было общих делителей, больших 1. Поскольку мы хотим факторизовать число 21, в качестве параметра  $a$  возьмём число 2. Обычно это значение и берут, но иногда необходимо брать другое значение.

<b>x</b>		0	1	2	3	4	5	6	7
<b>a<sup>x</sup></b>	<b>2<sup>x</sup></b>	1	2	4	8	16	32	64	128
<b>a<sup>x</sup> mod M</b>	<b>2<sup>x</sup> mod 21</b>	1	2	4	8	16	11	1	2

Рис.1. Пример нахождения периода

Из этой таблицы видно, что в данном конкретном примере периодом функции является значение  $r = 6$ . Другими словами, чтобы найти период при помощи такой таблицы, необходимо найти минимальное значение  $x$ , большее 1, для которого  $a^x \equiv 1 \pmod{M}$ . Соответственно, алгоритм Шора выигрывает именно на этом этапе, поскольку при помощи модели квантовых вычислений найти период функции можно с полиномиальной сложностью, чего нельзя сделать в рамках классической вычислительной модели.

Далее – дело техники. Множители числа  $M$  определяются как наибольшие общие дели-

тели (НОД) между  $a^{\binom{r}{2}} \pm 1$  и  $M$ . Отсюда следует ещё одно ограничение на значение параметра  $a$ , выбираемого перед запуском алгоритма, – найденный период должен быть чётным, т. е. нацело делиться на 2. Поскольку в рассматриваемом примере  $r = 6$ , данное условие выполнено, так что в качестве  $a^{\binom{r}{2}} \pm 1$  получаются два числа  $-2^3 - 1 = 7$  и  $2^3 + 1 = 9$ . Вычислить наибольшие общие делители можно с помощью уже упоминавшегося алгоритма Евклида, и для чисел 7 и 9 с числом 21 это будет пара (7, 3). Собственно, эта пара и есть разложение числа 21 на простые множители.

В ходе работы был реализован данный алгоритм, который находит простые множители при известном числе  $N$ . Из-за достаточно малой мощности компьютера данная программа работает только с небольшими числами. Ввиду данного ограничения для реализации 4 шага не требуется квантовый компьютер.

2. Атака безключевого чтения позволяет расшифровать шифротекст при следующих известных параметрах: шифротекст и открытый ключ. В алгоритме атаки на шифрованный текст

определяется такой номер шага  $j$ , для которого  $y^{e^j} \pmod{n} = y$ . Из последнего соотношения видим, что при возведении в степень  $e$  значения получается начальный шифротекст. При выполнении алгоритма в степени  $j-1$  получается исходный текст сообщения. Сложность данного алгоритма сводится к вычислительным мощностям машины, на которой он запущен.

В ходе данной работы был рассмотрен алгоритм шифрования RSA, были рассмотрены и обоснованы дополнительные требования к данному алгоритму. Кроме этого, были изучены атаки на данный алгоритм и практически реализованы атака Шора и атака Безключевого чтения RSA. Они были реализованы на слабых мощных компьютерах в целях ознакомления и изучения данных методов. На настоящий момент RSA показывает большую надежность при больших ключах. Но при развитии квантовых компьютеров, данные вычисления будут производиться в разы быстрее, что приведет к снижению актуальности RSA.

**Библиографический список**

1. «Основы криптографии» А.П. Алферов, А.Ю. Зубов, А.С. Кузьмин, А.В. Черемушкин, Москва. «Гелиос АРВ», 2001г.
2. Криптография. Официальное руководство RSA Security. С. Беркет, С.Пэйн.

УДК 004.942

**А. А. Татарникова**

студент кафедры безопасности информационных систем

**А. М. Тюрликов**

д-р техн. наук – научный руководитель

## РАЗРАБОТКА АЛГОРИТМА ОБНАРУЖЕНИЯ ПАТОЛОГИЙ ОПОРНО-ДВИГАТЕЛЬНОГО АППАРАТА С ПОМОЩЬЮ АКСЕЛЕРОМЕТРОВ В МОБИЛЬНЫХ УСТРОЙСТВАХ

В данной работе исследуется возможность обнаружения патологий опорно-двигательного аппарата с применением акселерометров в мобильных устройствах без использования дополнительных измерительных приборов.

Ключевые слова: акселерометры в мобильных устройствах, дискретное преобразование Фурье, заболевания опорно-двигательного аппарата.

Современные сотовые телефоны кроме выполнения своей основной функции выполняют ряд других задач, в том числе так или иначе связанные со здоровьем человека. Например, существуют мобильные приложения, позволяющие измерять простые показатели (такие как давление или частота сердечных сокращений), но при этом они используются с дополнительными измерительными приборами. В пример можно привести такие приложения как: «Wireless Blood Pressure Monitor» компании «Withings» и «Scout» компании «Scanadu». Современные мобильные телефоны снабжены микроэлектромеханическими системами (МЭМС) (такими как акселерометры), которые позволяют измерять ускорения сотового телефона, но при этом они пока используются для достаточно примитивных задач, например, для позиционирования экрана при повороте устройства. Однако если зафиксировать мобильный телефон на теле человека, можно измерять ускорения при движении человека. Эти данные можно использовать как для идентификации человека по походке, так и для обнаружения возможных заболеваний опорно-двигательного аппарата человека, поэтому актуальной является задача разработки алгоритма обнаружения патологий опорно-двигательного аппарата с помощью акселерометров в мобильных устройствах.

Настоящая работа построена следующим образом. Описано два существующих решения: использование акселерометров в мобильных устройствах для решения задач идентификации движущихся объектов; использование специальных акселерометров и других датчиков для обнаружения заболеваний опорно-двигательного аппарата. На основе этих двух решений предложен способ обнаружения патологий опорно-двигательного аппарата с помощью акселерометров в мобильных устройствах. Описываются эксперименты, которые проводились для проверки предложенного способа. В заключение описываются направления будущих исследований.

В работе [1] с помощью акселерометров в мобильных устройствах идентифицируют классы транспортных средств.

В работе [2] рассматриваются различные способы оценки функционального состояния опорно-двигательного аппарата применением информативных параметров. Для решения данной задачи используются как разнообразные напольные датчики, так и акселерометры, крепящиеся непосредственно к нижним конечностям испытуемых. Однако во всех рассматриваемых методах требуются специальные громоздкие устройства. Например, датчики акселерометра, закрепленные на конечностях, с помощью провода присоединяются к прибору, выполняющему быстрое преобразование Фурье, который в свою очередь подсоединен к персональному компьютеру.

В работах [3–5] исследуется идентификация человека по походке с помощью акселерометров в мобильных устройствах.

Используя идею, представленную в работе [2], предлагается следующий подход к построению системы: при возникновении заболевания опорно-двигательного аппарата в частотном спектре возможны изменения (например, появление пиков на высоких частотах). По походке че-

ловека, с помощью мобильного приложения, которое будет фиксировать показания акселерометра в мобильном телефоне, можно обнаруживать патологии опорно-двигательного аппарата и отправлять соответствующие данные лечащему врачу.

Чтобы проверить данное предположение, был проведен ряд экспериментов, которые проходили следующим образом:

1. Мобильный телефон фиксировался на человеке, после чего испытуемый в течение, приблизительно, 15 с должен был пройти своей обычной походкой по прямой. В течение этого времени происходила запись данных с трех осей акселерометра. Отсчеты брались с частотой 100 Гц (каждые 10 мс).

2. Далее данные, полученные с акселерометра в мобильном телефоне, обрабатывались с помощью дискретного преобразования Фурье (ДПФ).

3. После этого, полученные спектры подвергались анализу на качественном уровне.

