

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ**



ИНФОРМАТИКА И КИБЕРНЕТИКА

3(5)

Донецк – 2016

УДК 004.3+004.9+004.2+51.7+519.6+519.7

**ИНФОРМАТИКА И КИБЕРНЕТИКА, № 3(5), – Донецк: ДонНТУ,
2016.**

Представлены материалы по вопросам приоритетных направлений научно-технического обеспечения в области информатики, кибернетики и вычислительной техники.

Материалы предназначены для специалистов народного хозяйства, ученых, преподавателей, аспирантов и студентов высших учебных заведений.

Редакционная коллегия

Главный редактор: Павлыш В.Н., д.т.н., проф.

Зам. глав. ред.: Андрюхин А.И., к.т.н., с.н.с.

Ответственный секретарь: Завадская Т.В., к.т.н., доц.

Члены редакционной коллегии: Аверин Г.В., д.т.н., проф. Аноприенко А.Я., к.т.н., проф.

Григорьев А.В., к.т.н., доц., Грунский И.С., к.т.н., проф., Зинченко Ю.Е., к.т.н., доц.

Звягинцева А.В., к.т.н., доц., Миненко А.С., д.т.н., проф., Скобцов Ю.А., д.т.н., проф.,

Шелепов В.Ю., д.т.н., проф., Карабчевский В.В., к.т.н., доц. Федяев О.И., к.т.н., доц.

Рекомендовано к печати ученым советом ГВУЗ «Донецкий национальный технический университет» Министерства образования и науки ДНР. Протокол № 9 от 23.12.2016.

Журнал зарегистрирован в Министерстве образования и науки ДНР № 307 от 06.08.2015

Контактный адрес редакции

ДонНТУ, ул. Артема, 58, Донецк, 83001

Тел.: +380 (62) 301-08-56. Эл.почта: infcyb.donntu@yandex.ru

Интернет: <http://infcyb.donntu.org>

© Донецкий национальный технический университет
Министерство образования и науки ДНР, 2016

СОДЕРЖАНИЕ

Интервальные оценки пороговых параметров хаотических чипов Андрюхин А.И.....	4
Разработка скоринговой модели с использованием методов логистической регрессии и ROC – анализа Анохина И.Ю.....	13
Система формирования трафика компьютерной сети на основе самоорганизованной критичности Д.В. Бельков, Е.Н. Едемская.....	29
Статистический анализ трафика беспроводной сети Бельков Д.В., Едемская Е.Н.....	26
Согласованная оптимизация в задачах управления эволюцией наукоемкого предприятия Г.П. Виноградов, В.Н. Кузнецов.....	35
Формирование баз знаний производственного типа на основе UML-моделей Н.О. Дородных, Ю.Юрин.....	44
Математический анализ вероятности возникновения всерных отключений электросетей Кузнецов П.А.,Юдин А.В., Масло С.В., Клюковкин В.Р.....	51
Разработка имитационной модели для анализа динамики движения материалов и сырья с учетом случайных факторов Д.И. Левченко, В.А. Светличная, К.К. Бабич.....	57
Моделирование внутренних операций процессорных элементов Р.В. Мальчева, Т.В. Завадская.....	65
К вопросу робастности средств JPWL в условиях пакетных ошибок С. В. Скороход, О.А .Касьянов.....	72
Численное моделирование и управление процессами теплопереноса в противоточных теплообменных аппаратах Толстых В. К, Пшеничный К А.....	80
Проектирование и разработка компилятора C-подобного языка программирования общего назначения с поддержкой исполнения кода на этапе компиляции Н.М. Ткачѳв, А.И. Андрюхин.....	86
Компьютерные расчеты и визуализации характеристик устойчивости систем: проблема центра-фокуса и гармонические функции Филер З.Е., Андрюхин А.И.....	92
Методы разработки систем учета научно-технических мероприятий. Ходич Д.В., Завадская Т. В., Грищенко В.И.....	98

УДК 004.0- 519.854.001

Интервальные оценки пороговых параметров хаотических чипов

А.И. Андрухин

Донецкий национальный технический университет, г. Донецк,
alexandruckin@ramber.ru

Андрухин А.И. Интервальные оценки пороговых параметров хаотических чипов. Эта работа относится к глобальной проблеме построения компьютеров нового типа, которые способны самоорганизовываться и адаптироваться во внешней среде. В работе рассматриваются теоретические обоснования для создания всех основных логических элементов на основе одной хаотической нелинейной системы. Решаемая в статье задача состоит в построении интервальных оценок пороговых параметров использования и управления хаотической временной структуры, генерирующей нелинейные временные ряды с целью получить гибкие и реконфигурируемые вычислительные устройства. Приведены результаты компьютерных расчетов. В расчетах использовался пакет Mathematica.

Ключевые слова: хаотический чип, интервальная оценка, самоорганизация

Введение

Современные компьютеры представляют собой детерминированные системы с очень большим, но конечным числом состояний. Живой организм для существования в изменчивой среде должен обладать элементами хаотического поведения. Искусственные системы, способные эффективно взаимодействовать с меняющейся внешней средой, должны быть в какой-то мере хаотичными.

Создание компьютеров нового типа - открытых систем с точки зрения термодинамики, способных самоорганизоваться и адаптироваться во внешней среде - это задача уже сегодняшнего дня (госпрограмма США BRAIN-2013).

Сейчас на основе динамического хаоса создают аппаратно стохастические чипы. Подчеркнем, что существует большая область построения новых типов транзисторов с использованием современных физических направлений.

Основными из этих направлений являются:

1) Транзисторы управляемые одним электроном Single-electron transistors SET.

2) Квантовые клеточные автоматы (QCA-quantum cellular automata) базирующиеся на основе квантовой точки (Quantum dot).

3) Туннелированные диоды (RTD-) Resonant tunneling diode

4) Устройства на быстрой одноквантовой логике (Rapid single-flux quantum (RSFQ) device)

5) Транзистор на основе карбоновых трубок (Carbon nanotube FET)

6) ДНК-вычисления (DNA computing)

Некоторые недостатки и их перспективы развития представлены в [1]

Известным направлением является реализация всех основных логических элементов на основе одной нелинейной хаотической системы [2-5]. Известно, что краеугольным камнем современной компьютерной архитектуры является бинарная цифровая логика, логика истинного и ложного. Булева логика отличается своей концептуальной простотой.

Строго показано, что любая логическая операция может быть реализована путем соответствующих соединений адекватного подключения NOR и/или NAND вентилей, т.е. построение универсальной вычислительной техники связано с реализацией фундаментальных NOR и NAND вентилей.

В работе рассматриваются теоретические обоснования для получения всех основных логических элементов на основе одной нелинейной системы.

Решаемая в статье задача состоит в построении интервальных оценок пороговых параметров использования и управления хаотической временной структуры, генерирующей нелинейные временные ряды с целью получить гибкие и реконфигурируемые вычислительные устройства.

Задача создания модели хаотического процессора

Рассмотрим хаотическую чип или процессор, основой которого может быть одномерная система, состояние которой представлена переменной x , и динамика которой задается нелинейным отображением $F(X)$.

Постановка задачи предполагает:

· выбор типов аттракторов, пригодных для работы с информацией;

выбор динамических явлений для реализации базовых операций обработки информации с использованием хаоса;

разработку принципов, позволяющих ставить во взаимно-однозначное соответствие информацию и траектории динамической системы;

разработку конкретных математических моделей, позволяющих работать с информацией как с траекториями отображений и

управлять динамическими явлениями для реализации базовых операций хаотического процессора;

создание программных комплексов для симуляции хаотических процессоров на компьютерах;

исследование модели хаотического процессора;

решение с помощью хаотического процессора сложных задач, плохо решаемых при традиционных подходах

Решение задачи

Отображение функционирования всех 2-входовых вентилях NAND, NOR, XOR, AND, OR и XNOR включает в себя следующие шаги:

Логические входы I1 и I2 для логических операций 2-входовых вентилях кодируют начальное состояние системы x_0 следующим образом:

1) $X_0 \rightarrow x_{gate} + X_1 + X_2$, где физическая величина X_1, X_2 , которой обычно соответствует напряжение, имеет значение 0, когда логические значения входов I1, I2 равно 0, и имеет значение L_{in} когда логические значения входов I1, I2 равны 1, где L_{in} – ранее оговоренная положительная константа (нижний порог). Аналогично X_1 и X_2 , x_{gate} является пороговым напряжением рассматриваемого вентиля, которое мы можем изменять.

2. Динамическая эволюция в n временных шагах (n -я итерация начального состояния), в результате чего в обновленном состоянии $X \rightarrow F_n(X_0)$, Конкретно мы возьмем $n = 1$ здесь, то есть $X = F_1(X_0)$.

3. Определение выхода: логическое значение выхода равен 0, если $F(X_0) < N_{out}$ и значение выхода равен 1, если $F(X_0) > N_{out}$ где N_{out} - ранее оговоренная положительная константа (верхний порог).

Так как F нелинейна, чтобы точно задать начальное X_0 , нужен механизм управления порогом для установки начального X_0 , т.е. определить L_{in} , N_{out} , x_{gate} для каждого вентиля (желательно для упрощения управления L_{in} определять единым).

Для того, чтобы получить все желаемые значения откликов различных вентилях, мы должны с помощью порогового механизма

управления удовлетворять условиям, перечисленным в таблице 1 одновременно.

Таблица 1. Логические значения выходов двухвходовых вентилях

I ₁	I ₂	And	Nand	Or	Nor	Xor	NXor
0	0	0	1	0	1	0	1
0	1	0	1	1	0	1	0
1	0	0	1	1	0	1	0
1	1	1	0	1	0	0	1

Основные результаты представлены в табл.2 во 2 и 3 строчках, показывающие точные решения для N_{out} , x_{gate} при $L_{in}=1/4$, которые удовлетворяют условиям в таблице 1 и динамическая эволюция определяется логистическим уравнением $F(x) = gx(1-x)$.

На рис.1 представлены графики отображения $F_n(X)$, где $F_n(X)=F(F_{n-1}(X))$ и здесь параметр нелинейности $g = 4$ и порог $L_{in} = 1/4$, общий для всех логических вентилях.

Возможно использование иных отображений, к примеру $F(X) = 6.5x^2(1-x)$, $x_{n+1}=2x_n/(1+x_n^{10})$ [5].

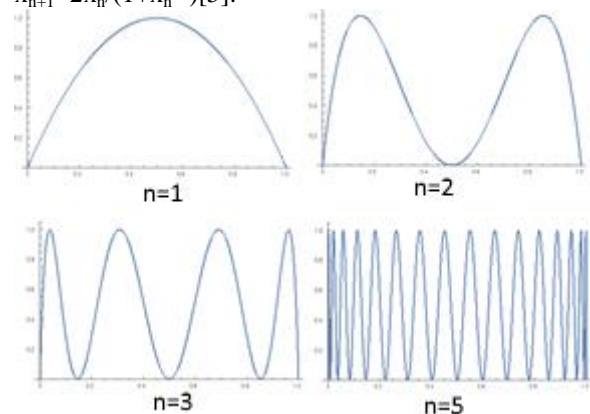


Рисунок 2. – Графики отображения $F_n(X)$.

Свойства отображения $F(X) = gx(1-x)$.

При интерпретации отображения $F(X) = gx(1-x)$ для эволюции популяции (Ферхольст) имеем, что при изменении значения параметра

1). $0 < g < 1$, популяция вымрет, независимо от начальных условий.

2). $1 < g < 2$, численность популяции быстро выйдет на стационарное значение $(g-1)/g$, независимо от начальных условий.

3). $2 < g < 3$, численность популяции придёт к стационарному значению $(g-1)/g$, но вначале будет колебаться вокруг него. Скорость сходимости линейна везде, кроме значения $g=3$, при котором она крайне мала.

4). $3 < g < 1 + \sqrt{6} \approx 3.45$, численность популяции будет бесконечно колебаться между 2 значениями.

5). $3.45 < g < \approx 3.54$, численность популяции будет бесконечно колебаться между 4 значениями.

б). $r > 3.54$, численность популяции будет колебаться между 8 значениями, потом 16, 32 и т.д.

Длина интервала изменения параметра, при котором наблюдаются колебания между одинаковым количеством значений, уменьшается по мере увеличения r .

Отношение между двумя длинами смежных интервалов стремится к числу Фейгенбаума $\delta \approx 4.669...$

Эта ситуация является типичным примером каскада бифуркаций удвоения периода, который представлен на рис..

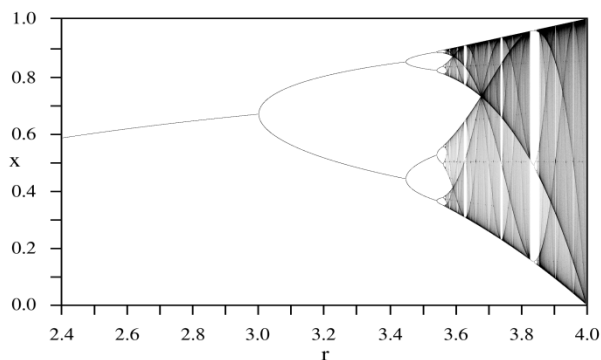


Рисунок 2. – Бифуркационная диаграмма.

При $r \approx 3.57$, начинается хаотическое поведение, а каскад удвоений заканчивается.

Колебания больше не наблюдаются.

Небольшие изменения в начальных условиях приводят к большим отличиям дальнейшего поведения системы во времени.

Это является главным признаком хаотического поведения.

При $r > 3.57$ обычно имеем хаотическое поведение, однако существуют небольшие изолированные интервалы значений r , при которых система ведет себя регулярно,

Их называют «окнами периодичности».

К примеру, начиная со значения $1 + \sqrt{8} \approx 3.83$, существует интервал параметров r , при котором наблюдаются колебания между 3 значениями.

Для больших значений r - между 6 значениями, потом 12 и т. д.

В системе можно найти периодические колебания с любым количеством значений.

Последовательность смены количества значений удовлетворяет порядку Шарковского.

При $r > 4$, значения отображения покидают интервал $[0,1]$ и расходятся при любых начальных условиях.

Основные понятия при построении хаотических компьютеров

В табл.2 во 2 и 3 строчках представлены нижний и верхний пороги для соответствующих элементов(операций). Итеративное применение хаотического логического отображения позволяет использовать эти логические элементы с иной логической функцией, т.е. выполнять их реконфигурацию без дополнительного аппаратного оборудования. Подчеркнем, что начальный нижний порог равен 0.316.

Таблица 2. Значения пороговых параметров двухвходовых вентилей

Операция	Nand	And	Nor	Xor	Or
$L_{op}(x_{gate})$	3/8	0	*	1/4	1/8
$H_{op}(H_{out})$	11/16	3/4	*	3/4	11/16
Итерация	1	2	3	4	5
Состояние чипа	X_1	X_2	X_3	X_4	X_5
Входы (0,0) $X_0=0.316$	0.864	0.468	0.996	0.016	0.063
Логич. значение	1	0	1	0	0
Входы (1,0)/(0,1) $X_0=0.566$	0.982	0.068	0.255	0.760	0.729
Логич. значение	1	0	0	1	1
Входы (1,1) $X_0=0.816$	0.601	0.959	0.155	0.525	0.997
Логич. значение	0	1	0	0	1

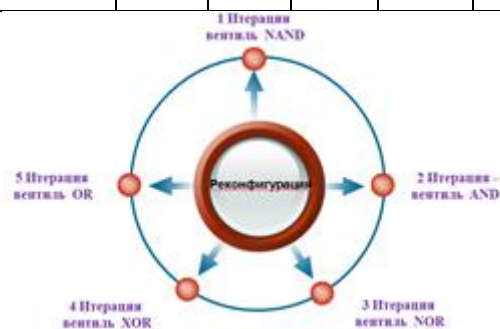


Рисунок 3а. –Итеративная реконфигурация хаотического элемента

Иной подход представлен на рис.3б, где реконфигурация выполняется с помощью динамического логического порога C так, что при одном значении C элемент выполняет функцию

Nand, а при другом значению функцию Nor и т.п.[5].

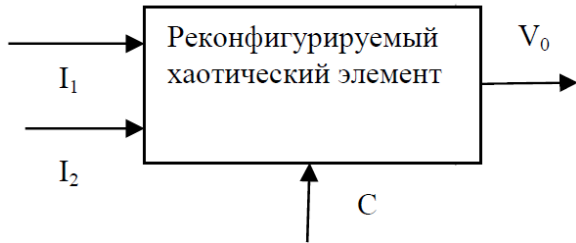


Рисунок 3б. –Реконфигурация хаотического элемента сигналом С.

В работе построены интервальные оценки пороговых параметров L_{op} , H_{op} и L_{in} для всех двухвходовых вентилях и на рис.5-22 представлены результаты для вентилях NAND и AND, [6,7].

Элемент NAND

Функционирование элемента должно удовлетворять следующим условиям

- 1) $F(L_{nand}) > H_{op}$,
 - 2) $F(L_{nand} + L_{in}) > H_{op}$,
 - 3) $F(L_{nand} + 2L_{in}) < H_{op}$.
 - 4) $H_{op} > L_{nand} + 2L_{in}$
- Здесь $L_{nand} = L_{op} = x_{gate}$ и $H_{nand} = H_{op}$.

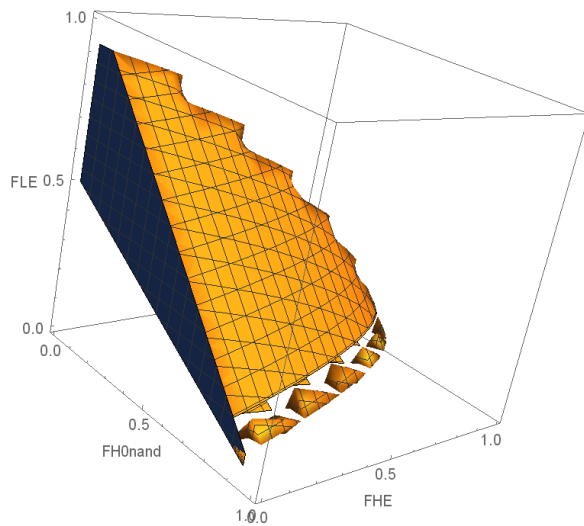


Рисунок 4. – 3-D область возможных решений для элемента NAND на 1 итерации.

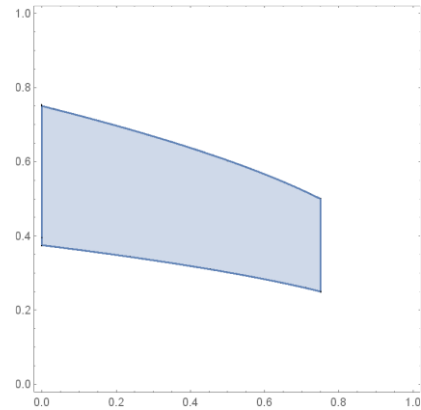


Рисунок 5. – 2-D область возможных решений для элемента NAND при $L_{in}=0.25$.

Элемент AND

Функционирование элемента должно удовлетворять следующим условиям

- 1) $F(L_{and}) < H_{op}$,
- 2) $F(L_{and} + L_{in}) < H_{op}$,
- 3) $F(L_{and} + 2L_{in}) > H_{op}$.
- 4) $H_{op} > L_{and} + 2L_{in}$

Аналогично предыдущему случаю, $L_{and} = L_{op} = x_{gate}$ и $H_{and} = H_{op}$.

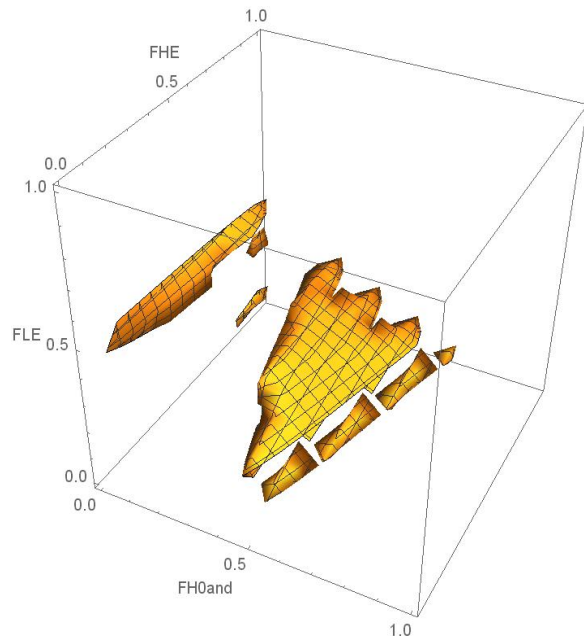


Рисунок 6. – 3-D область возможных решений для элемента AND на 2 итерации.

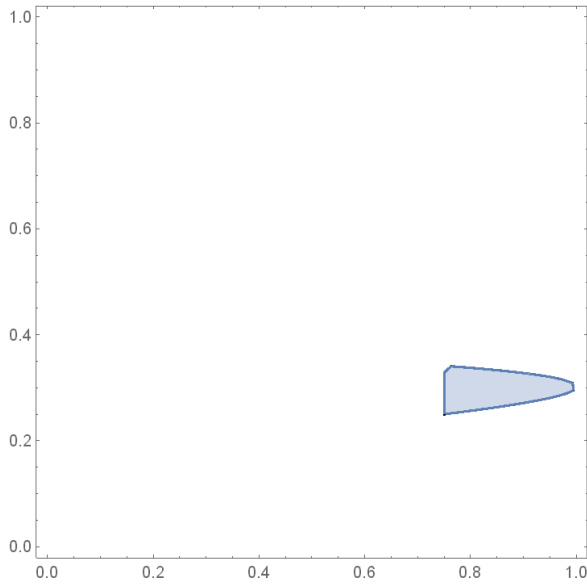


Рисунок 7. – 2-D область возможных решений для элемента AND при $L_{in}=0.25$ на 2 итерации.