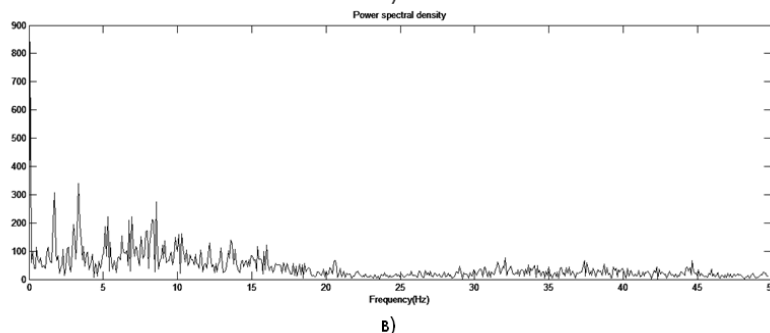
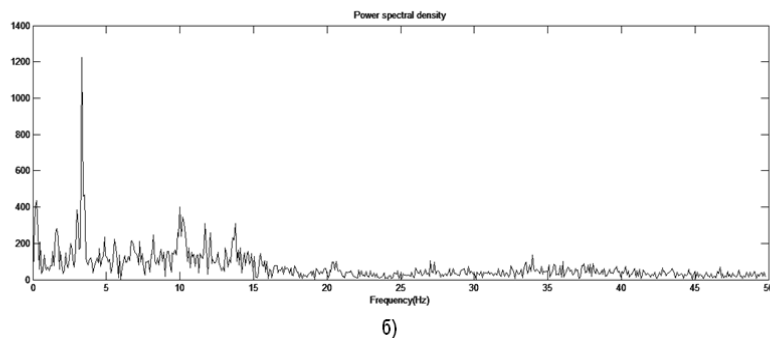
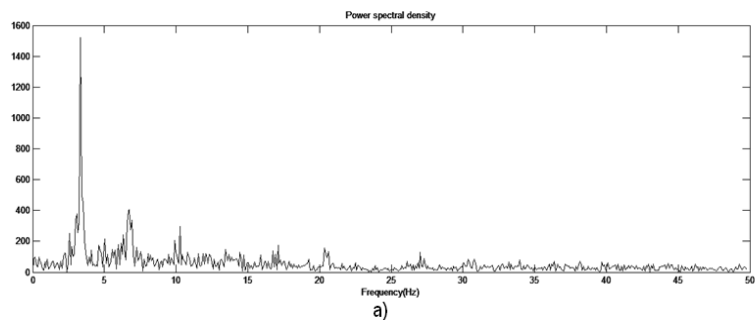
Всего было исследовано 8 испытуемых.

Были выявлены следующие типовые особенности в спектрах (рисунок):

1. Максимальные амплитуды колебаний по всем осям акселерометра у большинства испытуемых находятся на одних и тех же частотах.

2. У большинства испытуемых наиболее сильные колебания происходят по направлению вверх-вниз.

3. Также было выявлено, что с возрастом у человека появляются более сильные высокочастотные колебания вне зависимости от наличия заболеваний опорно-двигательного аппарата.



Типовой спектр – испытуемый №2 (23 года, пол – м): а – ось в направлении вверх-вниз, б – ось в направлении вперед-назад, в – ось в направлении вправо-влево

Характерные особенности в спектрах были выявлены у двоих испытуемых:

1. У испытуемого №1 (21 год, пол – м, заболевание опорно-двигательного аппарата – сколиоз третьей степени) – максимум колебаний по направлению вперед-назад смещен в область высоких частот.

2. У испытуемого №8 (80 лет, пол – м, заболевание опорно-двигательного аппарата – коксартроз) более высокая амплитуда колебаний выражена в направлении вправо-влево.

В настоящем исследовании отсчеты брались с самой высокой частотой, которую допускает API операционной системы мобильного телефона, хотя сами датчики могут брать отсчеты с куда более высокой частотой. Именно поэтому одной из основополагающих целей является проведение предварительных экспериментов для выяснения возможности обнаружения патологий опорно-двигательного аппарата с помощью акселерометров в мобильных устройствах без применения дополнительных измерительных приборов.

В данной работе было сделано следующее. Проведен ряд экспериментов, результаты которых подтвердили возможность разработать алгоритм, обнаруживающий патологии опорно-двигательного аппарата с помощью акселерометров в мобильных устройствах.

Дальнейшие исследования планируется осуществлять в следующих направлениях:

1. Уточнить характеристики процесса, на которых основывается решение о возможном наличии заболевания опорно-двигательного аппарата.

2. Уточнить алгоритм обработки данных, в частности, сколько требуется высокочастотных составляющих в спектре для медицинской диагностики, чтобы, в перспективе, уменьшить количество передаваемой информации.

#### **Библиографический список:**

1. Бардов В.М., Оберттов Д.Е. Алгоритм идентификации классов транспортных средств при помощи акселерометров// Информационно-управляющие системы. 2012. №5. С. 15–18.
2. Жилиев А.А. Биомеханические и электрофизиологические критерии оценки функционального состояния опорно-двигательного аппарата нижних конечностей: автореф. дис. ... д-ра техн. наук: 01.02.08/ Центр. научно-исследовательский ин-т травматологии и ортопедии им. Н. Н. Приорова. М., 2003. 34 с.
3. Kwapisz J.R., Weiss G.M., Moore S.A. Cell phone-based biometric identification// 2010 Fourth IEEE International Conference on Biometrics: Theory Applications and Systems (BTAS)/ IEEE. Washington, 2010. P. 1–7.
4. Derawi M.O., Nickel C., Bours P. et al. Unobtrusive user-authentication on mobile phone using biometric gait recognition// Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IHMSP)/ IEEE. Darmstadt, 2010. P. 306–311.
5. Yan L., Yue-e L., Jian H. Gait recognition based on MEMS accelerometer// 2010 IEEE 10th International Conference on Signal Processing (ICSP)/ IEEE. Beijing, 2010. P. 1679–1681.

**УДК 004**

**С.С. Ткачев**

студент кафедры технологий защиты информации

**С.В. Беззатеев**

д-р техн. наук, доцент – научный руководитель

### **ОБЗОР И АНАЛИЗ УЯЗВИМОСТЕЙ RFID-СИСТЕМ**

В данной статье рассмотрены основные задачи RFID-систем, области применения, возможные угрозы и методы борьбы с ними.

RFID (Radio Frequency Identification) приходит на смену идентификации с помощью штрих-кодов. Радиочастотная идентификация уже внедрена во многие сферы жизнедеятельности человека и продолжает развиваться, имея большие перспективы.

RFID-система состоит из следующих компонентов:

1) RF-подсистема, которая выполняет идентификацию и обеспечивает связь транзакций, используя беспроводную связь. Компоненты RF-подсистемы: RFID-метки (маленькое устройство, прикрепляемое к объекту или вставляемое в него; каждая метка имеет уникальный идентификатор, а также может иметь другие свойства), RFID-считыватели (устройство, которое с помощью беспроводной связи соединено с меткой для того, чтобы идентифицировать предмет, к которому прикреплена данная метка);

2) Enterprise-подсистема содержит в себе вычислительную систему, которая работает на специальном ПО, хранит, обрабатывает и анализирует данные, полученные с RF-подсистемы, чтобы сделать их полезными для бизнес-процессов;

3) Inter-enterprise-подсистема служит для соединения enterprise-подсистем в случаях, когда информацию необходимо распространить между частями организации [2].

Общий принцип работы любой RFID-системы достаточно прост: считыватель излучает в окружающее пространство электромагнитную энергию, метка (транспондер) принимает сигнал от считывателя и формирует ответный сигнал, который принимается антенной считывателя, обрабатывается его электронным блоком и по интерфейсу направляется в компьютер (вычислительную систему) [1] (рис. 1).

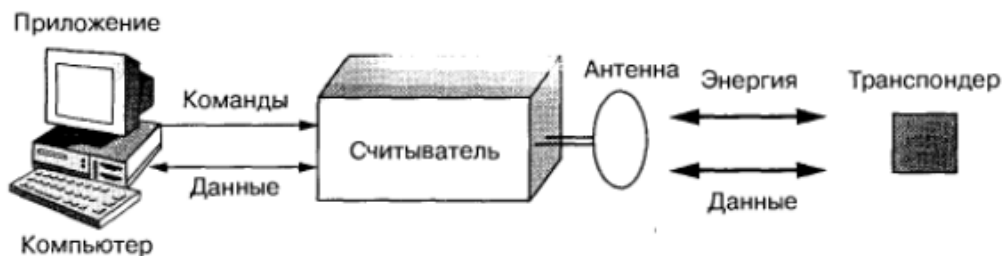


Рис.1. Общая схема радиочастотной идентификации

При этом в поле действия считывателя может находиться сразу несколько меток, при их одновременном ответе радиосигналы будут складываться друг с другом, вследствие чего возникают коллизии. Стандарт EPC Gen-2 определяет, что считыватель управляет множеством радиочастотных меток при помощи трех процедур:

1. Выбор – процедура, посредством которой считыватель выбирает множество меток для инвентаризации или криптографически вызывает множество меток для последующей аутентификации. Процедура выбора производится с помощью команд Select и Challenge.

2. Инвентаризация – процедура, посредством которой считыватель идентифицирует метки. Он начинает процедуру инвентаризации, выдавая команду Query в одном из четырех сеансов. Одна или более меток могут передавать ответ. Устройство опроса обнаруживает одиночный ответ и определяет единичную метку.

3. Доступ – процедура, посредством которой считыватель взаимодействует (считывает или записывает данные, проводит аутентификацию) с определенной меткой, которая должна быть однозначным образом опознана и идентифицирована до осуществления доступа. Доступ предполагает выполнение нескольких команд [3].

Перед командами должен идти сигнал заголовка или синхронизации фрейма. Заголовок должен предшествовать команде Query и обозначать начало цикла инвентаризации, в остальных случаях передачи начинаются сигналом синхронизации фрейма. Сигнал заголовка состоит из начального разделителя фиксированной длины 12,5 мкс, символа данных «0», символа калибровки линии связи считыватель – метка и метка – считыватель. Символ калибровки равняется сумме длительности символа «0» и длительности символа «1». Сигнал синхронизации фрейма, в отличие от сигнала заголовка, в своем составе не имеет символа калибровки линии связи метка – считыватель[3].

Ниже приведены примеры применения RFID-систем:

1. Электронный контроль доступа и перемещений персонала на территории предприятий;
2. Управление производством, товарными и таможенными складами, магазинами, выдачей и перемещением товаров и ценностей;
3. Автоматический сбор данных на железных дорогах, платных автомобильных дорогах, на грузовых станциях и терминалах;
4. Общественный транспорт: управление движением, оплата проезда и оптимизация пассажиропотоков;
5. Обеспечение безопасности (в комплексе с другими техническими средствами).
6. Защита и сигнализация на транспортных средствах [1].

Различают следующие виды рисков при использовании RFID-систем.:

1. угроза бизнес-процессу. прямые атаки на rfid-системы потенциально могут разрушить бизнес-процесс;
2. угроза информации. злоумышленник может получить доступ к информации, что может навредить организации, которая использует rfid-системы;
3. угроза приватности (личных данных, которые злоумышленник может получить, в том числе, и с различных меток путем комбинирования данных);
4. внешний риск (электромагнитная радиация и компьютерные атаки)[2].

Организациям, использующим RFID-системы, необходимо уметь минимизировать риски и понимать их источники. Для предотвращения таких рисков существуют различные способы их минимизации и контроля.

Политика использования описывает авторизованное и неавторизованное использование RFID-технологий в организации и задачи персонала. Политика IT-безопасности ограничивает сетевой трафик между RF- и Enterprise-подсистемами и между Enterprise-подсистемами и общественными сетями, управляет паролями, предусматривает тренировки безопасности RFID для системных администраторов и операторов, управляет связанными криптографическими системами, включая сертификацию прав и управления ключами. Хорошо разработанные политики снижают бизнес-риски, связанные с использованием RFID-технологий, но наличие политик не может гарантировать их соблюдение. Политики должны сочетаться с реализацией и исполнением соответствующих оперативных и технических средств управления, чтобы быть эффективными.

Формальные соглашения для связанных организаций распределяют роли и обязанности каждой. Обычно это документируется в Memorandum of Agreement (MOA) или Memorandum of Understanding (MOU). В них описываются, в том числе, и правила генерации, хранения и распределения паролей. Такие соглашения снижают возможность сбоев в защите.

Контроль физического доступа, включающий заборы, ворота, стены, закрывающиеся двери, турникеты, камеры наблюдения и охранников, не даст злоумышленникам подобраться близко к компонентам RFID-системы. Но и при этом остается вероятность риска в том случае, если один из сотрудников, имеющих доступ, окажется злоумышленником. Уменьшить риск угрозы в данном случае поможет распределение обязанностей среди как можно большего числа сотрудников таким образом, чтобы каждый работник взаимодействовал хотя бы еще с одним. Данный подход также способствует уменьшению количества ошибок, поскольку «второй» может заметить ошибку, которую допустил «первый».

В тех случаях, когда злоумышленник может подобраться к предмету с меткой и считать информацию о нем и организации, будет полезным использование формата идентификатора, который не позволяет получить такие данные. При этом организация теряет преимущества, которые ей предоставляет стандартный формат, например, при разработке базы данных в Inter-Enterprise-подсистемах поиск и запрос происходят намного легче, если идентификатор дает информацию о том, где находится предмет.



Контроль технического уровня не позволяет или затрудняет злоумышленнику встраиваться в приемо-передающий канал обмена между считывателем и меткой. Самый простой способ защиты данных на метке от несанкционированного доступа это защита команд на доступ к памяти, чтение и запись паролем. При этом генерация паролей должна быть случайна и назначение их меткам должно происходить в безопасной среде, чтобы избежать перехвата.

Для обеспечения целостности обмениваемых сообщений может использоваться код аутентификации, использующий хэш-функции (HMAC). Для этого метка и считыватель делятся общим секретным ключом, который также может быть использован для односторонней или совместной аутентификации. Однако недостаток заключается в том, что HMAC-ключ хранится на метке и, если злоумышленник получит к нему доступ, он сможет клонировать метку. Для более надежного доказательства подлинности используются цифровые подписи. Считыватель подписывает метку в цифровом виде с отметкой о времени и событием, связанным с ним, чтобы обеспечить отказ транзакций по метке. Полученные подписи хранятся на метках или в базе данных Enterprise-подсистемы. В этом случае на метке не хранится криптографический секрет, считыватель генерирует открытый и закрытый ключи и производит все необходимые вычисления и операции.

Когда необходимо скрыть прямой канал от прослушивания злоумышленником и при этом он не сможет прослушивать обратный канал, используется метод cover-coding (рис. 2). Считыватель отправляет сообщение на метку, запрашивая ключ, метка генерирует ключ и возвращает его считывателю, теперь считыватель может зашифровывать текст путем применения «исключающего или» с ключом. А метка сможет восстановить сообщение. [2]

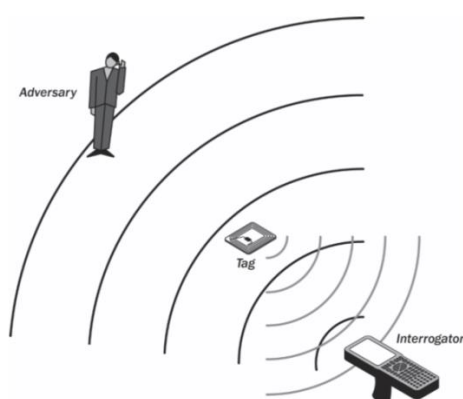


Рис.2. Cover-coding

Если же и обратный канал подвержен опасности, как и прямой, необходимо шифровать данные прямо на метке перед началом отправки. Такие сложные криптографические системы вводят дополнительную нагрузку не только на вычислительную способность метки, но и на потребляемую ею энергию. Метки, способные обеспечивать данный способ безопасности являются наиболее дорогостоящими. Также могут возникать неприемлемые задержки в системах RFID, которые требуют более быстрых транзакций.

Кроме сокрытия информации в радиочастотном канале для понижения рисков, можно работать с самим сигналом, например, с частотой, на которой происходит идентификация. Селекция радиочастот помогает избежать или уменьшать влияние радиопомех. Электромагнитное экранирование вовсе не позволяет распространяться сигналу. Оно может различаться по форме и размерам в зависимости от применения, как антисчитывающий материал на проездных документах или не пропускающие сигнал стены.

Чтобы метки было труднее обнаружить, их можно временно деактивировать: это очень полезно в тех случаях, когда известно время обращения легального считывателя к метке. Тогда

полностью предотвращаются несанкционированные транзакции в периоды бездействия метки и продлевается срок службы батареи.