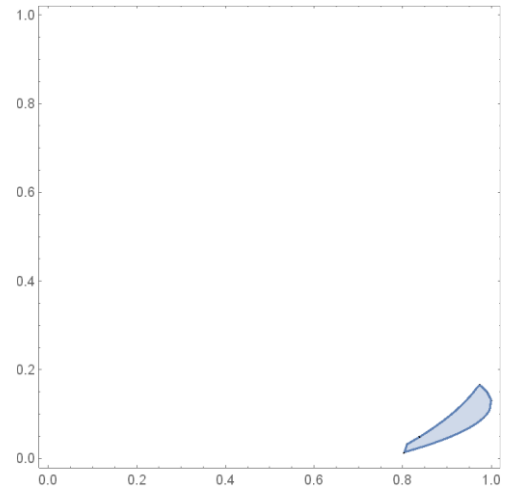


Рисунок 9. – 2-D область возможных решений для элемента AND при $L_{in}=0.25$ на 1 итерации.

Элемент NOR

Функционирование элемента должно удовлетворять следующим условиям

- 1) $F(Land) > Hop$,
- 2) $F(Land + Lin) < Hop$,
- 3) $F(Land + 2Lin) < Hop$.
- 4) $Hop > Land + 2Lin$

Аналогично предыдущему случаю, $Land=Lop=xgate$ и $Hand= Hop$.

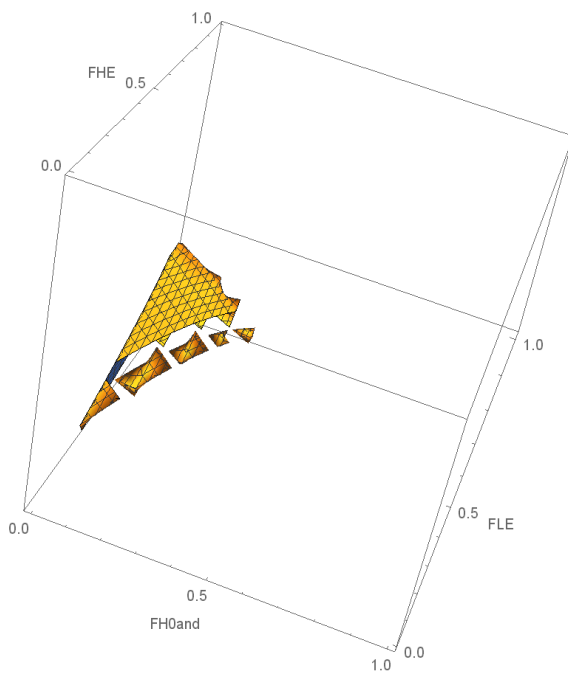


Рисунок 8. – 3-D область возможных решений для элемента AND на 1 итерации.

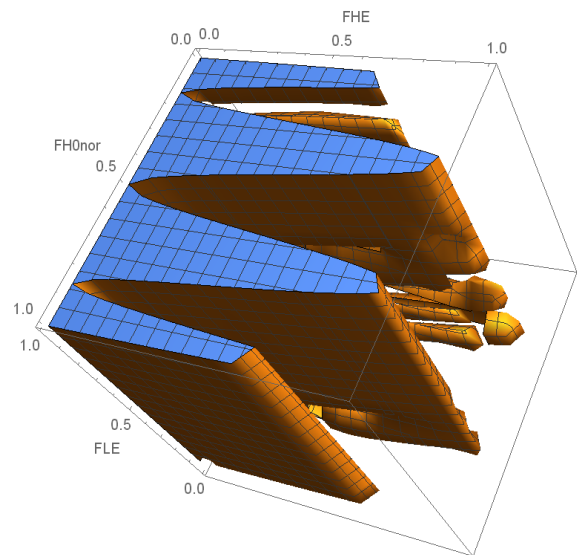


Рисунок 10. – 3-D область возможных решений для элемента NOR на 3 итерации.

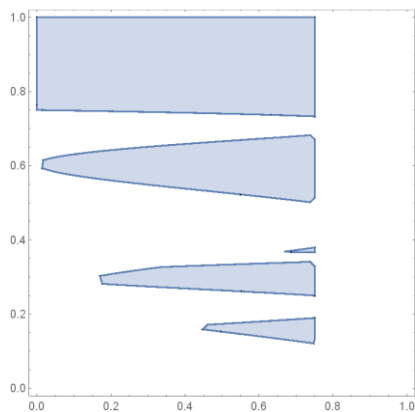


Рисунок 11. – 2-D область возможных решений для элемента NOR при $L_{in}=0.25$ на 3 итерации.

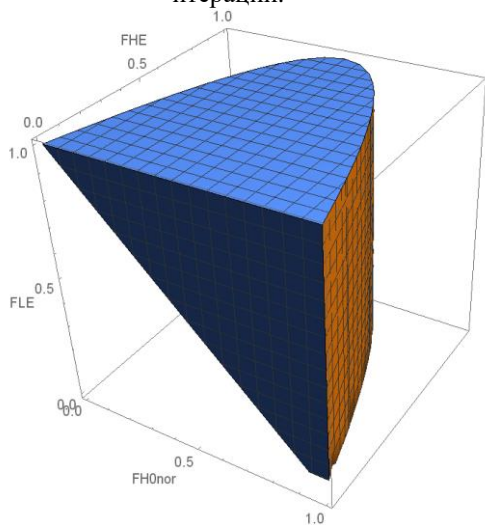


Рисунок 12. – 3-D область возможных решений для элемента NOR на 1 итерации.

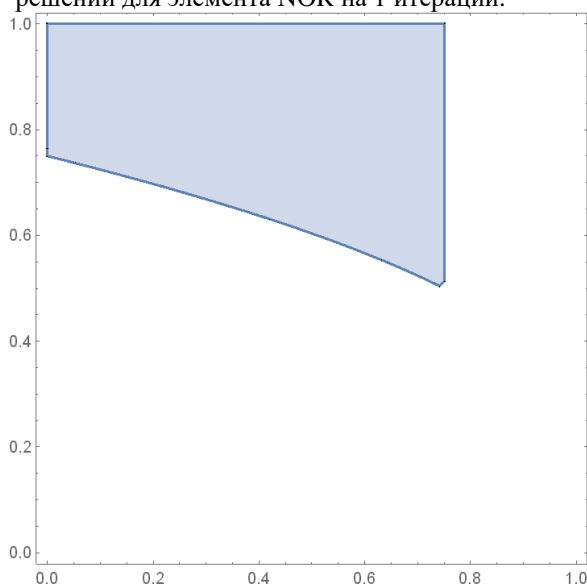


Рисунок 13. – 2-D область возможных решений для элемента NOR при $L_{in}=0.25$ на 1 итерации.

Функционирование элемента должно удовлетворять следующим условиям

- 1) $F(Land) < Hop$,
- 2) $F(Land + Lin) > Hop$,
- 3) $F(Land + 2Lin) < Hop$.
- 4) $Hop > Land + 2Lin$

Аналогично предыдущему случаю, $Land=Lop=xgate$ и $Hand=Hop$.

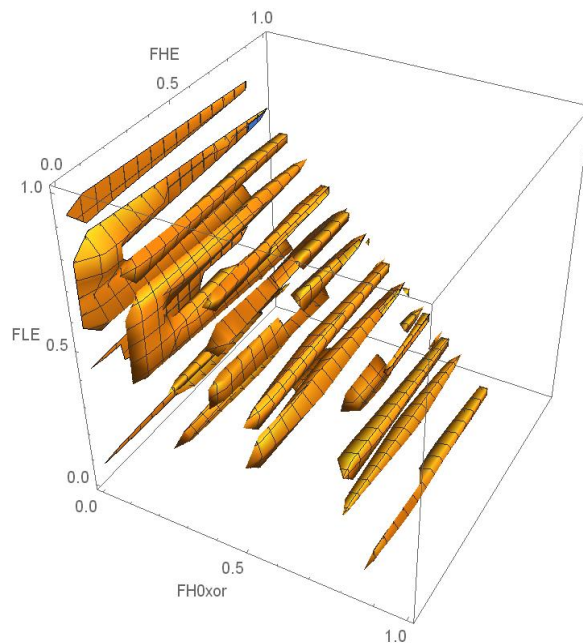


Рисунок 14. – 3-D область возможных решений для элемента XOR на 4 итерации.

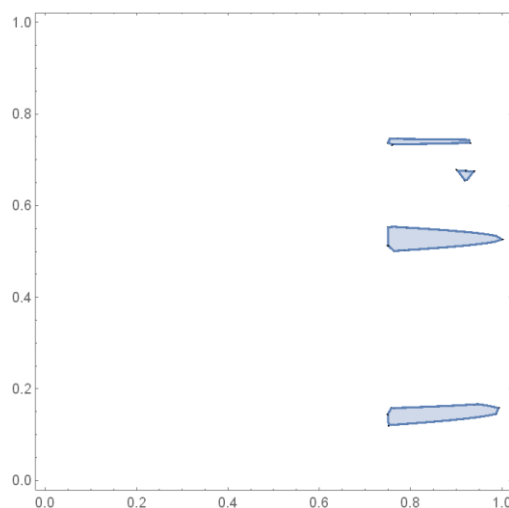


Рисунок 15. – 2-D область возможных решений для элемента XOR при $L_{in}=0.25$ на 4 итерации.

Элемент XOR

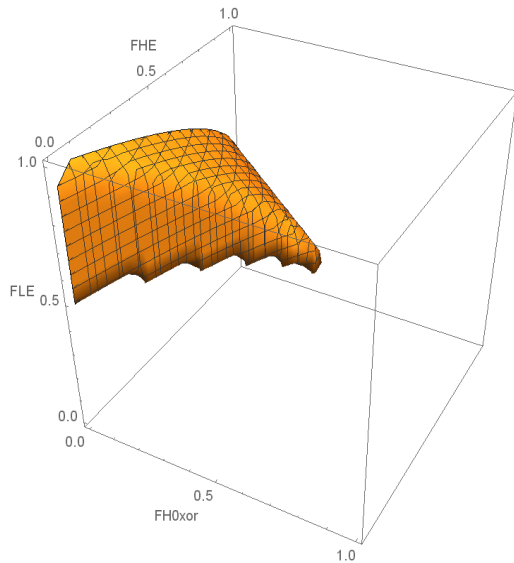


Рисунок 16. – 3-D область возможных решений для элемента XOR на 1 итерации.

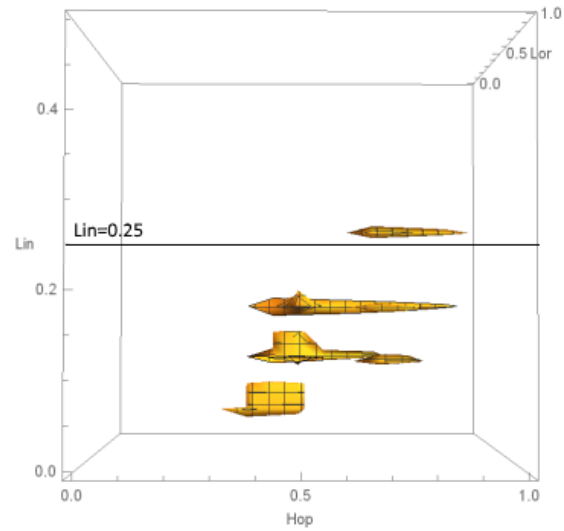


Рисунок 19. – 3-D область возможных решений для элемента OR на 5 итерации.

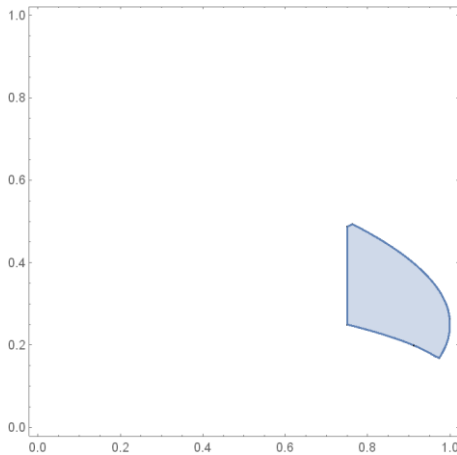


Рисунок 17. – 2-D область возможных решений для элемента XOR при $L_{in}=0.25$ на 1 итерации

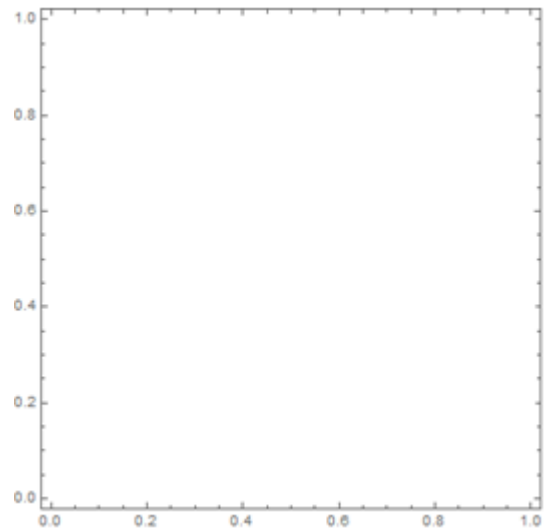


Рисунок 20. –Пустая 2-D область возможных решений для элемента OR при $L_{in}=0.25$ на 5 итерации

Интервальные оценки порогов элемент OR и их применение

Функционирование элемента должно удовлетворять следующим условиям

- 1) $F(Land) > Hop$,
- 2) $F(Land + Lin) < Hop$,
- 3) $F(Land + 2Lin) > Hop$.
- 4) $Hop > Land + 2Lin$

Аналогично предыдущему случаю, $Land=Lop=xgate$ и $Hand= Hop$.

Согласно рис.19 мы можем с некоторой долей уверенности подозревать, что нет решений при $L_{in}=0.25$, что подтверждает рис. 20.

Существуют решения для пороговой системы при $L_{in}=0.2$, что представлено на рис.21.

Этот пример показывает, что простая визуализация соотношения пороговых величин в трехмерном пространстве позволяет определять возможные решения, которые естественно мы более направлены исследовать.

Необходимо отметить, что кажущийся самым простым из базовых вентилях элемент OR с точки зрения определения оценок порогов является самым капризным и даже на 1 итерации область возможных решений невелика (см.рис.22)..

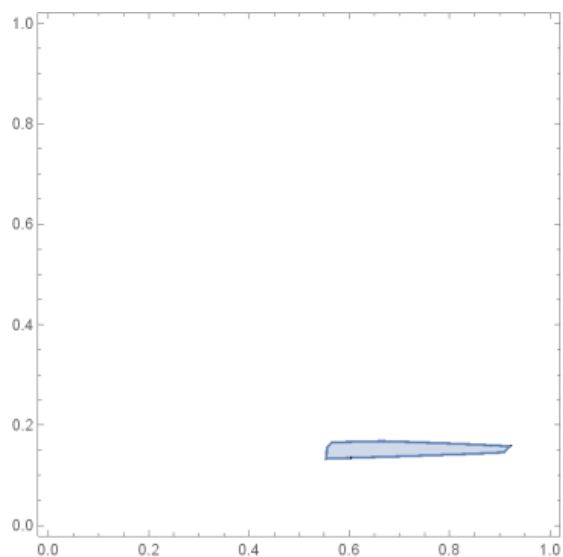


Рисунок 21. –Непустая 2-D область возможных решений для элемента OR при $L_{in}=0.2$ на 5 итерации.

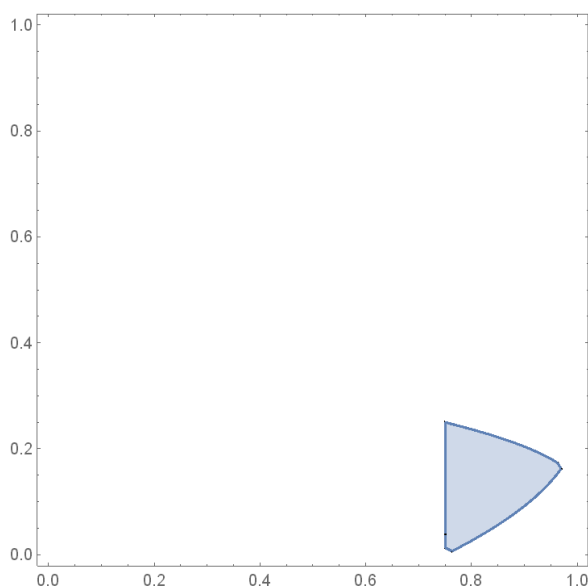


Рисунок 22. – 2-D область возможных решений для элемента OR при $L_{in}=0.25$ на 1 итерации.

Аппаратная реализация порогового механизма

Общая схема аппаратной реализации порогового механизма представлена на рис.22, где V соответствует пороговому значению.

Литература

- 1.Wong H.-S. P. Beyond the conventional transistor/ IBM Journal RES. & DEV. Vol. 46 no. 2/3 March/May 2002 .pp.133-168.
2. William L. Ditto,1 A. Miliotis,2 K. Murali,3 Sudeshna Sinha,4, Mark L. Spano. Chaogates:

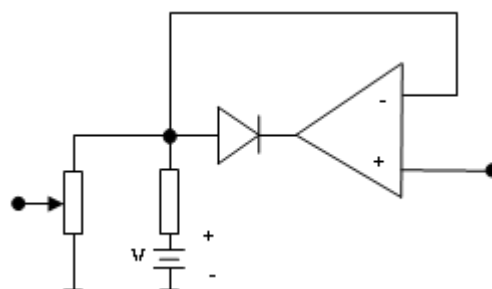


Рисунок 22- Аппаратная реализация порогового механизма

Выводы

Недавно фирмой ChaoLogix Inc. разработан и изготовлен чип, который демонстрирует возможность построения реконфигурируемых хаотических логических вентилей(стандартная CMOS-технология, 0,18 μ , процесс TSMC, частота 30 МГц, размер матрицы $3,1 \times 3.1$ mm, напряжение 1,8 в).

Основной блок ChaoGate схематически показано на рис. 5.

ChaoGates включены в массив ChaoGate в чипе СБИС, чтобы продемонстрировать их использование и более высокую функциональность за время, менее чем за один такт в:

А)в небольшом АЛУ, (множитель и сумматор).

Б) в протоколах связи (CP), в частности в последовательном периферийном интерфейсе (SPI, синхронный последовательный канал передачи данных) и реализациям шина Inter Integrated Circuit Control (I2C).

В настоящее время усилия сосредоточены на оптимизации конструкции одного ChaoGate до уровней, где они сопоставимы с одним вентилем с точки зрения мощности и размеров и способны менять свое поведение в границах одного тактового цикла.

Предварительные разработки показывают, что эта цель достижима, т.е все вентили в используемые для разработки компьютеров могут быть заменены ChaoGates для обеспечения дополнительной гибкости и производительности.

Morphing logic gates that exploit dynamical patterns // CHAOS 20, 037107 (2010).

3.William L. Ditto, K. Murali and Sudeshna Sinha. Construction of a Chaotic Computer Chip.

4.Ю.В.Андреев,А.С. Дмитриев, Д.А.Куминов Хаотические процессоры//Успехи современной

Радиоэлектроники, №10,1997

5.С.П. Шарый. Конечномерный интервальный анализ, 2013. – 613 с. – <http://www.nsc.ru/interval>

6. Андрюхин А.И. Оценка вариаций параметров нейроморфных сетей // Научные работы Донецького національного технічного університету,

серія: «Проблеми моделювання та автоматизації проектування» № 1 (10)-2(11), 2012,с.122-131.

7. Miliotis, K. Murali, S. Sinha, W. L. Ditto, and M. L. Spano, Chaos, Solitons Fractals 42, 809 2009

Андрюхин А.И. Интервальные оценки параметров хаотических чипов. Эта работа относится к глобальной проблеме построения компьютеров нового типа, которые способны самоорганизоваться и адаптироваться в окружающей среде. В работе рассматриваются теоретические обоснования для создания всех основных логических элементов на основе одной хаотической нелинейной системы. Решена в статье задача построения интервальных оценок пороговых параметров использования и управления хаотической временной структуры, которая генерирует нелинейные временные ряды с целью получения гибких и переконфигурируемых вычислительных устройств. Приведены результаты компьютерных расчетов. В расчетах использовался пакет Mathematica.

Ключевые слова: хаотический чип, интервальная оценка, самоорганизация

Andruchin A.I. Interval estimates of parameters of chaotic chips. This work refers to the global problem of building a new type of computers that are able to self-organize and adapt in the environment. The paper discusses the theoretical foundations for the creation of all basic logic elements on the basis of a chaotic nonlinear system. Solved in the article the problem is to construct a threshold interval estimates of parameters of use and management of chaotic time structure generating non-linear time series in order to obtain a flexible and reconfigurable computing devices. The results of computer simulations are presented. The calculations used Mathematica package.

Keywords: chaotic chip interval estimate, self-organization

Статья поступила в редакцию 20.11.2016

Рекомендована к публикации д-ром физ.-мат. наук С.Н. Судаковым

УДК 004.942

Разработка скоринговой модели с использованием методов логистической регрессии и ROC – анализа

И.Ю. Анохина

Донецкий национальный технический университет
ingatula@mail.ru

Анохина И.Ю. Разработка скоринговой модели с использованием методов логистической регрессии и ROC – анализа. Рассмотрены вопросы создания скоринговой модели на основании данных медицинских исследований. При моделировании используются методы интеллектуального анализа данных, к которым относится аппарат логистического регрессионного анализа, логико-статистические подходы к распознаванию образов, в том числе ROC – анализ. Описаны этапы разработки скоринговой модели. Проведена оценка точности моделирования на основании статистических данных, полученных при клинических испытаниях.