Некоторые метки обладают функцией защиты от несанкционированного вскрытия, которая помогает предотвратить изменение меток злоумышленником или удаление их от объектов, к которым они прикреплены. Например, использование хрупких или легко ломающихся антенн; если такая метка удаляется от предмета, разрывается электрическое соединение с антенной и метка оказывается в нерабочем состоянии. Оператор должен заметить такое отключение и принять соответствующие меры.

Когда метка становится ненужной, ее функциональность может быть отключена навсегда с помощью удаленной команды kill. Такая возможность снижает риск угрозы конфиденциальности после того, как метка вышла за рамки предполагаемой функциональной среды.

FRID-системы подвержены различным рискам в ходе их эксплуатации, и в зависимости от действующих факторов требуют определенных и отличных друг от друга методов защиты. Таким образом, полагаясь лишь на отдельные средства предосторожности, невозможно обеспечить минимизацию всех возможных рисков. Наиболее полная защита может достигаться только использованием защитных средств в совокупности.

### Библиографический список

1. Барсуков В.С. RFID или не RFID. ВОТ В ЧЕМ ВОПРОС// Специальная Техника. 2005. №6
2. Guidelines for Securing Radio Frequency Identification (RFID) Systems/ Tom Karygiannis, Bernard Eydt, Greg Barber и др. National Institute of Standards and Technology, 2007. 154 с.
3. Ассоциация автоматической идентификации «ЮНИСКАН/ГС1 РУС» / СТАНДАРТ ГС1 РУС, 2015. 243 с.

### УДК 004.057.4

**Д. В. Толопило**

студент кафедры технологий защиты информации

**С. В. Беззатеев**

д-р. техн. наук, доц. – научный руководитель

### SPDY: ЭКСПЕРИМЕНТАЛЬНЫЙ ПРОТОКОЛ ОТ GOOGLE

SPDY (произносится как «SPeeDy») – это экспериментальный протокол прикладного уровня, представленный корпорацией Google в 2009 году, основной целью которого является снижение времени загрузки веб-страниц и их элементов.

Как известно, на сегодняшний день основными протоколами, обеспечивающими веб-функциональность, являются HTTP и TCP. TCP является надежным транспортным протоколом, обеспечивающим гарантированную доставку, контроль и предотвращение перегрузки. HTTP это протокол прикладного уровня, обеспечивающий обмен сообщений по схеме «запрос-ответ».

К сожалению, HTTP очень чувствителен к задержкам. Кроме того, веб-страницы, передаваемые сегодня, значительно отличаются от веб-страниц, существовавших 15 лет назад, и требуют улучшения протокола. Основными особенностями HTTP, препятствующими оптимальной производительности, являются:

- односторонний запрос на подключение;
- исключительно иницируемые клиентом запросы;
- несжатые и избыточные заголовки запроса и ответа;
- дополнительное сжатие данных.

Проект SPDY определяет и реализует протокол прикладного уровня для Интернета, который значительно уменьшает время ожидания. Целями протокола SPDY являются:

1. Сокращение времени загрузки страницы на 50%.
  2. Минимизация сложности развертывания. SPDY использует TCP в качестве базового транспортного уровня, поэтому не требует никаких изменений в существующей сетевой инфраструктуре.
  3. Устранение необходимости каких-либо изменений в содержании по авторам сайта. Только изменения, необходимые для поддержки SPDY в приложениях пользователя агента клиента и веб-сервера.
  4. Разрешение множества одновременных HTTP-запросов для запуска через один сеанс TCP.
  5. Определение протокола, который легко реализовать и сделать сервер эффективным.
- Основные особенности SPDY:
- иницилируемые сервером потоки: spdy облегчает доставку контента клиенту без запроса клиента, используя продвинутое технологии потоков, иницилируемых сервером. доступно две опции: server push и server hint;
  - параллельные мультиплексированные потоки: spdy обеспечивает неограниченное количество одновременных потоков через одно соединение tcp;
  - запрос приоритетности: существующие в spdy запросы приоритетности позволяют преодолеть проблемы взаимных помех, назначая каждому запросу метку приоритета, что позволяет обрабатывать бесконечное количество запросов от клиентов;
  - сжатие: spdy сжимает http заголовки запросов и ответов, что облегчает передачу количества пакетов и байтов.

Для сравнения скорости HTTP- и SPDY-протоколов, было загружено 25 лучших веб-сайтов (таблица). Загрузка повторялась 10 раз для каждого сайта, после чего высчитывалось среднее время загрузки страницы каждого сайта, а также всех веб-сайтов в общем.

Среднее время загрузки страниц лучших 25 сайтов

Сравнение скорости HTTP- и SPDY- протоколов	DSL: 2 Мбит/с загрузка, 375 кбит/с отдача		Кабель: 4 Мбит/с загрузка, 1 Мбит/с отдача	
	Среднее время, мс	Прирост Скорости, %	Среднее время, мс	Прирост скорости, %
HTTP	3111,916		2348,188	
SPDY основное многодоменное соединение / TCP	2242,756	27,93%	1325,46	43,55%
SPDY основное однодоменное соединение / TCP	1695,72	45,51%	933,836	60,23%
SPDY однодоменное соединение + server push / TCP	1671,28	46,29%	950,764	59,51%
SPDY однодоменное соединение + server hint / TCP	1608,928	48,30%	856,356	63,53%
SPDY основное однодоменное соединение / SSL	1899,744	38,95%	1099,444	53,18
SPDY однодоменное соединение + приоритизация / SSL	1781,864	42,74%	1047,308	55,40%

Как видно из данных, представленных в таблице, прирост скорости над HTTP составил 27% – 60% от времени загрузки страницы по обычному TCP (без SSL) и 39% – 55% по SSL.

В настоящее время компания Google отказывается от SPDY в пользу HTTP/2. Основной задачей создания HTTP/2, в основу которого положен протокол SPDY, является повышение эффективности использования сетевых ресурсов и снижение задержек при соединении и обмене данными между клиентом и сервером в условиях современных реалий, при которых для загрузки сайта требуется отправить множество отдельных запросов для получения CSS, файлов JavaScript и картинок. Протокол HTTP/1.1, в силу конвейерной передачи данных в рамках одного соединения и высоких накладных расходов на отдачу ресурсов небольшого размера, не может обеспечить должную эффективность. HTTP/2 решает отмеченные проблемы благодаря таким особенностям, как бинарный протокол передачи данных, мультиплексирование и распараллеливание потоков в рамках одного TCP-соединения, сжатие HTTP-заголовков, приоритизация потоков, согласование расширений между клиентом и сервером, поддержка технологии Server push.

Таким образом, SPDY стал первым масштабным проектом реформирования HTTP, целью которого в первую очередь было ускорение работы веб-сайтов и приложений путем модификации традиционных способов приема и отправки запросов. HTTP/2, представленный шестью годами позже, во многом основывается на SPDY.

### Библиографический список

1. The Chromium Projects: SPDY (<https://www.chromium.org/spdy/spdy-whitepaper>).
2. Google Developers: SPDY protocol (<https://developers.google.com/speed/spdy/>).
3. Stenberg D. HTTP2 explained (<https://daniel.haxx.se/http2/>).
4. W3Techs: Usage of SPDY (<https://w3techs.com/technologies/details/ce-spdy/all/all>).

УДК 004.627

**Н.В. Устюжанин**

студент кафедры инфокоммуникационных систем

**М.Р. Гильмутдинов**

доцент, к.т.н. – научный руководитель

## АНАЛИЗ МЕТОДОВ МОДЕЛИРОВАНИЯ ФОНА ДЛЯ СИСТЕМЫ СЖАТИЯ ВИДЕОПОСЛЕДОВАТЕЛЬНОСТЕЙ

Аннотация. Предложены способы использования моделей фона для сжатия видеопоследовательностей, полученных с камер видеонаблюдения, продемонстрирована их эффективность.

Во многих структурах используются системы видеонаблюдения, однако современные стандарты сжатия видеопоследовательностей не ориентированы на данные системы. Из-за применения стандартов сжатия видеопоследовательностей, не ориентированных на системы видеонаблюдения, временная избыточность видеопотока с камер видеонаблюдения устраняется неэффективно. Так как камеры видеонаблюдения неподвижны, а в кадре, как правило, происходит малая активность, можно уменьшить временную избыточность за счет использования моделей фона.