Ключевые слова: статистические данные, скоринговые модели, методология разработки, риски, логистическая регрессия, ROC – анализ, пакет Statistica.

Скоринг и его история

Скоринг (от англ. scoring – подсчет очков в игре) - модель классификации статистической базы данных по различным группам, если неизвестна характеристика, которая разделяет эти группы, но имеется набор параметров, оказывающих влияние на исследуемую характеристику.

Классификация исследуемого объекта осуществляется на основании скоринговой карты, с помощью которой рассчитывается скоринговый балл конкретного человека.

Скоринговые модели и карты получили широкое распространение в различных областях жизнедеятельности человека.

Например, в банковской сфере это кредитный скоринг. По социальным характеристикам клиента (пол, возраст, место проживания, должность и т.д.) можно, на основе его анкеты, отнести клиента к группе сильно или слабо соответствующих бизнесу. Исходя из рассчитанного уровня доверия, принимается решение о предоставлении кредита. Основной целью скоринга является классификация клиентов банка на «хороших» и «плохих», исходя из чего кредитор может выбирать соответствующие действия по отношению к клиенту. Под «плохим» клиента, понимают клиента с низкой прогнозируемой вероятностью возвращения кредита.

Как раз с кредитного скоринга, видимо, и началась история его использования. В 1941 Давид Дюран опубликовал первую исследовательскую работу по кредитному скорингу, в которой оценивал роль различных

факторов на формирование рейтинга клиента. Взрыв спроса на кредиты после окончания Второй мировой войны обусловил необходимость автоматизации процесса принятия решений о выдаче кредитов. В 60-х годах началось внедрение компьютерных технологий в область скоринга.

Страховым компаниям необходимо принимать решение, стоит ли страховать конкретного человека и какую сумму он должен платить за страховку. При этом они основываются на вероятности наступления страховых случаев, например, аварий, если рассматривается вопрос автомобильного страхования. Факторами принятия решений являются такие как: история нарушений, опыт вождения, год приобретения и состояние автомобиля и т. д. Анализируя их, с помощью скоринговых карт клиента, прогнозируют риски.

Проведенные исследования, показали, что кредитная история является неоспоримым и точным индикатором риска того, что с данным лицом произойдет страховой случай. Для многих страхователей скоринговые системы позволяют платить меньшие страховые премии.

В фармацевтике скоринговые модели используются для определения класса препаратов, которые будут эффективны для различных групп клиентов.

В молекулярной генетике – это молекулярный диагноз заболеваний с учетом того, что генетический код можно использовать для расчета вероятности заболевания.

Постановка задачи

В современной медицине существует ряд задач, для решения которых требуются количественные, оценочные методы.

Поэтому существует необходимость в разработке математических моделей описания процессов, свойств человеческого организма и создания на их базе скоринговых карт или систем, описывающих состояние пациента, степень нарушения функций организма, степень тяжести его состояния.

Скоринговые системы позволяют определять вероятность летального исхода или выздоровления пациента, оценивать степень и стадии заболевания, причем в виде числовых характеристик, что невозможно даже для врача-клинициста с большим опытом. Такая система строится на основе модели, учитывающей симптомы заболевания и само заболевание [1].

Существуют два основных метода создания скоринговых моделей. Первый основан на анализе статистических данных, второй на экспертной оценке. Основная проблема первого метода - получение репрезентативной базы данных, содержащей не только часто встречающиеся, но и редкие случаи проявления болезней.

Второй метод (экспертные системы) - это интеллектуальные системы, ориентированные на тиражирование опыта высококвалифицированных специалистов (экспертов) в таких областях медицины, в которых качество принятия решений зависит от уровня экспертизы. Основная проблема этого метода - трудоемкость выявления факторов, сложность разработки и неполнота множества правил.

Чаще в медицине используют первый метод, основанный на анализе статистических данных.

Нам были предоставлены статистические данные, полученные путем анкетирования пациентов, проводимого в медицинских учреждениях г. Донецка. Поставлена задача разработки скоринговой модели для определения степени риска возникновения заболевания микрохолелитиаза (далее МХЛ, в просторечии, образование желудочных камней или сладжа).

Как показали исследования, существует ряд факторов, влияющих на вероятность образования сладжа. Частично эти факторы определяются путем анкетирования пациента. Учитывая, что клинические испытания достаточно дороги, целесообразно на первом этапе исследований выявить группы лиц, предрасположенных к заболеванию, и только в случае высокой степени риска назначать клинические обследования.

Была обследована группа из 131 пациента с установленным заболеванием и контрольная группа обследуемых, у которых диагноз не подтвердился.

Моделирование проводили по следующему алгоритму:

1. Создание скоринговой карты (перечня учитываемых в модели параметров, а именно факторов, способствующих возникновению заболевания).
2. Выбор метода обработки статистических данных.
3. Классификация обследуемых, т.е. создание модели на основе части выборки. Результатом этого этапа становится скоринговая модель, согласно которой по имеющейся информации о конкретном пациенте, выдается оценка степени риска заболевания в баллах.
4. Проверка модели на оставшейся части выборки.
5. При необходимости доработка модели с учетом новых статистических данных.

Первоначально исследовалась выборка, в которой количество анализируемых факторов было более двадцати. Проведение факторного анализа позволило сократить число исследуемых переменных до шести. Как показал анализ факторных нагрузок, на увеличение вероятности риска заболевания существенное влияние оказывают: наличие травм в анамнезе, индекс массы тела (здесь и далее ИМТ), уровень физической активности пациента, возраст, наследственность и изменения в режиме питания, включающие в себя соблюдение постов и голодание сроком более семи, а особенно более четырнадцати дней.

Для решения поставленной задачи выбран метод логистической регрессии.

Создание скоринговой модели

Логистическая регрессия – разновидность множественной регрессии, позволяющая установить зависимость между несколькими независимыми переменными и зависимой, причем как зависимая, так и/или независимые переменные могут принимать два значения (да/нет), в дальнейшем будем называть их бинарными.

Задачи, в которых используются бинарные коэффициенты, встречаются достаточно часто. Например, в качестве бинарной переменной можно рассматривать переменную, характеризующую отсутствие или наличие заболевания у пациентов в медицинских исследованиях; наличие собственности, вкладов в банке при построении скоринговых систем для оценки возможности кредитования или страхования; ответы да/нет при обработке результатов анкетирования в социологических

опросах и пр. [2].

Расчитанные в результате логистического анализа величины переменных в большинстве случаев принимают не точные значения 0 или 1, а значения в интервале. В этом случае полученные в ходе анализа результаты интерпретируют не как конкретные значения, а как вероятность P того, что результат может быть отнесен к определенному классу. Например, вероятность близкая к единице, позволит сделать вывод, что пациент находится в зоне риска (вероятнее болен, чем здоров) и для уточнения диагноза необходимы клинические исследования [3].

На рис.1 представлена часть исходных данных, на основании которых строилась модель.

	А	В	С	Д	Е	Г	
	Сладж	Травмы	Физическая активность	Посты, голодание	ИМТ	Наследственность	Возраст
1							
2	да	да	да	да	2	да	1
3	да	нет	да	да	1	да	2
4	нет	нет	да	нет	0	нет	1
5	нет	нет	да	нет	1	нет	1
6	нет	нет	да	нет	0	да	1
7	да	да	нет	да	0	да	3
8	нет	нет	нет	нет	0	нет	4
9	да	да	нет	да	2	нет	4
10	да	да	нет	да	3	нет	4
11	да	да	да	да	2	нет	3
12	да	да	нет	нет	3	нет	5
13	да	да	нет	да	1	да	3
14	нет	нет	да	да	1	нет	5
15	нет	нет	нет	нет	0	нет	1
16	да	нет	нет	да	1	нет	3
17	да	нет	нет	да	1	нет	1
18	да	да	нет	нет	1	да	4
19	да	нет	нет	да	1	да	3
20	нет	нет	да	нет	0	нет	5
21	да	да	нет	нет	1	нет	2
22	да	нет	нет	да	1	нет	3
23	нет	нет	нет	да	1	нет	3
24	да	нет	нет	нет	0	нет	3
25	да	нет	нет	нет	1	да	3
26	да	да	да	нет	1	да	3
27	нет	да	да	нет	0	нет	5
28	да	да	нет	нет	1	нет	2
29	да	да	нет	нет	1	да	3
30	да	да	нет	да	1	да	4
31	да	да	да	да	0	нет	2
32	нет	да	да	да	0	нет	2
33	да	да	нет	да	0	нет	5

Рисунок 1 – Исходные данные модели

Показанные на рис.1 данные первоначально были размещены в Microsoft Excel. В столбце А указан установленный диагноз (да – подтверждено наличие заболевания, 0 – нет); в столбце В – наличие ранее травм; в столбцах С и Д – данные по физической активности и резким изменениям в режиме питания. ИМТ был классифицирован следующим образом : 0 – нормальное телосложение или дефицит массы; 1 – первая стадия ожирения, 2 и 3 – соответственно вторые и третьи стадии. ИМТ определяли как отношение веса в килограммах к росту в метрах.

В норме ИМТ равен $20 \div 25 \text{ кг/м}^2$; при ожирении I степени – $25 \div 30 \text{ кг/м}^2$; при ожирении II степени – $30 \div 40 \text{ кг/м}^2$; при ожирении III степени – более 40 кг/м^2 . При дефиците массы

тела ИМТ равен $18 \div 20 \text{ кг/м}^2$.

В столбце Г – данные о наследственной предрасположенности к заболеванию.

Первичная обработка статистических данных позволила произвести градацию обследуемых по возрастному признаку (столбец Г таблицы).

Возраст до 25 лет обозначали как 1 (первая группа); от 25 до 34 лет – вторая группа; 3-ая группа – от 35 до 44 лет, 4-ая – от 45 до 54 лет; 5-ая группа – старше 54 лет.

Для обработки использовали пакет Statistica, модуль Nonlinear Estimate, в котором выбрана категория - **Logistic Regression**.

Для данных, представленных на рис.1 в качестве y рассматривали наличие заболевания ($y=1$ соответствует состоянию «сладж=да» и 0 в противном случае), в качестве независимых переменных значения в столбцах (x_1 характеризует наличие травм у пациента в анамнезе; x_2 наличие или отсутствие физической активности; x_3 – зафиксированные изменения в режимах питания; x_4 – ИМТ; x_5 - наличие наследственной предрасположенности к заболеванию; x_6 - возрастная категория).

При построении логистической модели определяются параметры уравнения вида

$$y = \frac{e^{b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + \dots + b_n \cdot x_n}}{1 + e^{b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + \dots + b_n \cdot x_n}} \quad (1)$$

где $b_0, b_1 \dots b_n$ - коэффициенты логистической регрессии, n - количество независимых переменных.

Исходя из формулы (1) можно определить вероятность наступления некоторого события, в нашем случае это вероятность подтверждения диагноза, по формуле:

$$P = \frac{1}{1 + e^{-(b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + \dots + b_n \cdot x_n)}}$$

Для решения задач логистической регрессии в пакете Statistica используется метод максимального правдоподобия.

В результате расчетов получены параметры модели:

Таблица 1. Коэффициенты логистической регрессии

b_0	b_1	b_2	b_3	b_4	b_5	b_6
-0.80	17.98	-18.29	1.33	1.51	17.92	-0.17

На рис. 2 представлено результирующее окно расчетов параметров. Кроме значений коэффициентов логистической регрессии, которые отображаются в строке **Estimate**, в окне отображения результатов указан p -уровень гипотезы. Если p -уровень менее 5%, модель считается значимой. В нашем случае $p=0.0001472$, что позволило сделать вывод о

значимости разработанной модели.

Оценивая величину хи-квадрата Пирсона (Chi-square =26.97191 на рис. 2) и сравнивая полученное значение критерия хи-квадрат с критическим, получаем $26.97191 > 9.488$, откуда делаем вывод о наличии статистически значимой зависимости параметра Y от выбранных независимых переменных X . Уровень значимости данной взаимосвязи соответствует $p < 0.05$.

Results

```
Model is: logistic regression (logit) No. of 0
                                         No. of 1
Dependent variable:      Y              Independ
Loss function is: maximum likelihood Final va
-2*log(Likelihood): for this model = 12.77762
Chi-square = 26.97191, df = 6, p = .0001472
```

Рисунок 2 – Параметры модели логистической регрессии

Еще одним важным шагом в проверке качества построенной модели является оценка параметра *Отношение несогласия*. На рис. 3 показана таблица с числом наблюдений, которые были правильно или неправильно классифицированы в соответствии с полученной моделью [4]. В данной статье рассматривается подвыборка, состоящая из 22 человек, которые классифицированы как больные, и десяти человек – условно здоровые.

Classification of Cases [new.sta]			
Continue...		Odds ratio: 90.000	
Observed	Pred. 0	Pred. 1	Percent Correct
0	9	1	90.00000
1	2	20	90.90909

Рисунок 3 – Принадлежность наблюдений к классам

Все наблюдения с предсказанными значениями (вероятностью) меньше или равными 0.5 классифицируются в пакете Statistica как неудача – «Failure», остальные, с предсказываемыми значениями больше 0.5, классифицируются как успех – «Success». *Отношение несогласия* вычисляется как отношение произведения чисел правильно классифицированных наблюдений к произведению чисел неправильно классифицированных результатов [5].

В столбце *Percent Correct* приведены проценты совпадений исходных и прогнозируемых данных. Так, в строке,

соответствующей $Observed=0$ (по нашей классификации сладж не обнаружен), рассматривается процент правильной классификации здоровых пациентов. Следовательно, в 90.9 случаях из 100 здоровый пациент идентифицируется именно как здоровый, и менее, чем в 10 случаях из 100 он будет считаться больным ошибочно. Это объясняется тем, что даже у здорового человека существует определенный процент риска, обусловленный его возрастом, полом, массой тела, наследственной предрасположенностью, наличием серьезных травм, но далеко не все люди страдают этим заболеванием.

Во второй строке, соответствующей $Observed=1$ (подтверждено наличие сладжа), выводится процент правильно идентифицированных больных. В 90 случаях из 100 модель позволяет предсказать наличие заболевания и в 10 случаях из ста больной пациент будет считаться здоровым (значение $Percent\ Correct=90$), т.е. болезнь не будет обнаружена на основании расчета вероятности риска по скоринговой модели.

Проведенные исследования и сопоставление полученных статистических данных и прогноза позволили считать возможным использование разработанной модели для начальной, доклинической диагностики.

Подставляя полученные коэффициенты логистической регрессии (см. табл. 1) в уравнение (1) можно получить прогнозируемые значения величины $Y_{теор}$.

Отметим, что если в исходном наборе данных зависимая переменная Y принимает значения 0 или 1 (да/нет), то результатом расчетов по полученной зависимости является массив данных в интервале от 0 до 1.

Например, для приводимого в табл.1. набора данных («нет» кодировалось как ноль, «да» – как единица), теоретические значения Y имели вид, показанный в табл. 2. В таблице приводятся первые десять данных выборки.

В табл. 2 в строке $Y_{теор}$ показана предсказанная вероятность того, что пациента можно отнести к классу больных, через Y обозначены реальные данные как в виде варианта да/нет, так и их бинарные значения.

Как видим, далеко не все значения во второй строке таблицы равны нулю или единице. В большинстве случаев значения колеблются в указанном интервале. В связи с этим возникает вопрос, следует ли считать больным, например, пациента под номером 5 с рассчитанной вероятностью 0.208 или пациента под номером 2 ($Y_{теор}=0.792$).

Таким образом, при расчете значения $Y_{теор}$ по уравнению логистической регрессии (1), возникает проблема, каким образом должны

быть преобразованы полученные числовые значения $Y_{\text{теор}}$ в бинарные классификаторы.

Таблица 2. Коэффициенты логистической регрессии

№	Сладж (Y)	$Y_{\text{теор}}$
1	да	1
2	да	1
3	нет	0
4	нет	0
5	нет	0
6	да	1
7	нет	0
8	да	1
9	да	1
10	да	1

Следовательно, необходимо определить такое пороговое значение P , при котором выполняется система неравенств:

$$\begin{aligned} y \geq P_{\text{порог}} &\rightarrow \text{пациент болен} \\ y \leq P_{\text{порог}} &\rightarrow \text{пациент здоров} \end{aligned} \quad (2)$$

Т.е. возникает вопрос, как определить пороговое значение вероятности, которое разделяет исследуемое множество на два класса: здоров/болен; кредитоспособен/ не кредитоспособен, эффективно лекарство для группы пациентов или нет. Причем от величины принятого порогового значения зависит точность модели и результаты предварительного диагностирования пациента.

Пороговое значение изменяется в интервале от нуля до единицы. Варьируя значение порогового P , каждый раз при расчетах будут получаться новые значения бинарного классификатора. В нашем случае, это то значение вероятности $P_{\text{порог}}$, которое следует использовать при разделении пациентов на больных и здоровых, формула (2).

Например, если в качестве порогового значения выбрать величину 0.5, то возникает проблема частичного отсечения больных пациентов, набравших недостаточно баллов риска.

Для решения этой проблемы использовали аппарат ROC (**R**eceiver **O**perating **C**haracteristic) – анализа.

ROC – анализ долгое время использовали в теории обработки сигналов и радиолокации для описания события правильного обнаружения сигнала и исключения ложных срабатываний. В настоящее время ROC

– анализ нашел широкое применение не только в задачах медицинской диагностики, но и в банковском, кредитном скорринге, теории принятия решений.

ROC-кривая показывает зависимость количества верно классифицированных наблюдений со значением параметра, равным нулю, от количества неверно классифицированных наблюдений со значением параметра, равным единице. При этом предполагается, что у классификатора имеется параметр, варьируя который, будет происходить разделение на два класса. Этот параметр часто называют порогом, или точкой отсечения (cut-off value). В зависимости от него будут получаться различные значения *ошибок I и II рода*.

На основании данных, приведенных на рис. 3, можно определить значения, необходимые для проведения ROC – анализа, а именно:

TP (*True Positives*) – верно классифицированные случаи с заболеванием, для рассматриваемого примера TP=20;

TN (*True Negatives*) – верно классифицированные случаи с отсутствием заболевания, для рассматриваемого примера TN=9;

FN (*False Negatives*) – случаи с заболеванием, но при прогнозировании оно выявлено не было (*ошибка I рода*), так называемый «ложный пропуск» – болезнь не была обнаружена, для рассматриваемого примера FN=2;

FP (*False Positives*) – случаи, когда было вынесено решение о наличии заболевания, хотя его не было (*ошибка II рода*). Это ложное обнаружение, т.к. при отсутствии события ошибочно выносятся решение о его наличии, для рассматриваемого примера FP=1.

Для построения ROC-кривых необходимо вычислить два параметра: чувствительность (**Sensitivity**) и специфичность (**Specificity**).

Чувствительность определяли по формуле:

$$SE = \frac{TP}{TP + FN} \cdot 100\% \quad (3)$$

Специфичность считали как:

$$SP = \frac{TN}{TN + FP} \cdot 100\% \quad (4)$$

Эти два параметра используются для оценки требуемого уровня точности решения поставленной задачи. С медицинской точки зрения существуют следующие правила:

– если ставится задача максимального предотвращения пропуска больных, то должен

быть выбран высокий уровень чувствительности SE (гипердиагностика);

– если же лечение связано с серьезными побочными эффектами, то гипердиагностика не допустима и следует использовать более высокие уровни специфичности SP [6].

В первом случае увеличивается риск того, что здоровым пациентам будет поставлен диагноз «болен», во втором - риск не обнаружить болезнь

В табл.3. показаны рассчитанные значения перечисленных выше параметров TP, TN, FN, FP и величин чувствительности и специфичности, которые будут использованы при построении ROC – кривой. Параметры определялись для каждого значения порога отсечения $P_{\text{порог}}$ в пределах от 0 до 1 с выбранным шагом 0.1.

Кроме этого определяли величину FPR – процент ложно обнаруженных случаев заболевания, где

$$FPR = \frac{FP}{TN + FP} \cdot 100\% . \quad (5)$$

При построении ROC-кривой по оси Y откладывают чувствительность SE, по оси X – значения параметра FPR.

Таблица 3. Параметры для построения ROC - кривой

№	$P_{\text{порог}}$	TP	TN	FN	FP	SE	SP	FPR
1	0.1	22	3	0	7	100%	30%	70%
2	0.2	22	4	0	6	100%	40%	60%
3	0.3	21	6	1	4	95%	60%	40%
4	0.4	20	8	2	2	91%	80%	20%
5	0.5	19	9	3	1	86%	90%	10%
6	0.6	19	9	3	1	86%	90%	10%
7	0.7	19	9	3	1	86%	90%	10%
8	0.8	19	9	3	1	86%	90%	10%
9	0.9	17	9	5	1	77%	90%	10%
10	1	0	10	22	0	0%	100%	0%

На рис. 4 показана построенная ROC – кривая.

Идеальная ROC-кривая должна проходить через верхний левый угол, в котором процент правильно идентифицированных случаев составляет 100% (идеальная чувствительность), а процент ошибки равен нулю.

Ясно, что в реальности построенная кривая должна стремиться к идеальной, однако, вряд ли будет достигать ее.

Отсюда следует, что чем ближе ROC -

кривая к верхнему левому углу, тем выше уровень качества прогнозирования разработанной модели. И наоборот, чем меньше изгиб кривой и чем ближе она расположена к диагональной прямой (прямая линия под углом 45°, соединяющая левый нижний и правый верхний углы, на рис.4 показана пунктиром), тем менее эффективна модель.