Моделирование фонового кадра  $B_t$  производится по некоторой предыстории из  $N$  кадров:

$$B_t = F(I_{t-1}, I_{t-2}, \dots, I_{t-N}) \quad (1)$$

При использовании различных способов моделирования фонового кадра возникают разные артефакты. Рассмотрены четыре способа получения модели фона:

усреднение:

$$B_t(x, y) = \frac{1}{N} \sum_{i=1}^N I_{t-i}(x, y), \quad (2)$$

где  $I_t$  – некоторый кадр видеопоследовательности;  $B_t$  – модель фона;

2) медиана – элемент с индексом  $\frac{N}{2}$  упорядоченной выборки.

3) экспоненциальное сглаживание:

$$B_t(x,y) = (1-\alpha)I_t(x,y) + \alpha B_{t-1}(x,y), \quad (3)$$

где  $\alpha$  – коэффициент забывания.

4) смесь гауссовских распределений:  $Y_t(x,y) \sim N(m_i, \sigma_i^2)$ , где  $i \in [1, K]$ ,  $K$  – количество наборов параметров.

Способы получения данных моделей фона описаны в статье [2]. Визуальное сравнение фоновых кадров, полученных с помощью данных способов приведено на рис. 1.

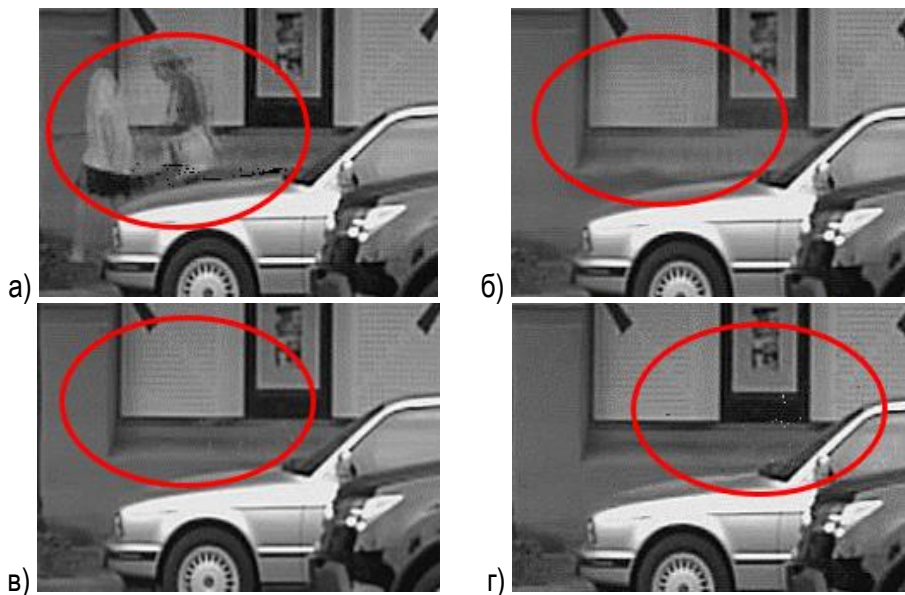


Рис. 1. Сравнение фоновых кадров, полученных с помощью разных способов моделирования фона: а – экспоненциальное сглаживание; б – медиана; в – усреднение, г – смесь гауссовских распределений.

Для анализа эффективности применения моделирования фона в задаче сжатия видеоданных была использована типовая модель видеокодера представленная на рис. 2.

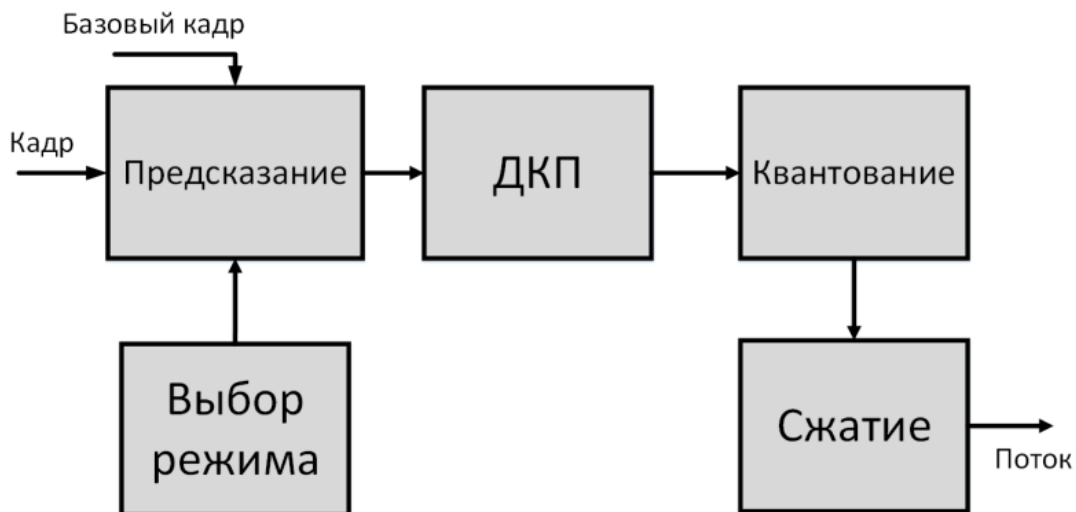


Рис. 2. Типовая модель видеокодера видеопоследовательностей

В реализации данной модели:

- используются GOP-структуры (в данной работе размер GOP равен 100 кадрам) с типами кадров I и P;
- каждый кадр разбивается на блоки размером 8x8 пикселей;
- для каждого блока выполняется выбор режима (Inter/Intra);
- в качестве базового кадра используется предыдущий;
- вычисляется разность с базовым кадром для Inter-моды;
- разностные блоки подвергаются ДКП, квантованию. Квантованные коэффициенты сжимаются арифметическим кодером с применением контекстных моделей.

Для анализа эффективности применения моделей фона типовая модель системы сжатия видеопоследовательностей была модифицирована. На иллюстрациях ниже – серые блоки – элементы типового видеокodeка, белые – модификации. В типовую схему видеокodeка было встроено два новых блока:

1. обновление модели фона (способы обновления зависят от модели, описаны выше);
2. классификация блока на передний план и фон.

Применение данных модулей в схеме сжатия возможно на разных этапах. В данной работе рассмотрено три различных подхода:

1. Фон обновляется на стороне кодера при захвате очередного кадра. На сторону декодера информация об изменении фона передается в битовом потоке ( $\Delta VM$ ). Информация для обновления фона передается только для тех блоков, которые требуется обновить. Пиксель с координатами  $x, y$  обновления фона вычисляется по формуле  $\Delta VM(x, y) = VM_t(x, y) - VM_{t-1}(x, y)$ , где  $t$  – номер текущего кадра.

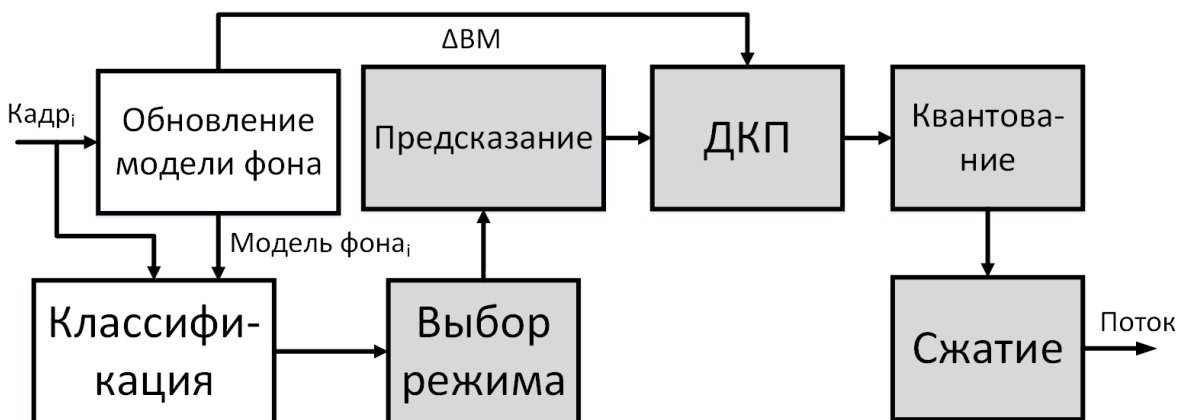


Рис. 3. Схема 1. Обновлением фона при захвате очередного кадра, запись информации для обновления фоновой модели в поток

2. Сжатие первых  $n$  кадров по типовой схеме (рис. 4). Для моделирования фона используются первые  $N$  восстановленных кадров. Фон фиксируется и используется с  $N + 1$  кадра до конца GOP. На стороне декодера используется аналогичный алгоритм моделирования, который использует те же данные, что и кодер (рис. 4, а).