Близость ROC-кривой к диагональной линии соответствует так называемому «бесполезному» классификатору, т.е. полной неразличимости классов [7,8]. А, следовательно, свидетельствует о не соответствии разработанной модели полученным статистическим данным.

Как можно заметить, построенная по разработанной модели ROC – кривая достаточно близка к идеальной.

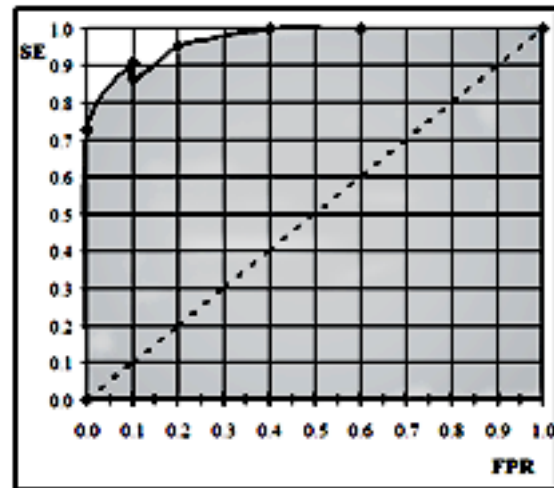


Рисунок 4 – ROC - кривая

Визуальная оценка кривых ROC не всегда позволяет выявить наиболее целесообразное значение порога и численно оценить качество разработанной модели.

Часто для оценки качества ROC-кривых используют оценку площади под кривыми (на рис. 4 закрашена серым цветом). Эта оценка может быть получена непосредственно вычислением площади под многогранником, ограниченным справа и снизу осями координат и слева сверху – экспериментально полученными точками (рис. 4).

Численный показатель площади под кривой называется AUC (Area Under Curve).

В литературе приводится экспертная шкала для значений AUC, по которой можно судить о качестве разработанной модели [9,10], см. табл. 4.

Нами было вычислено значение AUC, величина составила более 96%, что позволяет сделать вывод об «отличном» качестве модели.

Отметим, что на первом этапе

исследований нами не рассматривались такие факторы, как наследственность и возраст. Полученные результаты (площадь около 83%) говорили об «Очень хорошем» качестве. Введение дополнительных параметров позволило улучшить качество модели.

Таблица 4. Интервалы площади под ROC-кривой и оценка качества модели

Интервал AUC	Качество модели
0.9÷1.0	Отличное
0.8÷0.9	Очень хорошее
0.7÷0.8	Хорошее
0.6÷0.7	Среднее
0.5÷0.6	Неудовлетворительное

Наряду с вышеперечисленными факторами, на первом этапе исследований рассматривали влияние пола, приема определенного вида лекарств на вероятность возникновения болезни. Однако, эти факторы не оказали существенного влияния на точность модели и были исключены из рассмотрения.

Как видно из данных табл. 3 и рис. 4, значения рассматриваемых параметров SE, SP, FPR не меняются для порога, находящегося в пределах

$$0.5 \leq P_{\text{порог}} \leq 0.9 \quad (6)$$

Соответствующие строки выделены в табл. 3. Это означает, что пороговые значения могут быть любым числом в указанном интервале.

Неоднозначность полученного ответа обусловила необходимость в дополнительных исследованиях с целью уточнения порогового значения.

В качестве дополнительной проверки качества модели и уточнения интервала порогового значения определяли отношение суммы правильно идентифицированных случаев (как при отсутствии болезни, так и в случае ее наличия) к общему числу имеющихся экспериментальных данных. Данный параметр определили как переменную $F = (TP+TN)/N$.

Результаты зависимости переменной F от величины порога показаны в табл. 5.

Для уточнения значения порогового значения была поставлена и решена задача подбора параметров аппроксимирующей функции, описывающей зависимость числа правильно идентифицированных случаев от величины порога.

Для решения задачи использовали модуль Nonlinear Estimate пакета Statistica. Проведен подбор коэффициентов полиномиального уравнения. Как показала оценка погрешности подбора, полином пятой степени описывает данные с наименьшей погрешностью [11].

Таблица 5. Зависимость параметра F от величины порога

$P_{\text{порог}}$	F
0.1	81%
0.2	88%
0.3	91%
0.4	91%
0.5	91%
0.6	91%
0.7	91%
0.8	88%
0.9	81%

На рис. 5 показано результирующее окно пакета Statistica.



Рисунок 5 – Расчет параметров полиномиальной функции в пакете Statistica

Зависимость переменной F от величины порога можно описать уравнением вида:

$$F = 0.67 + 1.90 \times P_{\text{порог}} - 5.72 \times P_{\text{порог}}^2 + 7.64 \times P_{\text{порог}}^3 - 3.82 \times P_{\text{порог}}^4 \quad (7)$$

Дифференцируя функцию F , находим точку экстремума. Как показали расчеты, при значении порога $P_{\text{порог}} = 0.62$ имеет место максимальное количество правильно диагностированных случаев, равное 91%. Это значение является уточненным значением порога, см. формулу (6).

Таким образом, алгоритм идентификации пациентов свелся к нескольким шагам:

1. ввести значения физических характеристик обследуемого (возраст, значение ИМТ и пр. факторы, указанные в табл.1);
2. рассчитать по уравнению

логистической регрессии значения $Y_{теор}$

$$e^{-0.8+17.98 \cdot x_1 - 18.29 \cdot x_2 + 1.33 \cdot x_3 + 1.54 \cdot x_4 + 17.92 \cdot x_5 - 0.17 \cdot x_6}$$

$$1 + e^{-0.8+17.98 \cdot x_1 - 18.29 \cdot x_2 + 1.33 \cdot x_3 + 1.54 \cdot x_4 + 17.92 \cdot x_5 - 0.17 \cdot x_6} ;$$

3. сравнить полученные значения с пороговым значением (в нашем примере 0.62);

4. если $Y_{теор} > 0.62$ можно сделать вывод о высокой вероятности заболевания и необходимости клинического диагностирования;

5. в противном случае следует считать, что обследуемый скорее здоров, чем болен.

Выводы

В работе рассмотрены вопросы разработки скоринговой модели, позволяющей по предварительному анкетированию пациентов без дополнительной клинической диагностики оценивать степень возможного риска заболевания.

Выявлены факторы, оказывающее влияние на вероятность возникновения заболевания. Проведена оценка качества модели. Точность проведенных расчетов более 91%, что позволяет рекомендовать предложенную методику для составления скоринг - карт пациентов.

Литература

1. Мильчаков К.С., Шебалков М.П. Скоринговые карты в медицине: обзор и анализ публикаций. // Врач и информационные технологии, №1, 2015. - с.71-79.

2. Методы статистической обработки медицинских данных: Методические рекомендации для ординаторов и аспирантов медицинских учебных заведений, научных работников. / сост.: Кочетов А.Г., Лянг О.В.,

Масенко В.П., Жиров И.В., Наконечников С.Н., Терещенко С.Н. – М.: РКНПК, 2012. – 42 с.

3. Леонов В. Логистическая регрессия в медицине и биологии / Биометрика. Журнал для медиков и биологов, сторонников доказательной биомедицины. - Режим доступа: http://www.biometrica.tomsk.ru/logit_1.htm.

4. Бэстенс Д.-Э., Ван Ден Берг В.-М., Вуд Д. Нейронные сети и финансовые рынки. Принятие решений в торговых операциях. - М.: Научное издательство ТВП, 1997. - 235с.

5. Логистическая регрессия / Портал Знаний StatSoft. - Режим доступа: <http://statistica.ru/theory/logisticheskaya-regressiya/>

6. Логистическая регрессия и ROC-анализ - математический аппарат / Base Group Lab. - Режим доступа: <https://basegroup.ru/community/articles/logistic>.

7. Бююль А., Цёфель П. SPSS: Искусство обработки информации. Анализ статистических данных и восстановление скрытых закономерностей: - Пер. с нем. - СПб.: ДиаСофтЮП, 2005. - 608 с.

8. Логистическая регрессия. / Центр статистического анализа. - Режим доступа: <http://statmethods.ru/konsalting/statistics-metody/116-logisticheskaya-regressiya.html> .

9. Цыплаков А.А. Некоторые эконометрические методы. Метод максимального правдоподобия в эконометрии. Методическое пособие. - Новосибирск: НГУ, 1997. - 129 с.

10. Сергиенко В.И., Бондарева И.Б. Математическая статистика в клинических исследованиях. - М.: ГЕОТАРДМЕД, 2001. - 256 с.

11. Боровиков В.П., Боровиков И.П. Statistica. Статистический анализ и обработка данных в среде Windows. -М.: Информационно-издательский дом "Филинъ", 1997.- 608с.

Анохина И.Ю. Разработка скоринговой модели с использованием методов логистической регрессии и ROC – анализа. Рассмотрены вопросы создания скоринговой модели на основании данных медицинских исследований. При моделировании использованы методы интеллектуального анализа данных, к которым относится аппарат логистического регрессионного анализа, логико-статистические подходы к распознаванию образов, в том числе ROC – анализ. Описаны этапы разработки скоринговой модели. Проведена оценка точности моделирования.

Ключевые слова: статистические данные, скоринговые модели, методология разработки, риски, логистическая регрессия, ROC – анализ, пакет Statistica.

Anokhina I.Y. *Development of a scoring model using logistic regression and ROC methods - analysis. The problems of creating a scoring model based on medical research. The simulation used the methods of data mining, which includes the device logistic regression analysis, logical-statistical approaches to pattern recognition, including ROC - analysis. We describe the stages of development of the scoring model. The accuracy of the simulation.*

Keywords: *statistics, scoring models, methodology development, risks, logistic regression, ROC - analysis, Statistica package.*

*Статья поступила в редакцию 20.11.2016
Рекомендована к публикации д-ром техн. наук В.Н. Павлышом*

УДК 004.7

Система формирования трафика компьютерной сети на основе самоорганизованной критичности

Д.В. Бельков, Е.Н. Едемская

Донецкий национальный технический университет
belkov65@list.ru

Бельков Д.В., Едемская Е.Н. Система формирования трафика компьютерной сети на основе самоорганизованной критичности. Одной из парадигм синергетики является теория самоорганизованной критичности (SOC). В случае подхода SOC фрактальный сигнал является выходным сигналом нелинейной динамической системы, находящейся в критическом состоянии. Выходной сигнал – результат процесса самоорганизации. В данной работе для формирования трафика при дифференцированном обслуживании (DiffServ) предлагается система, основанная на понятии самоорганизованной критичности (Self-Organized Criticality, SOC). Разработана модель SOC BTWIDm, являющаяся модификацией модели Бака-Танга-Вайзенфельда. Предложена структура системы формирования трафика, выполняющая алгоритм BTWIDm. Приведены результаты вычислительного эксперимента при формировании самоподобного трафика. Исследования выполнены в среде Matlab. Самоподобный входной трафик был сглажен с помощью предложенной системы. Коэффициент пачечности выходного потока уменьшился на 20 единиц относительно коэффициента пачечности входного потока.

Ключевые слова: самоподобный трафик, качество обслуживания, дифференциальное обслуживание, система формирования трафика, самоорганизованная критичность, модель Бака-Танга-Вайзенфельда

Введение

В условиях развития телекоммуникаций до уровня сетей следующего поколения NGN (Next Generation Networks), обеспечивающих предоставление неограниченного набора услуг с заданными характеристиками качества обслуживания QoS (Quality of Service), являются актуальными задачи повышения качества обслуживания. Расширение спектра услуг и рост сложности телекоммуникационных систем и сетей требует решения проблемы разработки адекватных методов анализа и синтеза таких систем с целью получения достоверных оценок их характеристик, реализации задач их оптимизации в соответствии с выбранным критерием качества обслуживания и разработки соответствующих алгоритмов управления.

Количественная сторона процессов обслуживания потоков сообщений (трафика) в системах распределения информации (СРИ) является предметом теории трафика. На основе этой теории разрабатываются методы оценки характеристик качества обслуживания. Основным содержанием теории трафика является исследование пропускной способности СРИ, тесно связанное с оценкой QoS.

Теория трафика оперирует не с самими СРИ, а с их математическими моделями.

Многообразие видов сетей и способов выделения сетевого ресурса для обслуживания трафика требует разработки моделей, которые учитывают реальный характер потоков сообщений и детали обслуживания мультисервисного трафика различных коммуникационных приложений. Построить единственную модель, которая бы давала ответы на все вопросы относительно функционирования компьютерных сетей практически невозможно. На основе моделей СРИ разрабатываются методы оценки характеристик QoS, достоверность которых зависит от адекватности модели реальной ситуации, которая может возникнуть при проектировании и эксплуатации. Оценка качества обслуживания трафика является одним из важнейших научных направлений в исследованиях компьютерных сетей. В современных сетях потоки пакетов формируются множеством источников требований на предоставление сетью услуги и сетевых приложений, обеспечивающих услуги передачи информации. Пользователи, создающие потоки пакетов, существенно отличаются между собой значениями удельной интенсивности нагрузки, и поэтому трафик является неоднородным. Он имеет определенные требования к QoS и долгосрочные зависимости в интенсивности, например, из-за повторной передачи неверно принятых пакетов. Передачу

потоков различных служб обеспечивает единая сеть. Поскольку источники каждой службы могут иметь разные скорости передачи информации или изменять ее в процессе сеанса связи, то потокам пакетов свойственна пачечность (burstness). В реализации трафика присутствует некоторое количество больших пульсаций при относительно малом среднем уровне трафика. Пакеты поступают на узел не по отдельности, а пачкой. Это явление ухудшает характеристики (увеличивает потери, задержки, джиттер пакетов) при прохождении трафика через узлы сети. При анализе пачечного трафика может быть использована модель самоподобного (self-similarity) процесса [1-4].

Для формирования самоподобного трафика согласно определенным требованиям в данной работе предлагается система, основанная на понятии самоорганизованной критичности (Self-Organized Criticality, SOC).

Целью статьи является разработка системы формирования трафика на базе SOC. В работе решаются задачи построения модели SOC BTW1Dm и создания структуры системы формирования трафика, которая выполняет алгоритм BTW1Dm. Показаны результаты вычислительного эксперимента.

Модели BTW1D и BTW1Dm

В нелинейной динамике разработаны три парадигмы. В рамках первой было показано, что во многих открытых нелинейных системах вдали от равновесия происходит самоорганизация. При этом возникают пространственно-неоднородные стационарные структуры, которые И. Р. Пригожин предложил называть диссипативными структурами. Во второй парадигме основное внимание было уделено динамическому хаосу – сложному непериодическому поведению в детерминированных системах, где нет случайных факторов. Основным результатом исследований стало установление факта границ прогнозирования, т.е. существования конечного времени, через которое предсказание поведения системы становится невозможным. В основе этих парадигм лежат представления о самоорганизации и выделении из большого числа переменных, описывающих систему, небольшого числа величин, которые называются параметрами порядка. К ним на больших интервалах времени подстраиваются остальные степени свободы системы.

Третья парадигма связана с теорией самоорганизованной критичности. Эта теория призвана объяснить такие закономерности, как масштабная инвариантность, прерванное равновесие, фликкер-шум [5]. Нужно отметить, что именно такие закономерности

экспериментально найдены при исследованиях современного трафика [6].

В 1988 году американскими учеными П. Баком, Ч. Тангом и К. Вайзенфельдом было введено понятие о самоорганизованной критичности [7]. Базовой моделью теории SOC является куча песка. Рассмотрим кучу песка, изображенную на рисунке 1.

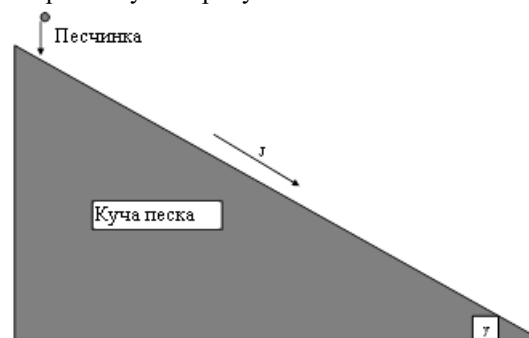


Рисунок 1 – Куча песка

Будем предполагать сцепление между песчинками достаточно большим. При этом возможно лишь поверхностное перемещение песка, причем инерцией его движения можно пренебречь. В таком случае состояние системы полностью определяется наклоном поверхности. В тех местах, где локальный наклон оказывается больше порога устойчивости, происходит осыпание песчинок вниз по склону на соседние участки поверхности. Состояние песка определяется углом наклона поверхности γ . При его изменении происходит фазовый переход от неподвижного состояния к состоянию непрерывного тока песка. При токе, соответствующем добавлению одной песчинки за один шаг, система самоорганизуется в состояние с критическим наклоном $\gamma = \gamma_c$. Если средний наклон поверхности γ меньше чем γ_c , то песок неподвижен. Если наклон превышает значение γ_c , возникает спонтанный ток песка J по поверхности. Возникает фазовый переход, в котором управляющим параметром является наклон γ , а параметром порядка – ток песка J . Критическое значение наклона γ_c разделяет хаотическую ($\gamma < \gamma_c$) и упорядоченную ($\gamma > \gamma_c$) фазы. Обе эти фазы соответствуют состоянию системы устойчивому к малым возмущениям. В хаотической фазе они еще быстро затухают во времени и пространстве, а в упорядоченной – уже не могут существенно влиять на величину тока. И только в критической точке, одна добавленная песчинка может вызывать лавину. В песчаной куче существует много областей неустойчивости, но критическое состояние является устойчивым. Такие характеристики тесно связаны с фракталами.

Самым простым вариантом модели кучи песка является одномерная модель Бака-Танга-Вайзенфельда (BTW-1D). Ее работу рассмотрим на примере [8], показанном на рисунке 2.

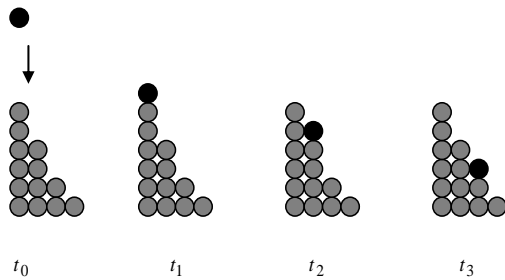


Рисунок 2 – Модель BTW-1D

Пусть задан критический порог $z_c = 2$ и куча песка, имеющая 4 слоя. В начальном состоянии первый слой кучи содержит 6 песчинок, второй – 4, третий – 2 и четвертый слой – 1 песчинку. Уровень песка в каждом слое отличается от уровня предыдущего слоя не больше, чем на величину z_c : $(6-4)=2$, $(4-2)=2$, $(2-1)=1$. По этой причине куча песка находится в устойчивом состоянии (песок неподвижен). Пуск модели выполняется добавлением новой песчинки на вершину. При этом уровень песка в первом слое повышается, и он становится неустойчивым: $(6+1)=7$, $(7-4)=3 > z_c$. В таком случае песчинка не остается на первом слое, а передвигается на второй слой и количество песка в первом слое возвращается к значению 6. Уровень песка во втором слое повышается и он становится неустойчивым: $(4+1)=5$, $(5-2)=3 > z_c$. Поэтому песчинка не останавливается на втором слое, передвигается на третий слой и количество песка во втором слое возвращается к значению 4. Уровень песка в третьем слое повышается, но он остается устойчивым: $(2+1)=3$, $(3-1)=2 = z_c$. Поэтому песчинка останавливается на третьем слое. Первый шаг моделирования завершен. На нем прошла лавина перемещений от первого к третьему слою.

Каждый новый шаг будет начинаться добавлением новой песчинки на вершину. На одном шаге моделирования возможно 3 ситуации: 1) новая песчинка остается на вершине (лавина нет), 2) новая песчинка останавливается на каком-то слое кучи, т.е. лавина не достигает основания кучи, 3) новая песчинка не останавливается ни на одном слое кучи и покидает кучу (лавина достигает основания кучи).

Такой процесс формирования кучи песка является процессом самоорганизации потому, что он идет без внешнего управляющего сигнала. Рост уровня слоев управляется правилами (1),

где z_i – количество песчинок, содержащихся в слое i .

$$\begin{aligned} z_{i+1} &= z_{i+1} + 1, z_i - z_{i+1} > z_c \\ z_i &= z_i + 1, z_i - z_{i+1} \leq z_c \end{aligned} \quad (1)$$

Величины z_i являются управляющими параметрами системы, значение z_c является критическим значением для управляющих параметров. Достижение этого значения качественно изменяет поведение системы. Процесс формирования кучи песка является процессом самоорганизации управляющих параметров в критическое состояние и поэтому система является системой самоорганизованной критичности.

В данной статье для системы формирования сетевого трафика предлагается модифицированная модель BTW-1Dm. Модификация заключается в следующем: 1) каждая песчинка имеет свой объем, т.е. песчинки не являются одинаковыми, 2) каждый слой имеет свой критический уровень (порог), 3) рост уровня слоев управляется правилами (2), где $z_{i,j}$ – количество песчинок, содержащихся в слое i на временном шаге j , x_j – объем песчинки j , z_{ci} – критический порог слоя i . Если добавление песчинки j к слою i приводит к превышению критического порога z_{ci} , то она перемещается на следующий уровень, иначе – песчинка добавляется на уровень i .

$$\begin{aligned} z_{i+1,j} &= z_{i+1,j} + x_j, z_{i,j} + x_j - z_{i,j-1} > z_{ci} \\ z_{i,j} &= z_{i,j} + x_j, z_{i,j} + x_j - z_{i,j-1} \leq z_{ci} \end{aligned} \quad (2)$$

Величины $z_{i,j}$ являются управляющими параметрами системы, значение z_{ci} является критическими значениями для управляющих параметров. Такая модификация сохраняет самоорганизованную критичность системы.

Система формирования трафика

Для обеспечения гарантированного качества обслуживания комитет IETF разработал модель дифференцированного обслуживания разнотипного трафика - DiffServ. Пакеты каждого класса должны обрабатываться в соответствии с определенными для этого класса требованиями качества обслуживания. Модель DiffServ описывает архитектуру сети как совокупность пограничных участков и ядра. Пример сети соответствующий этой модели показан на рисунке 3. Пограничные маршрутизаторы классифицируют и нормализуют трафик. Нормализация трафика предусматривает измерение его параметров, проверку соответствия заданным правилам предоставления услуг, профилирование. При

этом пакеты, не удовлетворяющие установленным правилам, теряются.

Преимущество модели DiffServ заключается в том, что она позволяет разделить весь трафик на небольшое количество классов и не анализировать каждый информационный поток отдельно. Для DiffServ при независимом поведении маршрутизаторов (PHB) определены два варианта обслуживания:

- срочная пересылка пакетов (Expedited Forwarding, EF);
- гарантированная пересылка пакетов (Assured Forwarding, AF).

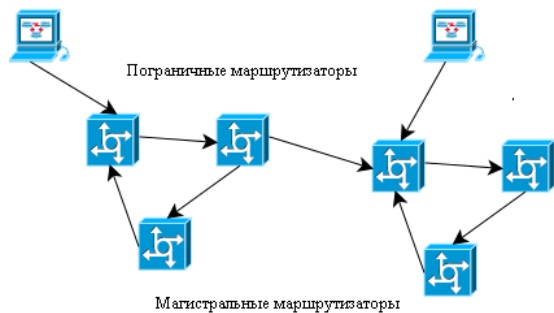


Рисунок 3 – Модель DiffServ

Вариант EF является наивысшим уровнем качества обслуживания. Любой трафик, с интенсивностью превышающей заданный уровень, отбрасывается. Вариант AF предусматривает четыре класса трафика. Трафик, интенсивность которого превышает заданный уровень, доставляется с меньшей вероятностью. Качество его обслуживания может быть снижено.

Механизм обеспечения QoS на уровне сетевого устройства в DiffServ, использует четыре операции. Сначала пакеты классифицируются на основании их заголовков. Потом они маркируются в соответствии с выполненной классификацией (в поле DiffServ). В зависимости от маркировки выбирается алгоритм передачи, позволяющий избежать заторов в сети. Заключительная операция состоит в организации очередей с учетом приоритетов [9].

На рисунке 4 показана структура сетевого процессора C5e фирмы Motorola [10].

На рисунке 4 выделена часть структуры, совпадающая по своему назначению со структурой, которая разрабатывается в статье на базе SOC. В сетевом процессоре C5e используется разновидность алгоритма Token Bucket - srTCM. Он контролирует пиковую (PIR) и выполнимую (CIR) скорость трафика, создает поток зеленых конформных пакетов, поток желтых малоконформных пакетов, поток красных неконформных пакетов.

Diffserv QoS Implementation for Wireless Network Interface (WNI)

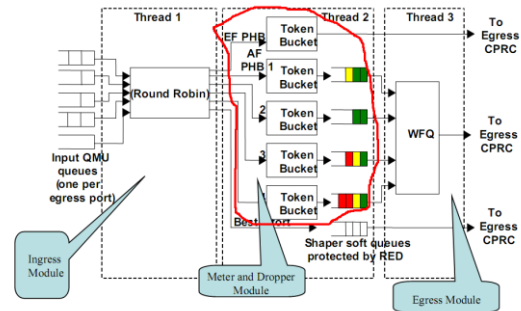


Рисунок 4 – Структура сетевого процессора C5e

Предлагаемая в статье система формирования трафика показана на рисунке 5.

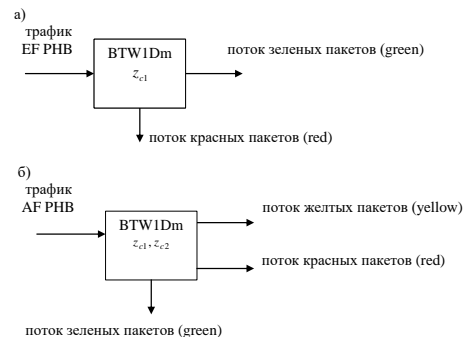


Рисунок 5 – Система формирования трафика на базе SOC

Система выполняет алгоритм BTW1Dm при следующей интерпретации для входного трафика EF PHB: $z_{1,j}$ - уровень скорости выходного потока зеленых конформных пакетов на такте j , $z_{2,j}$ - уровень скорости выходного потока красных неконформных пакетов на такте j , x_j - скорость входного потока на такте j , z_{c1} - заданная достижимая скорость трафика. Система по формулам (2) формирует и маркирует выходные потоки.

Если система применяется для входного трафика AF PHB определенного класса, то используется алгоритм BTW1Dm при следующей интерпретации: $z_{1,j}$ - уровень скорости выходного потока зеленых конформных пакетов на такте j , $z_{2,j}$ - уровень скорости выходного потока желтых малоконформных пакетов на такте j , $z_{3,j}$ - уровень скорости выходного потока красных неконформных пакетов на такте j , x_j - скорость входного потока на такте j , z_{c1} - заданная достижимая скорость трафика z_{c2} - заданная пиковая скорость трафика.

Рассмотрим предложенную систему в сравнении с Token Bucket. Система Token Bucket имеет: входной буфер пакетов, буфер выходных пакетов, буфер жетонов и сервер для сравнения входного потока с потоком жетонов. Жетоны генерируются с заданной интенсивностью и помещаются в очередь. Если буфер жетонов полон, то новый жетон теряется. В случае отсутствия жетонов пакеты должны сохраняться в буфере. Алгоритм Token Bucket основан на сравнении трафика с эталонным потоком, который представлен жетонами. Реализация алгоритма требует наличия счетчика, считающего жетоны. Счетчик увеличивается на N байт через равные интервалы времени и уменьшается при передаче пакета на объем переданного пакета. Когда счетчик уменьшается до нуля, передача пакетов останавливается. В случае обслуживания EF пакет перемещается сервером из входного буфера в выходную очередь зеленых пакетов, если в ведре достаточно жетонов, иначе сервер перемещает пакет в очередь красных пакетов.

Состав системы Token Bucket показан на рисунке 6а. Состав предлагаемой системы показан на рисунке 6б.

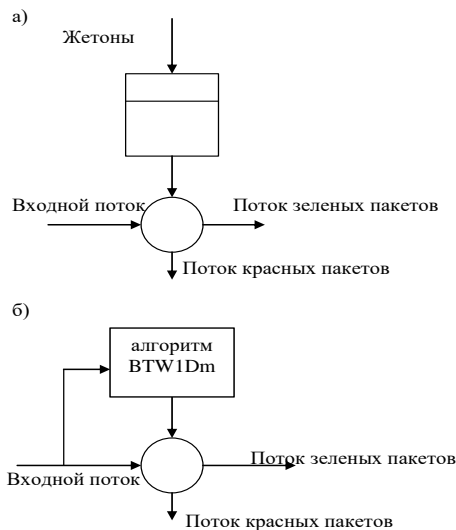


Рисунок 6 – Структуры Token Bucket и BTW1Dm

В предложенной системе входной пакет попадает во входной буфер, его копия поступает в модуль BTW1Dm, который маркирует пакет по формулам (2). Сервер согласно маркировке перемещает пакет в выходную очередь зеленых пакетов или в очередь красных пакетов.

Разработанная система предназначена для достижения того же результата, что и Token Bucket. В данной статье не ставится цель показать преимущества предложенной системы перед Token Bucket. Такая задача нуждается в дополнительных исследованиях. Однако разработанная система имеет принципиальное

отличие от ведра жетонов. Система Token Bucket является организационной (кибернетической) системой. Объект управления (входной пакет) изменяет свое состояние (перемещается из входного буфера в выходную очередь) при поступлении внешнего управляющего сигнала (жетона из ведра жетонов). Предложенная в статье система является самоорганизованной (синергетической) системой. Объект управления (входной пакет) изменяет свое состояние (перемещается из входного буфера в выходную очередь) без внешнего управляющего сигнала. Такое перемещение выполняется при поступлении внутреннего управляющего сигнала (маркированной копии пакета из модуля BTW1Dm).

Для проверки работы предложенной системы выполнен вычислительный эксперимент в среде Matlab. Входной трафик показан на рисунке 7.

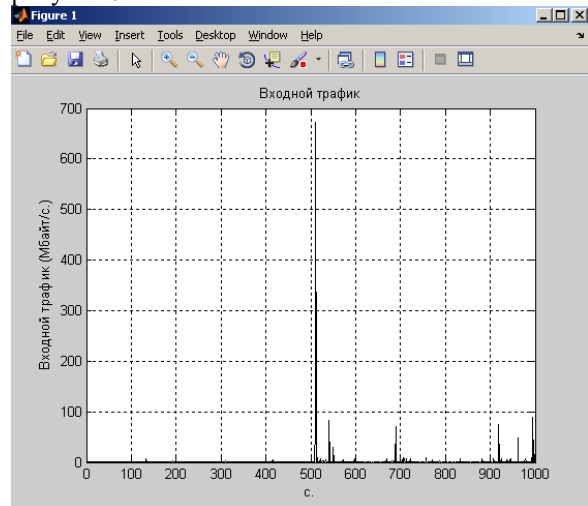


Рисунок 7 – Входной трафик

Это временной ряд скорости (Мбайт/с.) пакетов беспроводной сети доступа [11]. В первом варианте эксперимента работа системы исследовалась для случая передачи в режиме AF PNB при Мбайт/с. (CIR) и Мбайт/с. (PIR). Результаты эксперимента показаны на рисунках 8-10.

Система сформировала выходной поток зеленых пакетов, который является сглаженным потоком по отношению ко входному трафику. Были сформированы желтый поток малоконформных пакетов, который составляют более скоростные пакеты и красный поток пакетов, имеющих недопустимо большую скорость.

Максимальная скорость входного трафика превышает 6 Мбайт/с., максимальная скорость желтого трафика превышает 3 Мбайт/с., а максимальная скорость зеленого трафика не достигает 3 Мбайт/с. Скорость красного трафика находится в пределах от 4,5 до 6,3 Мбайт/с.

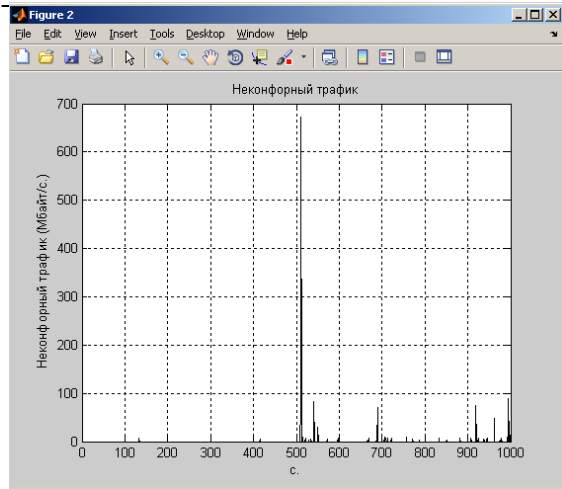


Рисунок 8 – Режим AF PNB. Неконформный трафик

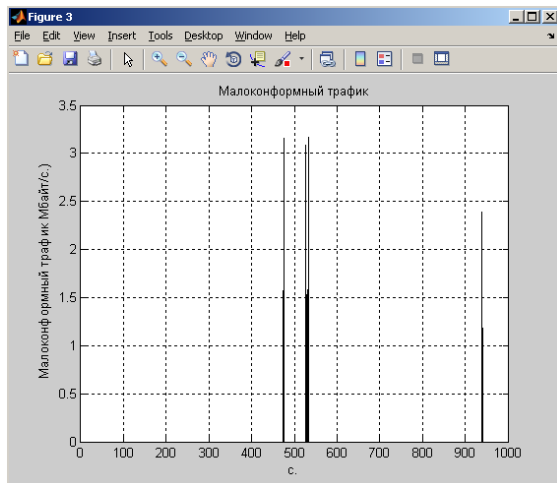


Рисунок 9 – Режим AF PNB. Малоконформный трафик

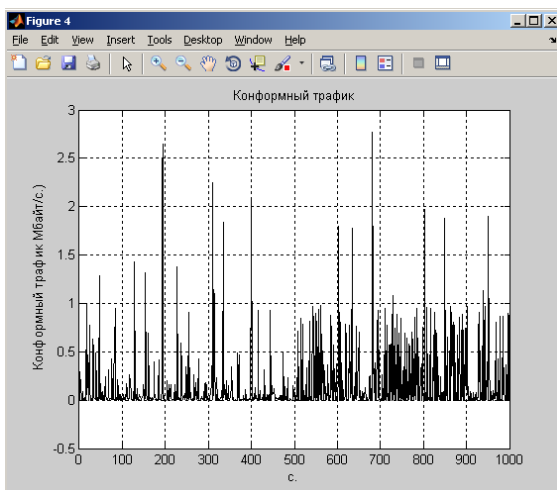


Рисунок 10 – Режим AF PNB. Конформный трафик

Входной трафик имеет показатель Херста $H_{input} = 0,419$, коэффициент пачечности

$b_{input} = 50,632$. Выходной зеленый трафик имеет показатель Херста $H_{green} = 0,463$, коэффициент пачечности $b_{green} = 27,037$. Таким образом, коэффициент пачечности зеленого трафика был снижен по отношению ко входному трафику. Коэффициенты пачечности желтого и красного трафика выросли относительно входного трафика и составили соответственно: $b_{yellow} = 229,274$, $b_{red} = 295,868$.

Во втором варианте эксперимента работа системы исследовалась для случая передачи в режиме EF PNB при $z_{c1} = 3,5$ Мбайт/с. Входной трафик не изменился. Результат эксперимента показан на рисунке 11.

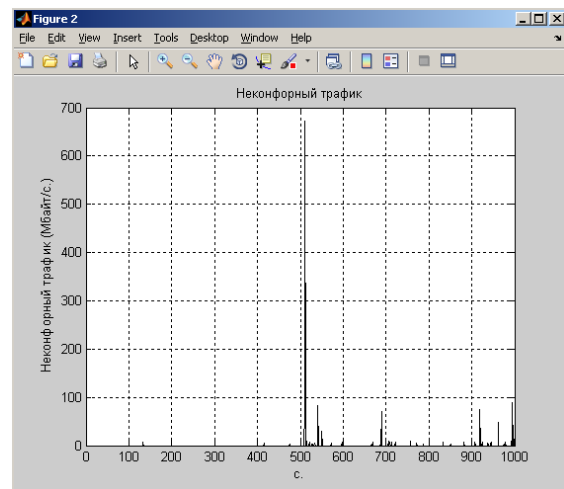


Рисунок 11 – Режим EF PNB. Неконформный трафик

Система сформировала выходной поток зеленых пакетов, который является сглаженным потоком относительно входного трафика. Был сформирован красный поток пакетов, имеющих сверхбольшую скорость. Максимальная скорость зеленого трафика достигает 3,2 Мбайт/с. Скорость красного трафика находится в пределах от 4,5 до 6,3 Мбайт/с. Выходной зеленый трафик имеет показатель Херста $H_{green} = 0,328$, коэффициент пачечности $b_{green} = 30,344$.

Таким образом, коэффициент пачечности зеленого трафика снижен по отношению ко входному трафику. Коэффициент пачечности красного трафика не изменился по сравнению с первым экспериментом: $b_{red} = 295,868$.

Выводы

В работе получены следующие результаты:

1. Разработана модель BTWIDm, которая является модификацией модели Бака-Танга-Вайзенфельда.

2. Разработана структура системы формирования трафика, выполняющая алгоритм BTW1Dm.

3. В ходе вычислительного эксперимента с помощью предложенной системы коэффициент пачечности исходного потока уменьшился на 20 единиц.

Литература

1. Park K. Self-Similar Network Traffic: An Overview. [Электронный ресурс], 2003. – Режим доступа: <http://pi.314159.ru/park1.pdf>
2. Willinger W., Taqqu M.S., Errimilli A. A bibliographical guide to self-similar traffic and performance modeling for modern high-speed networks. [Электронный ресурс], 2001. – Режим доступа: <http://linkage.rockefeller.edu/wli/reading/taqqu96.pdf>
3. Hae-Duck Joshua Jeong. Modeling of self-similar teletraffic for simulation. University of Canterbury, 2002. – 297 p.
4. Ложковський А.Г. Аналіз і синтез систем розподілу інформації в умовах мультисервісного трафіка. Автореферат дисертації. Одеса. - 2010. – 38 с.

5. Подлазов А.В. Теория самоорганизованной критичности – наука о сложности. [Электронный ресурс], 2008. – Режим доступа: <http://www.nonlin.ru/articles/podlazov/soc>
6. Городецкий А.Я., Заборовский В.С. Информатика. Фрактальные процессы в компьютерных сетях. — СПб.: СПбГТУ, 2000. – 102 с.
7. Bak P., Tang C., Wiesenfeld K. Self-organized criticality. //Physical review, Vol 38, № 1, 1988. p. 364-375.
8. Aschwanden M. Self-organized criticality in astrophysics. [Электронный ресурс], 2012. – Режим доступа: <http://ua.bookfi.org/book/1076440>
9. Олифер В.Г., Олифер Н.А. Компьютерные сети. - СПб: Питер, 2010. – 944 с.
10. C-5e Network Processor. [Электронный ресурс], 2012. – Режим доступа: http://www.freescale.com/files/netcomm/doc/data_sheet/C5ENPA1-DS.pdf
11. DieselNet Fall 2007 - AP Connectivity. [Электронный ресурс], 2007. – Режим доступа: <http://traces.cs.umass.edu/index.php/Network/Network>

Бельков Д.В., Едемская Е.Н. Система формирования трафика компьютерной сети на основе самоорганизованной критичности. Одной из парадигм синергетики является теория самоорганизованной критичности (SOC). В случае подхода SOC фрактальный сигнал является выходным сигналом нелинейной динамической системы, находящейся в критическом состоянии. Выходной сигнал – результат процесса самоорганизации. В данной работе для формирования трафика при дифференцированном обслуживании (DiffServ) предлагается система, основанная на понятии самоорганизованной критичности (Self-Organized Criticality, SOC). Разработана модель SOC BTW1Dm, являющаяся модификацией модели Бака-Танга-Вайзенфельда. Предложена структура системы формирования трафика, которая выполняет алгоритм BTW1Dm. Приведены результаты вычислительного эксперимента при формировании самоподобного трафика. Эксперимент выполнен в среде Matlab. Самоподобный входной трафик был сглажен с помощью предложенной системы. Коэффициент пачечности выходного потока уменьшился на 20 единиц относительно коэффициента пачечности входного потока.

Ключевые слова: самоподобный трафик, качество обслуживания, дифференциальное обслуживание, система формирования трафика, самоорганизованная критичность, модель Бака-Танга-Вайзенфельда

Belkov D.V., Edemskaya E.N. The traffic formed system based self-organized criticality. One of the synergetic paradigms is the theory of self-organized criticality (SOC). In case SOC approach, fractal signal is the output signal of the nonlinear dynamic system, but this system is in the critical state. The output signal is the result of the self-organization (synergetic) process.. In this article the traffic formed system based self-organized criticality (SOC) is present for the difference service (DiffServ). The SOC model BTW1Dm is design. It is the modification of the Bak–Tang–Wiesenfeld (BTW-1D) model. The system structure to run the BTW1Dm algorithm is design. The results of the calculating experiments for self-similar traffic formed is present. An experiment was executed in the Matlab environment. Output self-similar traffic was smoothed out by the present system. The burst coefficient job output stream diminished on 20 units in relation to the burst coefficient of input stream.

Keywords: self-similar traffic, quality of service, differential service, traffic formed system, Self-Organized Criticality, Bak–Tang–Wiesenfeld model.

Статья поступила в редакцию 20.11.2016

Рекомендована к публикации д-ром техн. наук В.Н. Павлышом

УДК 004.7

Статистический анализ трафика беспроводной сети

Бельков Д.В., Едемская Е.Н.

Донецкий национальный технический университет

belkov65@list.ru

Бельков Д.В., Едемская Е.Н. Статистический анализ трафика беспроводной сети. Недавние исследования трафика пакетов в сети показали, что это - самоподобный процесс и его Марковские (без последействия) модели приводят к некорректным решениям. Поэтому важной задачей является дальнейшее исследование современного трафика. Данная статья представляет результаты статистического анализа RTT-задержки беспроводной сети с GPRS-доступом на стороне отправителя и UMTS-доступом со стороны получателя.

Ключевые слова: трафик, задержка, джиттер, функция автокорреляции, спектральная плотность, параметр Херста.

Введение

В связи с тенденциями объединения различных телекоммуникационных приложений на базе универсальной сетевой инфраструктуры актуальной проблемой является разработка методов анализа и синтеза информационно-управляющих систем. Данная работа посвящена одной из важных задач развития этого направления – исследованию процессов в компьютерных сетях с беспроводным доступом.

Использование в сетях интегрированных информационных приложений и пакетной коммутации сопровождается сложными явлениями, изучение которых может быть выполнено с помощью вероятностных подходов.

Характерным для процессов пакетной передачи является обнаруженное на практике свойство фрактальности. В процессе передачи возникают большие всплески при относительно низком среднем уровне трафика. Данное явление, которое значительно влияет на потери и задержки пакетов, не учитывается в теории телетрафика. Расчет параметров системы распределения информации по классическим формулам дает некорректные, неоправданно оптимистические результаты. Алгоритмы обработки трафика, созданные для работы с простейшим потоком неэффективны для фрактальных потоков [1-6].

Появление новых свойств сетевого трафика, необходимость обеспечения высокого качества обслуживания приложений, делают актуальным статистический анализ трафика сетей.