3. Сжатие информации переднего плана, используя фоновый кадр, сформированный при обработке предыдущего кадра. Обновление фона осуществляется для каждого кадра по восстановленным данным, как на кодере, так и декодере (Рис. 4, б).

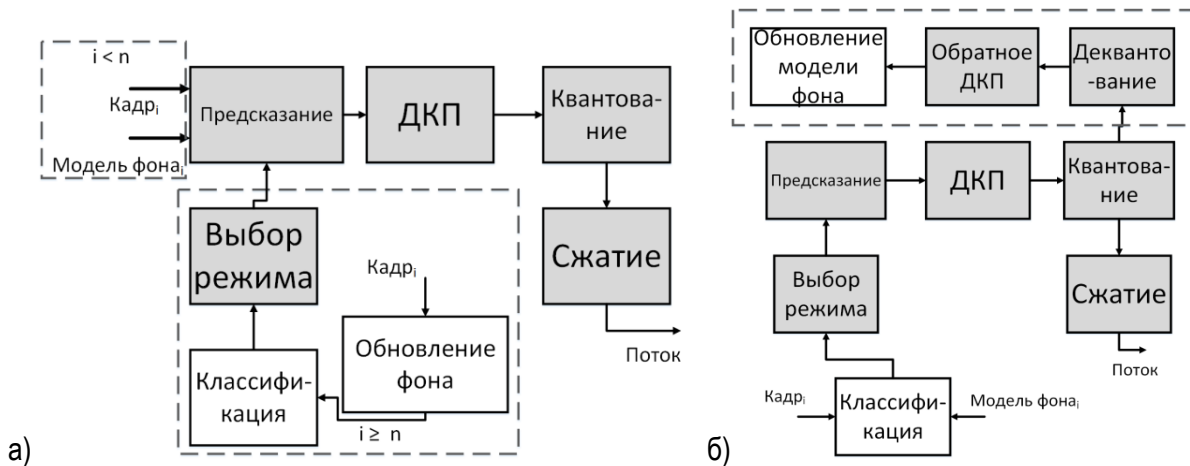


Рис. 4. Схемы 2 и 3: а – сжатие первых  $n$  кадров для моделирования фонового кадра, б – использование фонового кадра, полученного на предыдущей итерации. Информацию для обновления фонового кадра не нужно добавлять в поток, моделирование фона производится на кодере и декодере.

Предсказания строятся на анализе схожести блока текущего кадра с блоками фонового или предыдущего кадров:

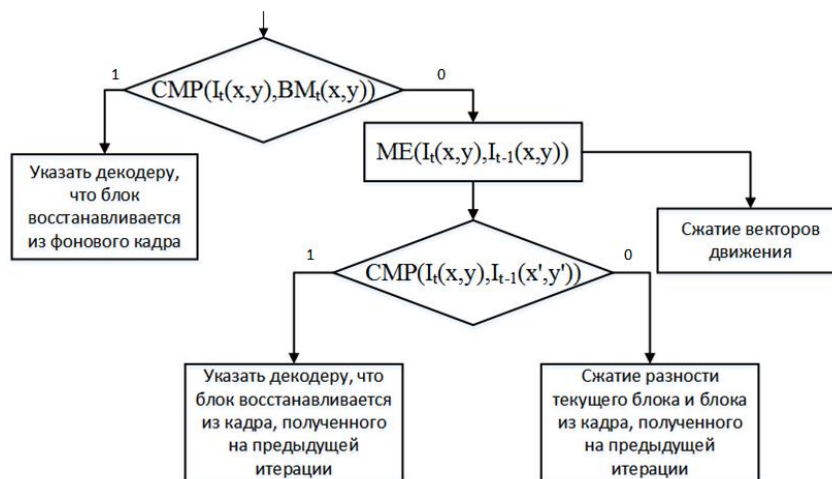


Рис. 5. Алгоритм устранения временной избыточности

Пояснения к рисунку: если блок текущего кадра классифицируется как фоновый, то в поток будет положен бит, указывающий, что блок нужно восстановить из фонового кадра. Иначе будет производиться оценка движения ( $ME$  – англ. Motion Estimation [4]) между текущим кадром, и кадром, восстановленным ранее. После чего производится сжатие векторов движения с помощью арифметического кодера с контекстным моделированием по аналогии со стандартом H.264[3]. Если блок текущего и предыдущего кадров схожи, то в поток будет положен бит, указывающий что блок нужно восстановить из предыдущего кадра по вектору движения. Иначе будет произведено сжатие разности. Классификация блоков вычисляется по формуле 4.

$$CMP(I_t(x,y), I_{t-i}(x,y)) = \begin{cases} 1, & MSE(I_t(x,y), I_{t-i}(x,y)) \leq T \\ 0, & MSE(I_t(x,y), I_{t-i}(x,y)) > T \end{cases}, (4)$$

где  $T$  – порог классификации; в работе  $T=2$  из учета, что  $PSNR(2) \approx 45\text{дБ}$  для 8битных компонент изображения.

Для сравнительного анализа схем с использованием моделирования фона были использованы две видеопоследовательности:

1. низкого разрешения (hall, 352x288 пикселей);
2. высокого разрешения (street traffic, 1920x1080 пикселей).

Сравнение эффективности моделирования фона производилось с помощью следующих критериев [5]:

– Полнота (recall):

$$R = \frac{TruePositive}{TruePositive + FalseNegative}$$

– Точность (precision):

$$P = \frac{TruePositive}{TruePositive + FalsePositive}$$

– Критерий Фишера:

$$K\Phi = \frac{2 * precision * recall}{precision + recall}$$

Данные – объем данных, затрачиваемых на сжатие информации для восстановления видеопоследовательности в килобайтах

Таблица 1

Сравнение моделей фона (видео высокого разрешения)

Модель	Точность	Полнота	КФ	Данные (схема 1) (кб)	Данные (схема 2) (кб)	Данные (схема 3) (кб)
Усреднение	0.997	0.996	0.997	2359	2951	<b>2868</b>
Экспоненциальное забывание	<b>0.997</b>	<b>0.998</b>	<b>0.998</b>	<b>1968</b>	<b>2903</b>	3023
Медиана	0.997	0.997	0.997	2357	3151	3066
Смесь гауссовских распределений	0.997	0.995	0.996	2159	3137	3143

Таблица 2

Сравнение моделей фона (видео низкого разрешения)

Модель	Точность	Полнота	КФ	Данные (схема 1) (кб)	Данные (схема 2) (кб)	Данные (схема 3) (кб)
Усреднение	0.959	0.965	0.962	161.7	90.12	167.4
Экспоненциальное забывание	<b>0.960</b>	<b>0.985</b>	<b>0.972</b>	128.4	89.91	<b>118.2</b>
Медиана	0.958	0.959	0.959	104.1	<b>83.64</b>	125.6
Смесь гауссовских распределений	0.959	0.958	0.958	<b>101.9</b>	89.48	131.8

Результаты сжатия видеопоследовательностей. Результаты для схем 1-3 в таблице 3 получены с помощью модели экспоненциального сглаживания

Таблица 3

Видеопоследовательность	Типовой (кб)	Схема 1 (кб)	Схема 2 (кб)	Схема 3 (кб)
Street traffic (1920x1080)	10550	1968	2903	3023
Hall (352x288)	548,7	128.4	89.91	118.2



При использовании моделей фона при сжатии видеопотока с камер видеонаблюдения можно улучшить результат сжатия от 3 до 5 раз по сравнению с типовой схемой (табл. 3). Наиболее эффективным способом моделирования фонового кадра является экспоненциальное сглаживание по критериям скорости обновления фоновых кадров и результатов сжатия. Выигрыш достигается для видеопоследовательностей, характерных для систем видеонаблюдения.