Целью настоящей работы является исследование структуры трафика сети с беспроводным доступом, направленное на выявление его характерных особенностей. Изучается один из основных сетевых процессов

– процесс RTT-задержки, который служит для получения информации о состоянии сети методом “черного ящика”, когда через сеть пропускается последовательность пакетов, и на основании времени их прохождения до удаленного узла и обратно делаются выводы о загрузке сети.

В работе решаются следующие задачи: оцениваются плотности распределения, автокорреляционные функции (АКФ), энергетические спектры изучаемых временных рядов, определяется степень фрактальности (показатель Херста). В исследовании использовались программы Fractan и Statistika.

Используемые сигналы

Для изучения выбраны шесть реализаций сетевого трафика [7], полученные в 2007 году в университете города Наполи (Италия). Согласно лицензии данные свободно доступны для анализа. Условные обозначения, принятые в работе, показаны в таблице 1.

Таблица 1. Условные обозначения

Обозначение	Описание
TCP_d64	Ряд измерений задержки TCP-пакетов объемом 64 байт
TCP_d256	Ряд измерений задержки TCP-пакетов объемом 256 байт
TCP_d1024	Ряд измерений задержки TCP-пакетов объемом 1024 байт

Измерения проводились каждые 10 миллисекунд, получено свыше 3000 отсчетов. Отправитель имел GPRS-доступ, получатель - UMTS-доступ, скорость передачи 100 pps,

операционная система Windows32, на каждой из сторон.

Для процесса RTT-задержки на маршруте прохождения пакета можно выделить постоянную минимальную составляющую, обусловленную отсутствием очередей и переменную составляющую, возникающую из-за задержек в очередях. RTT-задержка измеряется на стороне источника и складывается из времени обработки, времени распространения, времени коммутации, времени ожидания в очереди. Изучаемые временные ряды показаны на рисунках 1-3. Нижний уровень задержки соответствует отсутствию очередей.

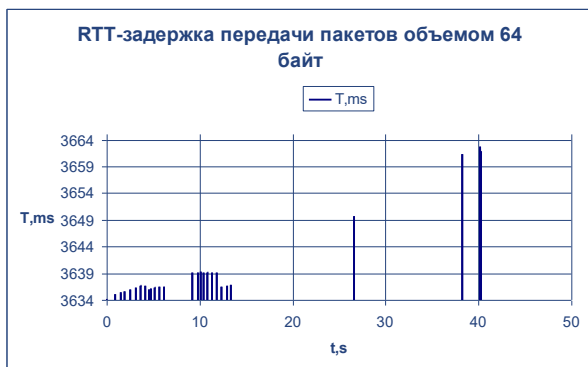


Рисунок 1 - Сигнал передачи пакетов объемом 64 байт



Рисунок 2 - Сигнал передачи пакетов объемом 256 байт

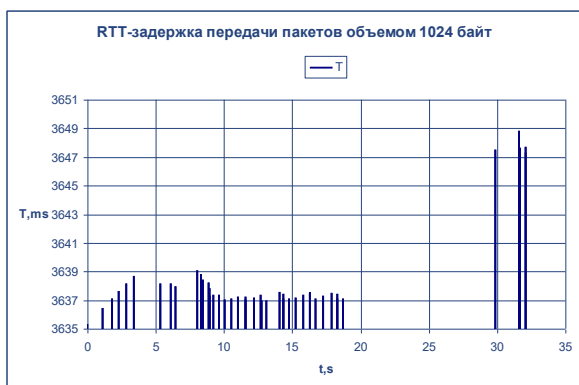


Рисунок 3 - Сигнал передачи пакетов объемом 1024 байт

Построение плотности распределения

В этом разделе выполнен анализ плотности распределения. Оценка проводится на основании гистограмм относительных частот, показанных на рисунках 4-6. Визуальный анализ позволяет сделать вывод, что исследуемые ряды не подчиняются нормальному распределению.

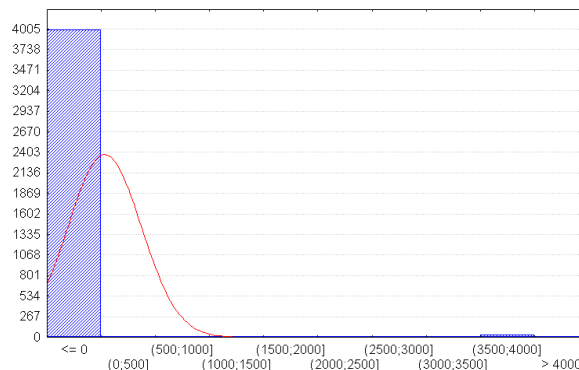


Рисунок 4 - Плотность распределения временного ряда TCP_d64

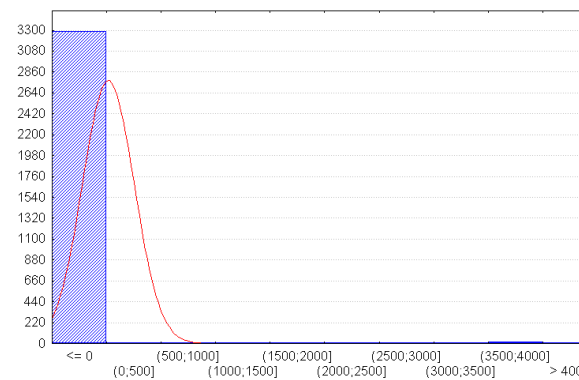


Рисунок 5 - Плотность распределения сигнала TCP_d256

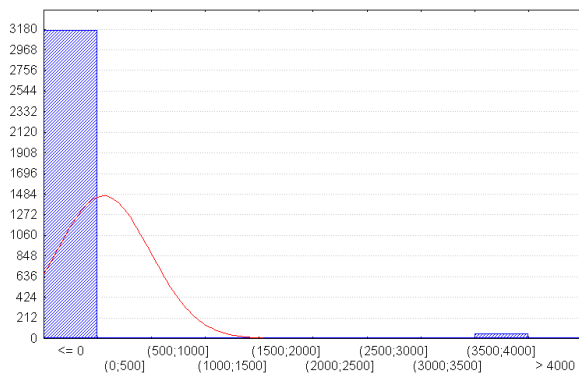


Рисунок 6 - Плотность распределения сигнала TCP_d1024

Построение автокорреляционных функций

В этом разделе необходимо определить обладают ли временные ряды медленно убывающей

зависимостью или быстро убывающей зависимостью. По определению процесс X обладает медленно убывающей зависимостью (МУЗ) [long-range dependence], если для его АКФ выполняется условие $r(k) \sim k^{-\beta} \cdot L_1(k), k \rightarrow \infty$, где $0 < \beta < 1$ и L_1 - медленно меняющаяся на бесконечности функция: $\lim_{t \rightarrow \infty} \frac{L_1(tx)}{L_1(t)} = 1$ для всех x. Процессы с

МУЗ характеризуются автокорреляционной функцией, которая убывает по степенному закону при увеличении временной задержки (лага). В отличие от процессов с МУЗ, процессы с быстро убывающей зависимостью (БУЗ) [short-range dependence] обладают экспоненциально спадающей АКФ вида $r(k) \sim e^{-k}, k \rightarrow \infty$.

Вычисления АКФ выполнены с помощью программы Fractan 4.4 по формуле

$$r(k) = \frac{\sum_{i=1}^{N-\tau} (X_i - \bar{X})(X_{i+k} - \bar{X})}{(N - \tau)\sigma^2(X)}, \quad \text{где } \bar{X} - \text{выборочное среднее ряда } X, \sigma^2(X) - \text{выборочная дисперсия ряда } X, k=0,1,\dots$$

Автокорреляционные функции исследуемых временных рядов приведены на рисунках 7-9. Их аппроксимация степенными функциями будет показана в разделе 5.

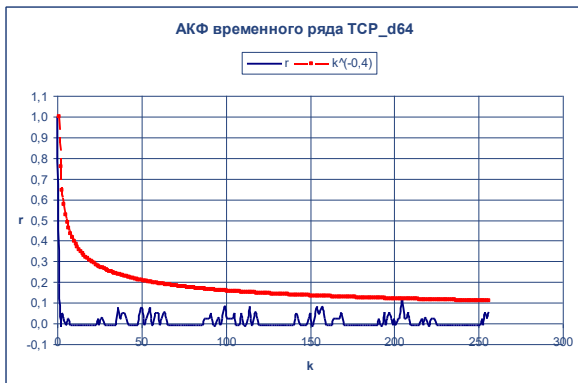


Рисунок 7 - АКФ сигнала TCP_d64

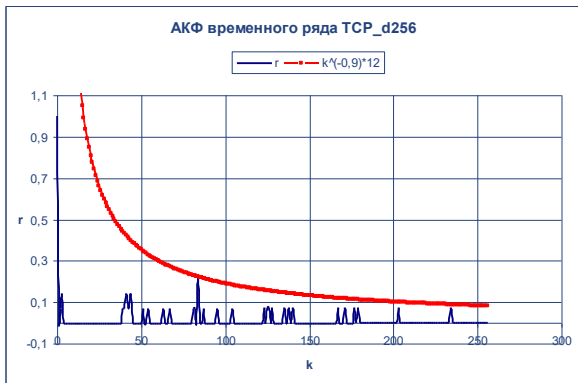


Рисунок 8 - АКФ сигнала TCP_d256

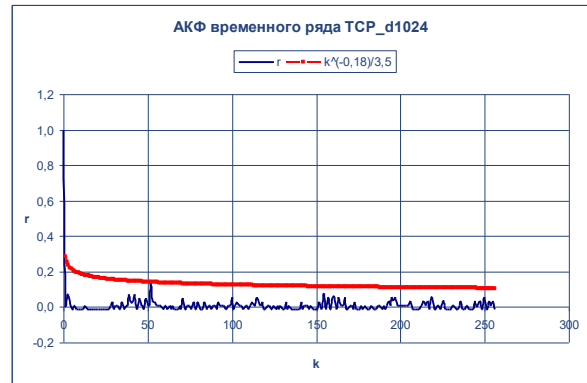


Рисунок 9 - АКФ сигнала TCP_d1024

Построение спектральной плотности сигналов

В этом разделе выполнен расчет энергетических спектров исследуемых временных рядов. Известно [8, 9], что в частотной области МУЗ приводит к степенному закону поведения спектральной плотности процесса. Процесс X обладает МУЗ, если для спектральной плотности $S(f) = \sum_k r(k) \cdot e^{ikf}$

выполняется условие $S(f) \sim f^{-b} \cdot L_2(f)$, где $f \rightarrow 0, b = \sqrt{-1}, 0 < b < 4$ и L_2 - медленно изменяющаяся в нуле функция.

Процесс с МУЗ обладает спектральной плотностью с особенностью в нуле: спектральная плотность стремится к бесконечности, по мере того как частота f стремится к нулю. Энергетические спектры реализаций представлены на рисунках 10-12.

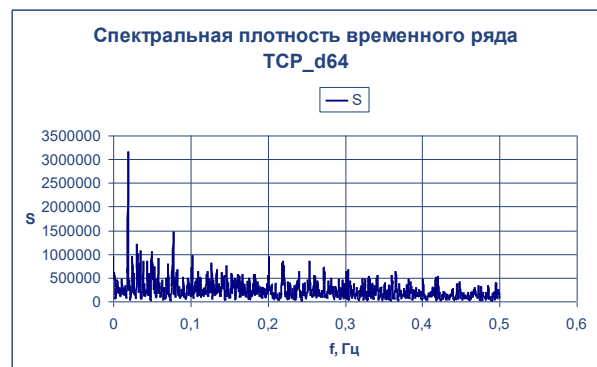


Рисунок 10 - Спектральная плотность сигнала TCP_d64

Визуальный анализ позволяет сделать вывод, что временные ряды RTT-задержки имеют на низких частотах самые мощные гармоники.



Рисунок 11 - Спектральная плотность сигнала TSP_d256

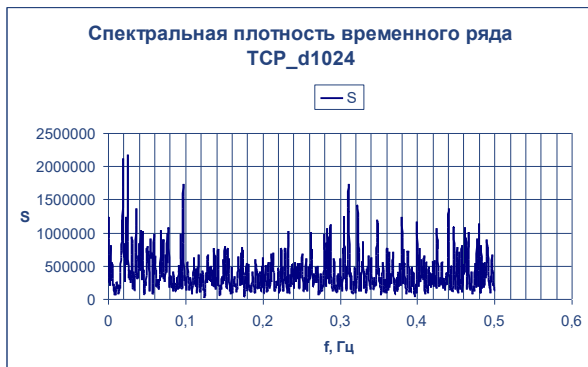


Рисунок 12 - Спектральная плотность сигнала TSP_d1024

Анализ спектральной плотности изучаемых рядов будет выполнен в следующем разделе.

Показатель Херста

В этом разделе оценивается показатель Херста H , который является мерой устойчивости статистического явления или мерой длительности долгосрочной зависимости процесса. Значение $H=0,5$ указывает на отсутствие долгосрочной зависимости. Корреляция между событиями отсутствует. Ряд является случайным, а не фрактальным. Чем ближе значение H к 1, тем выше степень устойчивости долгосрочной зависимости. При $0 \leq H < 0,5$ временной ряд является трендонеустойчивым (антиперсистентным). Он более изменчив, чем случайный ряд, поскольку состоит из частых реверсов спад-подъем. При $0,5 < H \leq 1$ ряд трендоустойчив. Тенденция его изменения может быть спрогнозирована. Показатель Херста связан с другими фрактальными параметрами формулами, показанными в таблице 2.

Таблица 2. Фрактальные параметры

Параметр	Расчетная формула
Фрактальная размерность D	$D = 2 - H$
Корреляционный параметр β	$\beta = 2(1 - H)$
Спектральный показатель b	$b = 2H - 1$

Для определения значения H в работе используется алгоритм R/S-анализа временных рядов, реализованный в программе Fractan 4.4 [10]. Алгоритм состоит в следующем. В исходной реализации Z последовательно выделяем отрезки $Z_\tau = z_1, z_2, \dots, z_\tau$, $\tau = 1000, 2000, \dots, 12000$, для каждого из которых

$$\bar{z}_\tau = \frac{\sum_{i=1}^{\tau} z_i}{\tau}$$

вычисляем текущее среднее \bar{z}_τ . Далее для каждого фиксированного Z_τ вычисляем накопленное отклонение для его отрезков

$$X_{\tau,t} = \sum_{i=1}^t (z_i - \bar{z}_\tau), t = 1, 2, \dots, \tau$$

длины t : (размах) между максимальным и минимальным накопленными отклонениями

$$R = R(\tau) = \max_{1 \leq t \leq \tau} (X_{\tau,t}) - \min_{1 \leq t \leq \tau} (X_{\tau,t})$$

и стандартное отклонение $S = S(\tau)$. Показатель Херста

$$\frac{R}{S} = \left(\frac{\tau}{2}\right)^H$$

вычисляем с помощью формулы. Логарифмируя обе части этого равенства, получаем последовательность координат точек (x_τ, y_τ) , где $x_\tau = 1000, 2000, \dots, 12000$,

$$y_\tau = H(\tau) = \frac{\lg(R(\tau)/S(\tau))}{\lg(\tau/2)}$$

Построив график зависимости $H(\tau)$ и линию тренда, определим аппроксимированное значение показателя Херста H .

На рисунке 13 показан результат R/S-анализа для процесса задержки пакетов объемом 64 байт. Показатель Херста вычислен с погрешностью $\Delta = \pm 0,2846$ и лежит в диапазоне между 0,62 и 1,19. Для определения значения H по графику спектральной плотности предлагается следующий метод исследования спектральной плотности временного ряда. Как показано на рисунке 14, строим линию L , которая огибает максимальные значения плотности S в направлении слева направо, и выполняем ее аппроксимацию степенной функцией. Показатель степени, равный 0,6, дает значение спектрального показателя b . Искомое значение $H=0,8$, вычисляется по формуле $H = (1 + b) / 2$.

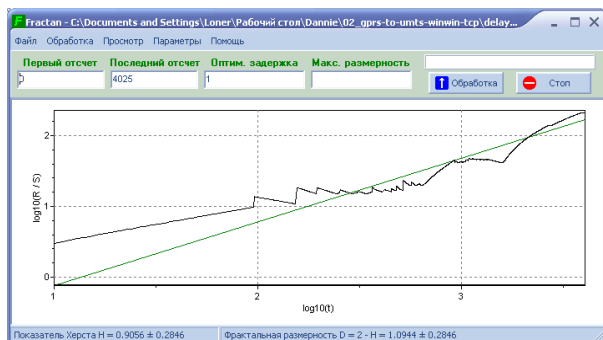


Рисунок 13 - Результат R/S-анализа сигнала TCP_d64

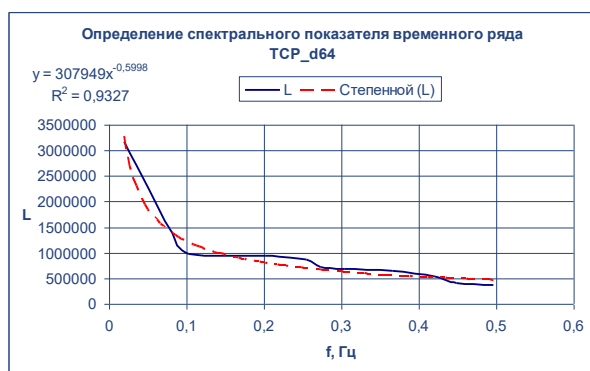


Рисунок 14 - Определение показателя b сигнала TCP_d64

Поскольку значение H лежит в диапазоне $0,5 < H \leq 1$, то ряд TCP_d64 трендоустойчив. Для аппроксимации АКФ этого ряда вычислено значение $\beta = 0,4$ по формуле $\beta = 2(1 - H)$.

На рисунке 15 показан результат R/S-анализа для процесса задержки пакетов объемом 256 байт. Показатель Херста лежит в диапазоне между 0,57 и 0,92. Определение спектрального показателя по графику спектральной плотности показано на рисунке 16. Значение $b=0,09$, $H=0,55$, $\beta = 0,9$. Ряд TCP_d256 близок к случайному.

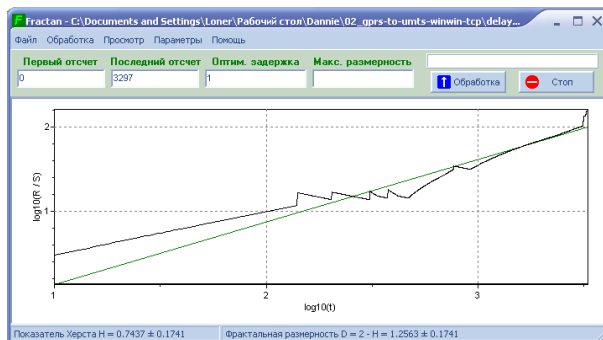


Рисунок 15 - Результат R/S-анализа сигнала TCP_d256

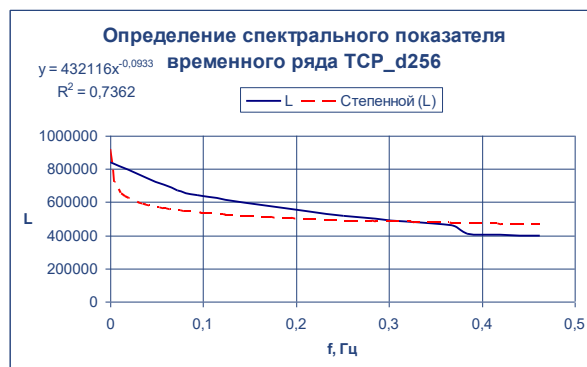


Рисунок 16 - Определение показателя b сигнала TCP_d256

На рисунке 17 показан результат R/S-анализа для процесса задержки пакетов объемом 1024 байт. Показатель Херста лежит в диапазоне между 0,59 и 1,07. Определение спектрального показателя по графику спектральной плотности показано на рисунке 18. Значение $b=0,18$, $H=0,59$. Аппроксимации АКФ этого ряда выполнена при $\beta = 0,82$.

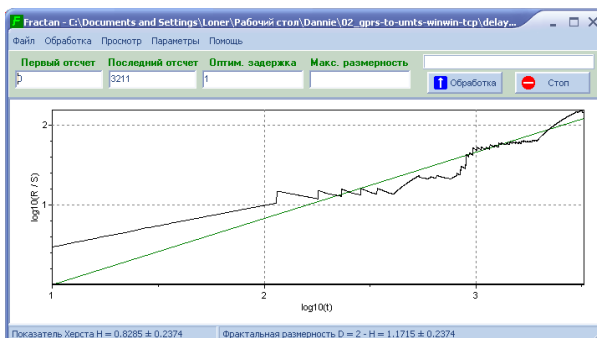


Рисунок 17 - Результат R/S-анализа сигнала TCP_d1024

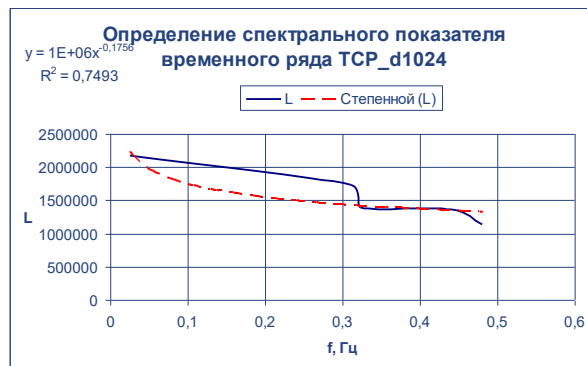


Рисунок 18 - Определение показателя b сигнала TCP_d1024

Выводы

В данной работе для сети с беспроводным доступом и реализаций [7] процесса задержки ТСП-пакетов выполнен статистический анализ. Предложен метод определения спектрального показателя по графику спектральной функции.

Получены следующие результаты:

1. Изучаемые временные ряды не подчиняются нормальному распределению.
2. Изучаемые временные ряды имеют медленно убывающую зависимость АКФ, т.е. у процессов есть долговременная память.
3. Спектральная плотность временных рядов стремится к бесконечности, по мере того как частота стремится к нулю.

Перспективным направлением дальнейших исследований является изучение структуры трафика сетей с беспроводным доступом методами нелинейной динамики.

Литература

1. Городецкий А.Я., Заборовский В.С. Информатика. Фрактальные процессы в компьютерных сетях. – СПб.: СПбГТУ, 2000. – 102 с.
2. <http://pi.314159.ru/park1.pdf> - Park K. Self-Similar Network Traffic: An Overview.
3. <http://linkage.rockefeller.edu/wli/reading/taqqu96.pdf> - Willinger W., Taqqu M.S., Errimilli A. A bibliographical guide to self-similar traffic

and performance modeling for modern high-speed networks.

4. Иванов А. В. Разработка и исследование алгоритмов прогнозирования и управления очередями в компьютерных сетях. Санкт – Петербург. – 2001. – 18 с.
5. Осин А.В. Влияние самоподобности речевого трафика на качество обслуживания в телекоммуникационных сетях. Автореферат диссертации. Москва. – 2005. – 20 с.
6. Ложковський А.Г. Аналіз і синтез систем розподілу інформації в умовах мультисервісного трафіка. Автореферат дисертації. – Одеса, 2010. – 38 с.
7. <http://www.grid.unina.it/Traffic/Traces/ttraces.php> - Network tools and traffic traces.
8. Петров В.В. Структура телетрафика и алгоритм обеспечения качества обслуживания при влиянии эффекта самоподобия. Автореферат диссертации. Москва. – 2004. – 20 с.
9. Шепухин О.И., Тенякшев А.М., Осин А.В. М. Фрактальные процессы в телекоммуникациях Москва: Радиотехника, - 2003.- 480с.
10. http://freesoft.ru/fractan_44 - Fractan 4.4. Фрактальный анализ временных рядов.

Бельков Д.В., Едемская Е.Н. Статистический анализ трафика беспроводной сети. Недавние исследования трафика пакетов в сети показали, что это - самоподобный процесс и его Марковские (без последействия) модели приводят к некорректным решениям. Поэтому важной задачей является дальнейшее исследование современного трафика. Данная статья представляет результаты статистического анализа RTT-задержки беспроводной сети с GPRS-доступом на стороне отправителя и UMTS-доступом со стороны получателя.

Ключевые слова: трафик, задержка, джиттер, функция автокорреляции, спектральная плотность, параметр Херста.

Belkov D.V., Edemskaya E.N. The statistical analysis of the traffic traces for wireless network. Recent network packet traffic investigations have demonstrated, that it is self-similar process and more conventional models, such a simple Markovian (memory-less) models have fault. Analysis the real traffic is an important task. This paper presents the results of the statistical analysis of RTT-delay traces with GPRS-access to the network at sender side and UMTS-access to the network at receiver side.

Keywords: traffic, delay, jitter, autocorrelation function, spectral density, Hurst parameter.

Статья поступила в редакцию 20.11.2016
Рекомендована к публикации д-ром техн. наук В.Н. Павлышом

УДК 004.94

Согласованная оптимизация в задачах управления эволюцией наукоемкого предприятия

Г.П. Виноградов, В.Н. Кузнецов
Тверской государственной технической университет
wgp272ng@mail.ru

Введение

В статье рассматриваются формализация и постановки задач согласованной оптимизации целевой программы развития промышленного предприятия (ЦПР). При этом применяется методология согласования интересов в экономических информационных системах [10], разработанная в Тверском государственном техническом университете совместно с Институтом проблем управления им. В.А. Трапезникова РАН. Предполагается, что отраслевые условия определяют формат и структуру ЦПР, а также традиции, субъекты и правила принятия решений.

1 Подход к согласования интересов

Методология согласования интересов в экономических информационных системах отличается от других подходов согласования экономических интересов, например от [2, 4, 6,7, 8, 11, 12], следующим:

1. Ставится и решается проблема повышения эффективности управленческих решений на основе разработки и применения формальных методов и средств обработки и отображения информации, и согласованной оптимизации.

2. Стратегия исследования. Модели теории активных систем дополняются необходимыми для описания человеко-машинных процессов согласования и оптимизации составляющими. Это

- Модели человеко-машинных процессов согласования и оптимизации на языке теории активных систем.

- Модели задач многокритериальной оптимизации с неизвестными функциями полезности, описывающими интересы пользователей активной системы, на языке эвристических процедур и теории активных систем.

- Эвристические алгоритмические модели человеко-машинных процессов согласования и оптимизации в виде алгоритмов "сводимости", т. е. в виде алгоритмических предписаний.

- Формальные алгоритмические модели человеко-машинных процессов решения экстремальных задач, описываемые с помощью алгоритмических методов оптимизации.

- Модель устойчивости коллективной стратегии пользователей активной системы на языке теории игр с противоположными интересами.

- Математические модели экстремальных задач, решаемых на каждой итерации человеко-машинных процессов, на языке исследования операций и активных систем.

- Структурные модели информационных процессов, описывающие информационную технологию согласования и оптимизации на языке структурного программирования.

- Экспериментальные модели человеко-машинных процессов согласования и оптимизации в виде компьютерных моделей и имитационных игр в активных системах.

- Модели интеллектуальной поддержки согласованной оптимизации.

Управляющей и интегрирующей средой этих моделей друг с другом и с моделями теории активных систем является концептуальная модель человеко-машинных процессов согласования и оптимизации, дающая содержательное представление о существенных свойствах этих процессов и главных связях между ними.

3. Применяется, разработанный на основе принципа согласованного управления [5], принцип согласованной оптимизации на множестве компромиссных решений. Принцип согласованной оптимизации это композиция: {коалиция согласования решений + принцип оптимального согласованного планирования состояний коалиции + человеко-машинная процедура согласования и согласованной оптимизации на множестве компромиссных решений + описание задачи согласованной оптимизации на множестве компромиссных решений + схема функционирования активной системы с коалицией согласования решений + коллективная стратегия согласованной оптимизации, обеспечивающая устойчивость коллективных решений}. За счет применения этого принципа получают решения по Нейману - Моргенштерну. Эти решения

принадлежат множеству компромиссных решений, обладающему свойствами внутренней устойчивости, состоящей в том, что полученные оптимальные согласованные решения нельзя противопоставлять друг другу, и внешней устойчивости, состоящей в возможности каждому не принадлежащему решению по Нейману - Моргенштерну и претендующему на роль более эффективного противопоставить оптимальное согласованное решение, принадлежащее этому же множеству.

Алгоритмические предписания человеко-машинных процессов согласованной оптимизации строятся на основе алгоритмов выпуклого программирования.

Для этого применяются метод Франка – Вольфа [9] и градиентные методы. Они обладают хорошей сходимостью на первых шагах поиска и устойчивы к выбору направления и шага спуска. Это обеспечивает сходимость человеко-машинных процессов и устойчивость скорости сходимости к точности определения градиентов на каждом шаге поиска. В этом случае на каждом шаге поиска можно применять или субъективные оценки направления движения в пространстве критериев или субъективные оценки в виде коэффициентов их значимости.

4. Можно осуществлять разработку на стадии проектирования автоматизированной экономической информационной системы концептуальных, информационных ее моделей, математических, алгоритмических и структурных моделей информационных процессов анализа, согласования и оптимизации для конкретной экономической системы, моделей представления и отображения информации на языке пользовательского интерфейса и языке деловой графики.

5. Методология позволяет формализовать некоторые концептуальные модели экономических и гуманитарных наук.

С помощью этой методологии были поставлены и решены задачи научных исследований согласованного управления проектами вооружения и военной техники ВМФ России [3],

согласованной оптимизации бюджетных целевых программ региона [13],

развития на множестве компромиссных решений информационного управления финансово - хозяйственной деятельности предприятий [11],

согласованной оптимизация имущественно – земельного комплекса [12],

согласования экономических интересов в анализе деятельности проектных дорожных организаций с целеустремленными агентами и др.

2 Модель согласованной оптимизации целевой программы развития промышленного предприятия

Задача согласованного управления заключается в разработке, принятии и реализации проектов ЦПР промышленного предприятия, обеспечивающих на рассматриваемом периоде времени максимальный гарантированный социально – экономический результат при условии удовлетворения интересов собственников, инвесторов, руководства и персонала предприятия.

Проект ЦПР включает в себя сведения о составе и наименовании стратегических целей, тактических задач и о их содержании; о сущности, выражении и значениях целевых показателей; о распределении финансирования по целям и задачам; о перечне и наименовании мероприятий по достижению задач, о их содержании и о перечне, наименовании и значениях факторных показателей мероприятий.

Математическое описание цели согласованного управления в социальных и экономических системах на множестве компромиссных и кооперативных решений для разработки ЦПР будет иметь следующий вид.

$$p \times \max_{y \in P(x,f)} f_0^2(x,y) + (1-p) \times \min_{y \in P(x,f)} f_0^2(x,y) \rightarrow$$

$$\rightarrow \max_{x \in X(a)}, P(x, f_0^2) = \prod P_i(x_i, y_i),$$

$$P_i(x_i, f_i^2) = \text{Arg max } f_i^2(x_i, y_i), \quad i \in I, \quad y_i \in Y_i, x \in X(a), \\ y_i \in Y_i(b_i), \quad X(a) = \prod_{i \in I} X_i(a_i), \quad Y(b) = \prod_{i \in I} Y_i(b_i),$$

$$f_0^2(x,y) = \min_{a \in A} f_0(x,y), \quad f_i^2(x_i, y_i) = \\ x \in X(a)$$

$$= \min_{b_i \in B_i} f_i(x_i, y_i).$$

$$y_i \in Y_i(b_i)$$

где: $f_0(.)$ - целевая функция, математически описывающая с помощью функции удельной ценности предпочтения собственников предприятия социально – экономических его состояний; $f_i(.)$ - целевая функция i -го целеустремленного агента, участвующего в процессе реализации ЦПР, математически описывающая его интересы; p – коэффициент сотрудничества целеустремленных агентов, участвующих в процессе реализации ЦПР; $x = (x_1, \dots, x_i, \dots, x_h)$ – составляющие проекта ЦПР предприятия; $x_i = (x_{i1}, \dots, x_{ij}, \dots, x_{im})$ – составляющие проекта ЦПР по i -ой цели; $y = (y_1, \dots, y_i, \dots, y_h)$ – реализация проекта ЦПР предприятия; $y_i = (y_{i1}, \dots, y_{ij}, \dots, y_{im})$ – реализация проекта ЦПР по i -ой цели; $X(a)$ - множество допустимых проектов ЦПР; $a = (a_1, \dots, a_i, \dots, a_h)$ – вектор факторных параметров, определяющих ограниченное

множество допустимых проектов ЦПР; $X_i(a_i)$ - множество допустимых проектов ЦПР; $a_i = (a_{i1}, \dots, a_{ij}, \dots, a_{ik})$ - вектор факторных параметров, определяющих ограниченное множество допустимых проектов ЦПР по i -ой цели; $P(x, f)$ - множество решений игры целеустремленных агентов или множество всех их локально-оптимальных состояний; $P_i(x_i, f_i)$ - множество локально-оптимальных состояний i -го целеустремленного агента; $Y_i(b_i)$ - множество реализаций проектов ЦПР по целям; $b_i = (b_{i1}, \dots, b_{ij}, \dots, b_{ik})$ - вектор факторных параметров, определяющих ограниченное множество допустимых значений факторных показателей i -го целеустремленного агента.

Представления у целеустремленных агентов, участвующих в разработке ЦПР, о согласовании интересов можно формализовать в виде множества согласованных значений факторных показателей.

$$X_c[n] = \{ x \mid f_{oi}^2(x[n], y[n]) \geq f_{oi}^2(x[n=0], y[n=0]) \} \\ + \varphi_{oi}(x[n], y[n]), \varphi_{oi}(x[n], y[n]) \geq \sum_{n \in N} \Delta_{oi}[n], \\ x[n] \in X, (y = x) \cup (y \in Y(x)), \Delta_{oi}[n] > 0, n \in N, \\ N = \{ n \mid n = 1, 2, \dots, n^* \}.$$

с использованием функций выигрыша $\varphi_{o,i}(x, y)$ элементов активной системы относительно исходного состояния, гарантируемых им центром при вступлении в коалицию. Эта функция представляет также функцию нечувствительности относительно выбора элементами активной системы (x, y) или $(x[n=0], y[n=0])$ в исходной точке поиска для $n=0$ с учетом входных барьеров, затрат на поиск и ограниченных возможностей человеко-машинных ресурсов, или относительно решений задач оптимального или оптимального согласованного планирования, определяющих гарантированные значения $\nabla_{o,i} > 0$ функции нечувствительности.

Эти выражения означают, что согласованное управление ЦПР определяет планируемые способы действия исполнителей, которые являются лучшими для собственников, руководства и персонала предприятия по их функциям предпочтения, отражающим их интересы.

Для учета применяемых целеустремленными агентами стратегий сотрудничества и компромисса запишем их функции выигрыша следующим образом.

$$\varphi_i(\cdot) = d_{io} \times f_o^2(\cdot) + \sum_j d_{ji} \times f_i^2(\cdot), \\ d_{io} + \sum_j d_{ji} = 1.$$

Коэффициенты d_{io} представляют коэффициенты сотрудничества активных элементов с хозяйствующим субъектом, а d_{ij} - коэффициенты компромисса. Если все $d_{ij} = 0$ и $d_{io} = 1$, то это стратегия сотрудничества. Если $d_{ij} > 0$ и $d_{io} < 1$, то это стратегия компромисса.

Если выполняется предположение, что в условиях возможности конфликта $(x, y) \notin X_c$ обязательно множество решений игры элементов $R(x, f) \in X_c(x)$, то тогда

$$\max \{ f_o^2(x, y) \mid x \in X, y \in Y, (x, y) \in X_c \} \geq \\ \geq \max \{ \min f_o^2(x, y) \mid y \in R(x, f) \subseteq X_c(x), x \in X \},$$

где множество решений игры элементов $R(x, f)$ образуют множества всех их локально-оптимальных состояний $R_i(x, f)$.

$$R(x, f^2) = \prod_{i \in I} R_i(x_i, f_i^2), R_i(x, f) = \\ = \text{Arg} \max_{y \in X_c} f_i^2(x, y), X_c(x) \in \{ y \mid x \in X \}.$$

Из этого соотношения следует, что применение согласованного управления ЦПР не менее эффективно по сравнению со всеми описанными и примененными на практике процедурами согласованной оптимизации.

Процесс согласованного управления ЦПР имеет следующую структуру: согласованная оптимизация проектов ЦПР, разработка и утверждение целевой программы, формирование на конкурсной основе контрактов, мониторинг реализации целевой программы. В настоящей работе рассматривается только согласованная оптимизация информационного содержания проектов по ЦПР, которое включает состав и наименование стратегических целей и тактических задач, их содержание, распределение финансирования по целям и задачам, перечень и наименование мероприятий по достижению задач, их содержание, перечень, наименование и значения факторных показателей мероприятий.

Реализация данной цели осуществляется в процессе продвижения, который состоит из следующих этапов.

1. Анализ социально – экономического состояния промышленного предприятия за прошлый и текущий периоды.

2. Определение направления движения от исходного состояния в пространстве стратегических целей, тактических задач и мероприятий по выполнению ЦПР.

3. Определение направления движения от исходного состояния в пространстве финансирования ЦПР по тактическим задачам.

4. Определение направления движения от исходного состояния в пространстве факторных показателей мероприятий по достижению

тактических задач при заданном их финансировании.

5. Оценка удовлетворительности ЦПР и анализ слабых и сильных ее сторон последовательно группой стратегического анализа, командной группой предприятия, советом по развитию, в который входят представители собственников, руководства и персонала предприятия.

6. Остановка процесса. Принятие, доработка или непринятие ЦПР.

Для оценки продвижения и ЦПР используются показатели продвижения к цели согласованного управления ЦПР, определяемые как приращения функций удельной ценности социально - экономических состояний за анализируемый, прогнозируемый или плановый периоды. Понятие удельной ценности результатов было введено в системном подходе при описании поведения целеустремленных систем [1]:

$$w = w_o(w_1, \dots, w_i, \dots, w_n) \quad - \quad \text{удельная}$$

ценность социально – экономического состояния предприятия,

$$i \in I = I_n \cap I_z \cap I_a \cap I_{om};$$

$$w_i = w_i(k_{i1}, \dots, k_{ij}, \dots, k_{im_i}) \quad - \quad \text{удельная}$$

ценность социально – экономического состояния предприятия по целям,

$$j \in J = J_n \cap J_z \cap J_a \cap J_{om};$$

$$k_{ij} = k_{ij}(x_{11k}, \dots, x_{ijk}, \dots, x_{nm_i}) \quad - \quad \text{удельная}$$

ценность социально – экономического состояния предприятия по задачам,

$$m \in M = M_n \cap M_z \cap M_a \cap M_{om};$$

$$x_{ijk} \quad - \quad \text{факторный показатель } k\text{-го}$$

мероприятия;

$$s_o, s_i, s_{ij}, s_{ijk} \quad - \quad \text{общие финансовые затраты}$$

на достижение удельных ценностей предприятия, по целям, по задачам и по мероприятиям;

$$k_{ij} = k_{ij}^s(s_{ij}), \quad x_{ijk} = x_{ijk}^s(s_{ijk}),$$

$$s_{ijk} = s_{ijk}^x(x_{ijk});$$

$$\Delta w, \Delta w_i, \Delta k_{ij} \quad - \quad \text{целевые показатели}$$

(темпы прироста) развития предприятия, стратегических целей и тактических задач;

$$\Delta x_{ijk} \quad - \quad \text{темпы прироста факторного}$$

показателя мероприятия;

$$\Delta s_o, \Delta s_i, \Delta s_{ij}, \Delta s_{ijk} \quad - \quad \text{дополнительное}$$

финансирование темпов прироста;

$$\Delta w \cong \sum_{i \in I} \sum_{j \in J} (\partial w_o / \partial w_i) \times (\partial w_i / \partial k_{ij}) \times \Delta k_{ij},$$

$$\Delta w \cong \sum_{i \in I} \sum_{j \in J} (\partial w_o / \partial w_i) \times \partial w_i / \partial k_{ij} \times (\partial k_{ij} / \partial s_{ij}) \times \Delta s_{ij},$$

$$\Delta w_i \cong \sum_{j \in J} (\partial w_i / \partial k_{ij}) \times \Delta k_{ij},$$

$$\Delta w_i \cong \sum_{j \in J} (\partial w_i / \partial k_{ij}) \times (\partial k_{ij} / \partial s_{ij}) \times \Delta s_{ij},$$

$$\Delta k_{ij} = (\partial k_{ij} / \partial s_{ij}) \times \Delta s_{ij},$$

$$\Delta k_{ij} \cong \sum_{k \in K} (\partial k_{ij} / \partial x_{ijk}) \times \Delta x_{ijk},$$

3 Выбор согласованного состава стратегических целей, тактических задач и мероприятий

Задача согласованной оптимизация состава стратегических целей и тактических задач по выполнению ЦПР предприятия описывается моделью.

$$CO(\Delta w(r_w, r_k, k^*)) \rightarrow \max, r_w = \{r_i | i = 1_w, 2_w, \dots, n_w\},$$

$$r_k = \{r_{ij} | j = 1_{ik}, 2_{ik}, \dots, m_{ik}\}, \Delta k^* = \{\Delta k_{ij}^* \} \in \Delta K^*,$$

$$\sum_{i \in I} r_i \leq n_w < n^*, \quad \sum_{j \in J_i} r_{ij} \leq m_{ik} < m_i^*$$

$$CO(\Delta w(r_w, r_k, k^*)) \cong \sum_{i \in I} \sum_{j \in J_i} CO(\partial w_o / \partial w_i) \times r_i \times$$

$$\times CO(\partial w_i / \partial k_{ij}) \times r_{ij} \times CO(\Delta k_{ij}^*),$$

$$CO(\Delta w_i(r, k^*)) \cong \sum_{j \in J} CO(\partial w_i / \partial k_{ij}) \times r_{ij} \times$$

$$\times CO(\Delta k_{ij}^*) \geq CO(\Delta_i^w).$$

$$i \in CO(I) = CO(I_c) \cap CO(I_{un}) \cap CO(I_p) \cap CO(I_n) =$$

$$= CO\{i | i = 1, 2, \dots, n^*\},$$

$$j \in CO(J_i) = CO(J_{ci}) \cap CO(J_{uni}) \cap CO(J_{pi}) \cap$$

$$\cap CO(J_{ni}) = CO\{j | j = 1_i, 2_i, \dots, m_i^*\},$$

$$r_i = \begin{cases} 1, & \text{если выбирается цель } i; \\ 0, & \text{в противном случае.} \end{cases}$$

$$r_{ij} = \begin{cases} 1, & \text{если выбирается задача } ij; \\ 0, & \text{в противном случае.} \end{cases}$$

где: $CO(\dots)$ – субъективная оценка,

$\Delta w(r_w, r_k, k^*)$ - темпы прироста социально – экономического результата предприятия за

рассматриваемый период времени, r_w - множество выбранных целей; r_k - множество выбранных задач, $\Delta k^* = \{\Delta k_{ij}^*\} \in \Delta K^*$ - вектор непрерывных числовых переменных, компонента которого определяет значение желаемого темпа прироста по задаче j цели i , $\partial w_o / \partial w_i$ - весовой коэффициент (значимости) цели i по темпу прироста социально – экономического результата предприятия, равный частной производной функции социально – экономического результата предприятия по функции результата по цели i , $\partial w_i / \partial k_{ij}$ - весовой коэффициент (значимости) задачи j по темпу прироста цели i , равный частной производной функции результата предприятия для цели i по функции результата по задаче j , $I_c, I_{ин}, I_p, I_n, I$ - множество возможных целей, желаемых собственниками, инвесторами, руководством и персоналом предприятия и их пересечение, $J_c, J_{ин}, J_p, J_{ов}, J$ - множество возможных задач, желаемых собственниками, инвесторами, руководством и персоналом предприятия и их пересечение,

Задача согласованной оптимизации состава стратегических целей, тактических задач по выполнению ЦПР предприятия заключается в определении таких целей и задач из заранее построенных множеств и таких темпов прироста результатов социально – экономического развития по задачам, которые определяют оптимальное значение темпа прироста социально – экономического состояния предприятия. Множество строится заранее по результатам научных и социологических исследований, по результатам диагностики собственников, инвесторов, руководства и персонала с применением методов практической социальной психологии, по результатам обследования предприятия и по результатам проводимых конкурсов предложений на лучшие цели и задачи.