#### **Библиографический список**

1. *L. Yu, S. Chen, J. Wang*, Overview of AVS-video coding standards. *Signal Process. Image Commun.* 24, 247-262 (2009).
2. *Устюжанин Н.В.*, «Анализ моделирования переднего плана и фона в задаче детектирования движения», 68 МНСК ГУАП, 2015 год, сборник I часть, технические науки, с.324-с.329.
3. *Thomas Wiegand, Gary J. Sullivan*, Overview of the H.264/AVC Video Coding Standard, *IEEE Transactions on circuits and systems for video technology*, vol. 13, no. 7, july 2003.
4. *Сэлмон Д.*, «Сжатие данных, изображений и звука», М.: Техносфера, 2004.
5. Голованов Р.В., Воротнев Д.В., Методика сравнения алгоритмов сегментации движения, DSPA-2016, М., 2016, с.633-637

## СОДЕРЖАНИЕ

### ТЕХНИЧЕСКИЕ НАУКИ

Аэрокосмические приборы и системы .....	3
<i>Асенчик А. Ю.</i> Исследование влияния функции отказа на логистическую кривую.....	3
<i>Борисов П. О.</i> Исследование системы управления газотурбинным двигателем с селектором каналов.....	6
<i>Булавин С. А.</i> Сравнительный анализ радиотехнических систем посадки воздушных судов.....	11
<i>Булухова А. Н.</i> Методико-дидактические рекомендации для повышения эффективности работы с дистанционными адаптивными обучающими системами .....	15
<i>Виноградов В. В., Федоров И. Р.</i> Методы планирования задач в системах реального времени на многоядерных процессорах.....	19
<i>Виноградов А. Ю.</i> Сопоставление протоколов канального уровня для сети spasewire.....	24
<i>Волков В. А.</i> Реализация методов удаленного и автономного управления автомобилем на базе cots-платформы .....	28
<i>Гоголев К. В.</i> Методы прогнозирования загруженности городских причалов Санкт-Петербурга.....	32
<i>Горкалов С. С.</i> Проектирование системы управления оптической станции на основе трехмерной твердотельной модели и физико-математического моделирования.....	36
<i>Гулягина А. В.</i> Вопросы разработки имитационной модели морского пассажирского порта.....	40
<i>Деготинский Н. А.</i> Измерение глубины по дефокусировке изображения .....	45
<i>Дикан А. Ю., Коньшев Н. С.</i> Система определения центра масс самолета в полете.....	49
<i>Дуркин Р. П., Григорьев А. П.</i> Разработка учебного стенда для исследования технологий беспроводной передачи данных.....	54
<i>Евдокимов А. С.</i> Обзор алгоритмов поиска места подключения менеджеров plug-and-play для сети spasewire.....	58
<i>Иванов И. А.</i> Использование облачных технологий в автоматизированных системах обработки экспериментальных данных .....	62
<i>Иванова С. А.</i> Исследование алгоритмов и разработка программного обеспечения для фильтрации и классификации навигационных сигналов.....	64
<i>Колесников Д. Г., Григорьев А. П.</i> Актуальные вопросы применения дирижаблей в задачах пожаротушения и осаживания пепла при извержении вулканов .....	67
<i>Колокольцова М. С.</i> Бикритериальная нелинейная задача перевозки неоднородных грузов на разнородном транспорте.....	71
<i>Красавин А. М.</i> Сравнительный анализ детерминистических алгоритмов генерирования больших простых чисел для российского стандарта цифровой подписи ....	73
<i>Крохалева М. А.</i> Исследование оптимального адаптивного метода фильтрации сигналов .....	77

<i>Курбанов Л. И.</i> Применение подхода поиска р-медианы при размещении специализированных узлов в сети-на-кристалле .....	80
<i>Лебедев И. В.</i> Совместная оптимизация параметров фильтра нижних частот и оптимального фильтра параметров линейного движения квадрокоптера .....	85
<i>Лукашов Д. Ю.</i> Разработка методики подготовки тестов для контроля и диагностики аналого-цифровых параметров бортового оптического измерительного комплекса .....	88
<i>Мазиков М. С.</i> Использование ультразвуковых дальномеров на борту бпла .....	92
<i>Максимов А. Г.</i> Математическая модель разворота тяжелого самолета для имитации в тренажере .....	95
<i>Мелеховец Д. А.</i> Вопросы выбора рационального маршрута транспортировки негабаритных грузов железнодорожным транспортом .....	97
<i>Мелидин Е. А., Егоров В. С., Григорьев А. П.</i> Интеллектуальный электронный учебник для подготовки авиационных специалистов .....	100
<i>Миненко В. И.</i> Сравнительный анализ бортовых телекоммуникационных сетей .....	104
<i>Мухина Е. Н.</i> Исследование вычислителя бортового измерительно-оптического комплекса .....	108
<i>Пахарев С. М.</i> Методы формальной верификации приложений .....	113
<i>Понедельникова А. Н., Крузе А. Ю., Колесов А. М.</i> Анализ точности результатов расчета различными методами задач о прогибе пластинки .....	116
<i>Рыбальченко А. А.</i> Анализ распределенной системы прерываний на основе стандарта sрасewire для бортового вычислительного комплекса .....	124
<i>Рябов Д. А.</i> Автоматическая конфигурация вычислительных систем sрасewire plug-and-play .....	127
<i>Салий Н. Л.</i> Исследование метода комплексной нелинейной фильтрации сигналов с учетом надежности измерителей .....	129
<i>Тарасов Г. А.</i> Стема стабилизации бокового движения беспилотного летательного апта в условиях турбулентности .....	131
<i>Фальзан В. С.</i> Смарт – системы в туризме .....	134
<i>Филиппова Е. А.</i> Разработка алгоритма определения координат посредством совмещения фотоснимка сцены с цифровой картой места .....	140
<i>Хрусталев В. В.</i> Платформа для исследования различных микросхем памяти .....	144
<i>Чернелевский А. О.</i> Элементы дискретной математики при построении интеллектуальной навигационной тренажерно-обучающей системы .....	146
<i>Шакуров Д. М., Иванова Е. Р.</i> Виды и способы иерархического планирования задач в системах реального времени на многоядерных процессорах .....	150
<i>Ширяев С. А.</i> Модернизация бортового устройства регистрации полетной информации .....	157
<i>Шкаликов И. А.</i> Структура электронного магнитного компаса .....	161
Радиоэлектроника, электроника и связь .....	164
<i>Афанасьева К. Р.</i> Особенности технологического проектирования 3D mid-конструкций .....	164

<i>Баранова О. В.</i> PDM-система процесса производства замкнутой системы управления нейтрализацией выхлопных газов автомобиля.....	167
<i>Белолипецкая П. В.</i> Обеспечение параметров технологической среды при изготовлении электронной аппаратуры.....	171
<i>Бондарец Е. А.</i> Модальный анализ.....	176
<i>Бондарчук С. А.</i> Исследование пластины на случайную вибрацию .....	180
<i>Борискин А. А.</i> Математическая модель однородной прямоугольной платы, свободно опертой по краям, с внешним демпфирующим слоем .....	184
<i>Дегтярь И. В.</i> Стационарный тепловой анализ пластины.....	187
<i>Жданов Д. А.</i> Обзор возможностей современных пакетов моделирования тепловых режимов .....	189
<i>Желудева К. В.</i> Факторы, определяющие выбор аккумуляторных батарей для бортовых систем электроснабжения .....	191
<i>Канатова А. А.</i> Возможность использования электродвигателя типа ДПР в механизме подачи порошковой проволоки .....	193
<i>Корчигин Н. В.</i> Анализ задачи электромагнитной совместимости приемно-передающей системы.....	196
<i>Кривалова А. И.</i> Направления развития технологий плазменной очистки и травления.....	198
<i>Кузьменко В. П.</i> Анализ причин отказов электронных узлов .....	202
<i>Леппик А. В., Булякулов Р. Р., Поляков Д. Д., Суворов А. П.</i> Использование обратного синтезирования апертуры антенны для достижения высокой разрешающей способности по углу .....	205
<i>Орлов А. А.</i> Корреляционный анализ в радио и оптическом диапазоне .....	211
<i>Осколков Б. В.</i> Фотовольтаические эффекты как физическая основа для создания нового поколения датчиков.....	214
<i>Параскун А. С.</i> Процесс фотодетектирования и оценка энергетического спектра оптического излучения многоканальным резонансным спектральным прибором .....	217
<i>Рудых А. А.</i> Исследование зависимости качества паяных соединений электронных модулей от факторов, определяемых паяльной пастой .....	220
<i>Федоров С. Н.</i> Современное измерительное устройство – наномеханический резонатор.....	224
<i>Хабаров Д. Л.</i> Анализ технологий изготовления чувствительных элементов на основе микромеханических систем.....	228
<i>Черноусова П. М.</i> Факторы, определяющие надежность паяных соединений при поверхностном монтаже .....	232
<i>Чугунов А. А.</i> Simulink-модель бистабильной потенциальной ямы в курсе квантовых вычислений .....	236
<b>Вычислительные системы и программирование.....</b>	<b>240</b>
<i>Александров О. Д.</i> Программное средство поиска антипаттернов в исходном коде на языке java .....	240
<i>Баранов Я. В.</i> Сбор, обработка и хранение личной статистики. Проект TRACKALLMYLIFE .....	242

<i>Беляевский А. О.</i> Управление несколькими аналоговыми сервоприводами с помощью одного таймера на примере отладочной платы stm32f429i-disco .....	244
<i>Бобрышов Д. А.</i> Разработка трехмерных веб-приложений на языке WebGL.....	247
<i>Боженко В. В., Игошин А. Н.</i> Сравнение методов одномерного поиска экстремума .....	250
<i>Виноградов А. Ф., Клеченов А. С.</i> Комбинированные усилители звуковой частоты .....	252
<i>Гнутова Ю. Д.</i> Проектирование и разработка автоматизированной информационной системы создания рабочих программ дисциплин на кафедре .....	256
<i>Горелик Д. В., Чурбанов А. А.</i> Создание дистанционной системы обучения на платформе moodle .....	258
<i>Григорьев Е. К., Журавлев А. С., Юдин И. А.</i> Способ определения размера зоны поиска радиолокационными средствами для решения задач по спасению людей структурами МЧС в чрезвычайных ситуациях .....	262
<i>Гусманова М. Р.</i> Компьютерный анализ обусловленности матриц фробениусова типа .....	266
<i>Даванс А. И.</i> Исследование характеристик пульсометра типа Polar Team2 Pro .....	271
<i>Даванс Д.</i> Исследование колебаний неподвижного человека с использованием пакета LabVIEW .....	276
<i>Давыдова О. В.</i> Организация электронного портфолио студента .....	279
<i>Дьяконов Д. И.</i> Трехмерный пользовательский интерфейс в стратегической игре .....	281
<i>Еремченко А. Ю.</i> Создание аналитической модели для решения задач анализа данных .....	283
<i>Золотухин С. С.</i> Выбор алгоритма поверхностного кодирования и коррекции ошибок при проектировании программно-определяемой системы хранения данных.....	286
<i>Иванова М. С., Липецкий П. А.</i> Анализ методов распределения ресурса канала связи .....	290
<i>Костин В. В.</i> Стратегическая игра. Мультиюзер .....	294
<i>Кузнецова Ю. А.</i> Тренажер для ввода смеси разноязыковых текстов.....	296
<i>Легкий Н. С., Степанов П. А.</i> Разработка комплекса лабораторных работ по курсу “основы микропроцессорной техники” на основе оценочной платы STM32F4DISCOVERY .....	298
<i>Логвинчук Е. А.</i> Расширение функциональности olar-сервера для повышения производительности от его использования .....	301
<i>Манкевич М. В.</i> Применение мультиплексированного асинхронного сетевого взаимодействия в интерактивных обучающих системах .....	304
<i>Масимчук О. О.</i> Сравнительный анализ систем дистанционного обучения (СДО) .....	308
<i>Миронов А. Ю.</i> Автоматизированная система управления производством по делам об административных правонарушениях на региональном уровне.....	314
<i>Нарышкин Д. Е.</i> Унификация адресов многоквартирных домов в Санкт-Петербурге с использованием ФИАС .....	319

<i>Пазенко Д. О.</i> Реконструкция трехмерной сцены на основе видеопотока .....	322
<i>Печенин Д. В.</i> Компьютерное моделирование одного класса задач терминального управления .....	324
<i>Пивоваров А. А.</i> Контейнеризация и виртуализация, перспективы, сравнительный анализ.....	328
<i>Пожарова К. Д.</i> Сбор, сохранение и анализ данных о тренировках и питании человека .....	332
<i>Полах О. В.</i> Формальные средства спецификации .....	335
<i>Розов А. А.</i> Сравнение картографических сервисов для реализации геоинформационной системы гидробиологического мониторинга (предложен анализ существующих сервисов на основе требований к геоинформационной системе – гис).....	338
<i>Свиридова А. А.</i> Математические основы геоэлектрических методов обнаружения подземных источников воды .....	341
<i>Сергеев А. В.</i> Средство визуализации состояния сети массового обслуживания .....	345
<i>Синяев А. И.</i> Возвратно-ориентированное программирование и обход рандомизации адресного пространства .....	348
<i>Смолиенко Д. А.</i> Организация защиты от нсд электронного документооборота.....	351
<i>Сысоева А. С.</i> Приенение методов и алгоритмов компьютерной графики для обработки и визуализации данных дистанционного зондирования земли .....	353
<i>Сытин А. Р.</i> Анализ критериев качества работы нейронной сети в задачах классификации .....	356
<i>Топчий А. С.</i> Возможности программной интерпретации и генерации индивидуальных особенностей рукописного шрифта.....	360
<i>Фролова А. Ю., Шлюев Е. С.</i> Особенности механизма блокировок в сетевой версии access .....	362
<i>Цветков И. А.</i> Модели представления знаний, алгоритмы манипуляции знаниями и их спользование в интеллектуальных системах поддержки принятия решений.....	365
<i>Шведчиков В. А.</i> Маршрутизация транспортного потока с ограничением по грузоподъемности .....	370
<i>Шульга И. В.</i> Заочное голосование собственников помещений в многоквартирном доме в форме электронного голосования .....	374
Информационные системы и защита информации.....	377
<i>Воропаева Е. Н.</i> Использование системы моделирования для проверки знаний .....	377
<i>Голос Н. А.</i> Анализ безопасности протокола https .....	380
<i>Клюев М. А., Минаева Т. А.</i> Стеганоанализ статических изображений .....	383
<i>Козыревский В. К.</i> Способ уменьшения сложности обучения алгоритма виолы–джонса.....	387
<i>Миронова И. А.</i> Анализ протокола передачи FCoE.....	391
<i>Нестерук С. В.</i> Обзор подходов лингвистической стеганографии .....	393
<i>Прощенко С. Ю.</i> Анализ алгоритма шифрования RSA.....	396

<i>Татарникова А. А.</i> Разработка алгоритма обнаружения патологий опорно-двигательного аппарата с помощью акселерометров в мобильных устройствах .....	400
<i>Ткачев С. С.</i> Обзор и анализ уязвимостей RFID-систем.....	402
<i>Толопило Д. В.</i> Spdy: экспериментальный протокол от GOOGLE.....	406
<i>Устюжанин Н. В.</i> Анализ методов моделирования фона для системы сжатия видеопоследовательностей .....	408

Научное издание

ШЕСТИДЕСЯТ ДЕВЯТАЯ  
МЕЖДУНАРОДНАЯ СТУДЕНЧЕСКАЯ  
НАУЧНАЯ КОНФЕРЕНЦИЯ ГУАП

Часть I  
Технические науки

Сборник докладов

Ответственный за выпуск:  
А. В. Бобович

Материалы публикуются в авторской редакции  
Подготовка к печати *М. И. Дударева*

---

Подписано к печати 06.10.16. Формат 60x84 1/16. Бумага офсетная.  
Печать офсетная. Усл. печ. л. 24,41. Уч.-изд. л. 26,25. Тираж 150 экз.  
Заказ № 348.

---

Редакционно-издательский центр ГУАП  
190000, Санкт-Петербург, ул. Б. Морская, д. 67