Создается центр ответственности (ЦО) за разработку ЦПР, стратегических целей и тактических задач. При центре ответственности создается фокус – группы. Проводятся заседания по сессиям в виде совещаний и определяются в интерактивном процессе субъективные оценки множеств возможных целей

$$CO(I) = CO(I_c) \cap CO(I_{ин}) \cap CO(I_p) \cap CO(I_n)$$

собственников, инвесторов, руководства, персонала предприятия

$$CO(J_i) = CO(J_{ci}) \cap CO(J_{инi}) \cap CO(J_{pi}) \cap CO(J_{mi})$$

субъективные оценки весовых коэффициентов $CO(\partial w_o / \partial w_i)$ целевого показателя развития предприятия (темпа прироста социально –

экономического состояния предприятия) Δw_o по целевым показателям стратегических целей (темпы прироста) Δw_i ; весовых коэффициентов $CO(\partial w_i / \partial k_{ij})$ целевых показателей стратегических целей w_i по целевым показателям тактических задач (темпы прироста) Δk_{ij} , целевых значений темпов прироста социально – экономического состояния отрасли по тактическим задачам Δk_{ij}^* ; условий согласования по целевым показателям целей $CO(\Delta_i^w)$ и задач $CO(\Delta_{ij}^w)$. При этом применяется метод фокус – групп.

Фокус – группы предполагают меньшие затраты времени и средств. Предполагают анализ слов и образов, а не цифр. Применяется наблюдение и смысловая характеристика. Основным методом фокус – групп является групповая дискуссия, которая имеет следующие организационные этапы.

- определение целей и темы дискуссия;
- сбор информации (сведений, знаний, суждений, мнений, новых идей, предложений всех участников дискуссии) по обсуждаемой проблеме;
- упорядочивание, интерпретация и оценка полученной в ходе дискуссии информации;
- подведение итогов: сопоставление целей дискуссии и ее результатов.

Количество участников 4 – 5 человек. Выделяется модератор, который является лидером группы. Должна быть максимальная однородность членов группы по интересам. Это позволяет обеспечить сотрудничество и лояльность друг к другу. Целью фокус – группы является получение «субъективной информации», а не достижение единства мнений.

Если пересечение окажется пустым, то делается вывод о наличии конфликта. Деструктивный конфликт не допустим [2]. В этом случае применяются жесткие меры по реструктуризации руководства предприятия, по совершенствовании системы их стимулирования, по пересмотру кадровой политике предприятия и др. Продуктивный конфликт играет положительную роль [2] ускорения процесса принятия решений и повышения его эффективности. Но он допустим только в процессе согласованной оптимизации. При формировании окончательного решения он должен быть решен. Для этого предусмотрен внешний цикл алгоритмического предписания согласованной оптимизации целей и задач, цикл по сближению интересов. Он включает поиск ЦО способов согласования интересов: поиск общих потребностей и опасений, изменение приоритетов,

состыковка интересов, оказание взаимных услуг, осуществление не специфических компенсаций и применение санкций. После этого оказывается влияние ЦО в процессе общения на собственников, руководство, инвесторов и персонал. Проводятся беседы, совещания, консультации, дискуссии, переговоры, тренинги, коучинг и компьютерные презентации.

После этого на каждом шаге интерактивного процесса осуществляется решение задач оптимизации субъективной оценки целевого показателя $CO(\Delta w)$ и согласование субъективных оценок целевых показателей $CO(\Delta w_i)$ или методом линейного целочисленного программирования [14] или методом компьютерного моделирования, и определения информационного содержания целей и задач, значений Δk_{ij}^* .

Если существуют сильные ограничения по времени, специалистам, трудовым и финансовым затратам, по общению с собственниками, инвесторами, руководством и персоналом, то задача согласованной оптимизации упрощается. В этом случае применяются последовательный анализ вариантов [18], используемый при проектировании сложных систем, компьютерное моделирование и следующая схема согласованной оптимизации. Центром ответственности формируется первый вариант целей и задач. Осуществляется его компьютерное моделирование и презентация (оценка собственниками, инвесторами, руководством и персоналом). Определяются слабые и сильные его стороны. Формируется второй вариант целей и задач и т.д. Когда накапливается достаточное количество вариантов, то осуществляется их оценка и групповой выбор на совещаниях, советах и конференциях. Может быть создана комиссия.

После определения стратегических целей формируются центры ответственности по целям (ЦО_i).

4 Согласование объема затрат ресурсов на целевую программу

Модель согласованной оптимизации ресурсов ЦПР на множестве компромиссных и кооперативных решений собственников, инвесторов, руководства, исполнителей ЦПР, органа оценки качества и принятия ЦПР по тактическим задачам с использованием субъективных оценок центра ответственности по программе, по целям будет иметь следующий вид.

$$CO(\Delta w(\Delta s_{ij})) \cong \sum_{i \in I} \sum_{j \in J_i} CO(\partial w_o / \partial w_i) \times \\ \times CO(\partial w_i / \partial k_{ij}) \times CO(\partial k_{ij} / \partial s_{ij}) \times \Delta s_{ij} \xrightarrow{\Delta s_{ij}} \max,$$

$$c_o \geq \Delta s_o = \sum_{i \in I} \sum_{j \in J_i} \Delta s_{ij} \geq CO(\Delta s_o),$$

$$I_w = \{i | i = 1_w, 2_w, \dots, n_w\}, J_i = \{j | j = 1_{ik}, 2_{ik}, \dots, m_{ik}\}.$$

$$CO(\Delta w_i(\Delta s_{ij})) \cong \sum_{j \in J_i} CO(\partial w_i / \partial k_{ij}) \times$$

$$\times CO(\partial k_{ij} / \partial s_{ij}) \times \Delta s_{ij} \geq CO(\Delta_i^w),$$

$$CO(\Delta k_{ij}) \geq \Delta k_{ij}^*, \quad \Delta s_i = \sum_{j \in J_i} \Delta s_{ij} \geq CO(\Delta_i^s),$$

где: $CO(\dots)$ – субъективная оценка, Δs_{ij} –

переменная, определяющая размер финансирования задачи j цели i , $\Delta w(\Delta s_{ij})$ – функция, значения которой определяют темпы прироста социально – экономического результата предприятия от размеров финансирования по задачам, $\partial k_{ij} / \partial s_{ij}$ –

темпы прироста результата по задаче j цели i , приходящийся на единицу финансирования этой задачи, $\Delta w_i(\Delta s_{ij})$ – функция, значения которой

определяют темпы прироста социально – экономического результата цели i от размеров финансирования по задачам, Δs_{ij} – размер

финансирования по задаче j цели i , c_o – ограничения на финансовые ресурсы,

$$\Delta s_o = \sum_{i \in I} \sum_{j \in J_i} \Delta s_{ij} - \text{финансирование по ЦПР,}$$

Δs_o^s – желаемый размер финансирование по ЦПР,

Δk_{ij} – темпы прироста результата по задаче j , Δk_{ij}^* –

желаемый темпы прироста результата по задаче j ,

$$\Delta s_i = \sum_j \Delta s_{ij} - \text{размер финансирования по цели } i,$$

Δs_i^s – желаемый размер финансирование по цели i .

Задача согласованной оптимизации ресурсов ЦПР на множестве компромиссных и кооперативных решений заключается в определении таких размеров финансирования по задачам, которые определяют оптимальное значение темпа прироста социально – экономического состояния предприятия. Ограничения на финансовые ресурсы определяет финансовая служба предприятия с учетом фонда развития, поступающей выручки, возможностей взятия кредитов и возможностей инвестирования. Значения ограничений приближительны и

варьируются, за счет чего формируются различные варианты.

5 Модель согласованной оптимизации распределения ответственности за показатели по стратегическим целям

Модель согласованной оптимизации ЦО_i показателей мероприятий по стратегическим целям на множестве компромиссных и кооперативных решений с использованием субъективных оценок ЦО_i по целям и задачам будет иметь следующий вид.

$$CO(\Delta k_{ij}) \cong \sum_{k \in K} CO(\partial k_{ij} / \partial x_{ijk}) \times \Delta x_{ijk} \\ \xrightarrow{\Delta x_{ijk} \geq \Delta_{зн}} \max, \\ CO(\Delta k_{ij}) \geq \Delta k_{ij}^*, \Delta s_{ij} \leq \sum_{k \in K} CO(\partial k_{ij} / \partial x_{ijk}) \times \\ \times \Delta x_{ijk} \geq CO(\Delta s_{ij}^s), i \in I_w = \{i | i = 1_w, 2_w, \dots, n_w\}, \\ j \in J_{ik} = \{j | j = 1_{ik}, 2_{ik}, \dots, m_{ik}\}.$$

где: $CO(\cdot)$ – субъективная оценка, Δk_{ij} – темп прироста результата по задаче j цели i , $k \in K$ – индекс показателя мероприятия и множество их значений, Δx_{ijk} – темп прироста результата показателя мероприятия k по задаче j цели i , $\Delta x_{зн}$ – значимое значение прироста, Δs_{ij}^s – желаемый размер финансирования задачи j цели i , $\partial s_{ijk} / \partial x_{ijk}$ – финансирование мероприятия k по задаче j цели i на единицу темпа прироста результата показателя мероприятия k по задаче j цели i .

Задача согласованной оптимизации ЦО_i показателей мероприятий по стратегическим целям на множестве компромиссных и кооперативных решений центра ответственности по целям и задачам заключается в определении таких размеров темпов прироста результата по мероприятиям, которые определяют оптимальное значение темпа прироста социально – экономического состояния по задачам. При этом определяются и перечень мероприятий, для которых $\Delta x_{ijk} \geq \Delta x_{зн}$.

При ЦО_i создаются фокус – группы. Проводятся заседания по сессиям в виде совещаний и определяются в интерактивном процессе субъективные оценки весовых коэффициентов $CO(\partial w_o / \partial w_i)$ целевого показателя развития предприятия (темпы прироста социально –

экономического состояния предприятия) Δw_o по целевым показателям стратегических целей (темпы прироста) Δw_i ; весовых коэффициентов $CO(\partial w_i / \partial k_{ij})$ целевых показателей стратегических целей w_i по целевым показателям тактических задач (темпы прироста) Δk_{ij} , целевых значений темпов прироста социально – экономического состояния предприятия по тактическим задачам Δk_{ij}^* ; весовых коэффициентов $CO(\partial k_{ij} / \partial x_{ijk})$ целевых показателей тактических задач по темпам прироста факторных показателей мероприятий x_{ijk} ; субъективные гарантированные оценки $CO(\partial k_{ij} / \partial s_i)$ темпов прироста по задаче ij на единицу финансирования Δs_i ; субъективные гарантированные оценки $CO(\partial s_{ijk} / \partial x_{ijk})$ темпов прироста финансирования на единицу факторного показателя мероприятия x_{ijk} ; условий согласования по целевым показателям целей $CO(\Delta_i^w)$ и задач $CO(\Delta_{ij}^w)$, по финансированию ЦПР $CO(\Delta_o^s)$, целей $CO(\Delta_i^s)$ и задач $CO(\Delta_{ij}^s)$; субъективные оценки весовых коэффициентов критериев оценки ЦПР $CO(\partial W / \partial K_l)$. При этом применяется метод фокус – групп.

После этого на каждом шаге интерактивного процесса осуществляется решение задачи линейного программирования или проводится компьютерное моделирование и групповая оценка, и определение информационного содержания мероприятий, значений Δx_{ijk} .

6 Информационное обеспечение целевой программы

Задачи согласованной оптимизация информационного содержания ЦПР на основе оценки ее удовлетворительности в плане [1] и анализа слабых и сильных ее сторон будет иметь вид.

$$CO(\Delta w(\Delta P_l(x, z^*))) = \sum_{l \in L} CO(\partial W / \partial P_l) \times \\ \times CO(\Delta P_l(x, z^*)) \rightarrow \max, \\ x \in CO(X_c) \cap CO(X_{ин}) \cap CO(X_p), l \in CO(L_c) \cap \\ \cap CO(L_{ин}) \cap CO(L_p),$$

где: $CO(\cdot)$ – субъективная оценка, $\Delta w(\Delta P_l(x, z^*))$ – функция, определяющая значение темпа прироста социально – экономического результата предприятия от значений X, Z^* информационного содержания ЦПР и требований к ней, $\Delta P_l(x, z^*)$ – функция, определяющая значение темпа прироста критериев социально – экономического результата предприятия $\Delta P_l, l \in L$ от значений x, z^* информационного содержания ЦПР и требования к ней, x, z^* – информационное содержание ЦПР и требования к ней, $\partial W / \partial P_l$ – весовой коэффициент (значимости) по темпу прироста критерия l социально – экономического результата предприятия, равный частной производной функции социально – экономического результата региона по функции критерия l , $X_c, X_{ин}, X_p$ – возможные множества информационных содержаний ЦПР по представлениям персонала, собственников и руководства, $L_c, L_{ин}, L_p$ – возможные множества критериев ЦПР по представлениям собственников, инвесторов и руководства.

Задача согласованной оптимизации информационного содержания ЦПР на основе оценки ее удовлетворительности и анализа слабых и сильных ее сторон заключается в определении таких информационных содержаний ЦПР, которые определяют удовлетворительное для целеустремленных агентов (собственников, инвесторов и руководства) [1] значение темпа прироста социально – экономического состояния предприятия:

$w(x) = W[P(x)]$ – удельная ценность социально – экономического состояния региона или отрасли, $P(x)$ – критерии оценки ЦПР,

x, z^* – информационное содержание ЦПР и требования к ней.

Осуществляется оценка удовлетворительности ЦПР центрами ответственности предприятия и потом советом по развитию предприятия с использованием субъективных оценок

$$CO(X) = CO(X_H) \cap CO(X_Z) \cap CO(X_A),$$

$$CO(L) = CO(L_H) \cap CO(L_Z) \cap CO(L_A) \text{ и}$$

$$CO(\partial W / \partial P_l). \quad \text{Если условие}$$

удовлетворительности не выполняется, то корректируются субъективные оценки (в первую очередь условия согласования) и снова осуществляется решение задач оптимизации. Если удовлетворяется, то ЦПР сначала утверждается генеральным директором, а затем принимается

собранием собственников. При этом применяется метод фокус – групп.

Заключение

В статье приведены научные результаты, полученные при решении задачи научных исследований согласованной оптимизации целевой программы развития промышленного предприятия. Это формализация и постановки задач согласованной оптимизации. Рассмотрены также концептуальные схемы их решения. Полученные результаты были внедрены на АО «Лихославский завод светотехники».

Литература

1. Акофф Р., Эмери Ф. О целеустремленных системах. – М.: «Советское радио», 1974. – 274 с.
2. Андреева Г.М. Социальная психология. – Изд. 3. М.: Наука, 1994. – 325 с.
3. Арбузов В.А. Согласованное управление проектами ВВТ ВМФ России. // Программные продукты и системы. – 2002. - №3. – стр. 45-48.
4. Гаррет Б., Дюссож П. Стратегические альянсы: Пер. с англ. – М.: ИНФРА-М, 2002. – 332 с.
5. Бурков В.Н. Основы математической теории активных систем. – М.: «Наука», 1977. – 255 с.
6. Васильев Н.Н. Тренинг преодоления конфликтов. – СПб.: Речь. 2002. – 174 с.
7. Введение в практическую социальную психологию. Под ред. Ю.М. Жукова, Л.А. Петровской, О.В. Соловьевой. – М.: Наука, 1994. – 255 с.
8. Гераскин М.И. Согласование интересов в корпоративных структурах. – М.: ИПУ РАН. Изд-во «Анко». 2005. – 293 с.
9. Джофрион А., Дайер Дж., Файнберг А. Решение задач оптимизации при многих критериях на основе человеко-машинных процедур. – В кн.: Вопросы анализа и процедуры принятия решений. М.: Мир, 1976.
10. Кузнецов В.Н. Согласование и оптимизация в иерархических системах с активными элементами. – М.: Институт проблем управления, 1996. – 132 с.
11. Кузнецов В.Н., Пашаев Ф.А. Информационное управление финансово-хозяйственной деятельностью предприятий на множестве компромиссных решений. Задачи, модели и методы развития. Монография. – Тверь, ТГТУ, 2003. – 207 с.
12. Кузнецов В.Н., Петухов А.А., Соловьева С.А. Программная система согласованной оптимизации стратегических альянсов // Программные продукты и системы: Научный журнал. 2006, №4 – С.18–24.

13. Кузнецов В.Н., Семенов Н.А., Цибарев М.В. Программная система согласованной оптимизации бюджетных целевых программ. // Программные продукты и системы, №2, 2005 г., – С. 37-39

14. Кузнецов Ю.Н., Кузубов В.И., Волощенко А.Б. Математическое программирование. Учеб. пособие для вузов. – М.: «Выш. школа», 1976. – 352 с..

15. Майер. Э. Контроллинг как система мышления и управления. – М: Финансы и статистика, 1993. – 96 с.

16. Мескон М.Х., Альберт М., Хедоури Ф. Основы менеджмента: Пер. с англ. 2-е изд. – М.: Дело, 2004.– 800 с.

17. Миронов В.А., Спрогис Л.Л. Согласованная оптимизация имущественно-земельного комплекса // Современные сложные системы управления (СССУ/HTCS 2004): Сб. тр. науч.-прак. конф. – Тверь, ТГТУ, 2004. – С. 405-407.

18. Михалевич В.С., Волкович В.Л. Вычислительные методы исследования и проектирования сложных систем. – М.: Наука. Главная редакция физико-математической литературы, 1982. – 286 с.

Виноградов Г.П., Кузнецов В.Н. Согласованная оптимизация в задачах управления эволюцией наукоемкого предприятия. В статье рассматривается проблема согласования интересов в организационно-экономических, эффективность функционирования которых в значительной степени определяется поведением производственного персонала. Рассматриваемая в данной статье задача научных исследований отличается от известных решений тем, что в ней рассматривается согласованная оптимизация, как на множестве компромиссных решений, так и на множестве кооперативных решений. Такой подход позволил разработать методы и алгоритмы согласованного управления, заключающиеся в разработке, принятии и реализации проектов ЦПР промышленного предприятия, обеспечивающих на рассматриваемом периоде времени максимальный гарантированный социально – экономический результат при условии удовлетворения интересов собственников, инвесторов, руководства и персонала предприятия. Кооперация является необходимым элементом совместной деятельности. Алгоритмическое предписание решения дискретной задачи согласованной оптимизации строится на основе алгоритма решения многокритериальной дискретной задачи оптимального управления для функции полезности, ядром которого являются градиентные методы математического программирования.

Ключевые слова: согласованная оптимизация, целевая программа развития, активные субъекты, информационное управление.

Vinogradov G.P., Kuznecov V.N. Consistent optimization in control problems of the evolution of knowledge-intensive enterprises. The problem of coordination of interests in organizational and economic efficiency of functioning of which is largely determined by the behavior of the production staff. Considered in this article the problem of research is different from the prior art in that it deals with consistent optimization as on the set of trade-offs, as well as on a set of cooperative solutions. This approach allowed us to develop methods and algorithms for consistent management, the development, adoption and implementation of projects EDL industrial enterprise, providing the considered time period, the maximum guaranteed socio - economic results provided to satisfy the interests of owners, investors, management and staff of the enterprise. The cooperation is an essential element of joint activity. Algorithmic discrete problem solving prescription coordinated optimization is based on the algorithm for solving multicriteria discrete optimal control problem for the utility function, the core of which are the gradient methods of mathematical programming.

Keywords: consistent optimization, target of the program, active subjects, information management.

Статья поступила в редакцию 20.11.2016
Рекомендована к публикации д-ром физ.-мат. наук А.С. Миненко

УДК 004.89

Формирование баз знаний продукционного типа на основе UML-моделей

Н.О. Дородных, А.Ю. Юрин

Институт динамики и теории управления им. В.М. Матросова СО РАН (ИДСТУ СО РАН)

Иркутский национальный исследовательский технический университет (ИрНТУ)

tualatin32@mail.ru, iskander@icc.ru

Дородных Н.О., Юрин А.Ю. Формирование баз знаний продукционного типа на основе UML-моделей. В работе рассмотрено программное средство (Personal Knowledge Base Designer, PKBD) автоматизирующее разработку баз знаний продукционного типа. В качестве целевого языка представления знаний используется CLIPS (C Language Integrated Production System). Данное средство позволяет интегрироваться с CASE-средствами IBM Rational Rose и StarUML, в части импорта UML-моделей и генерации на их основе кода баз знаний в формате CLIPS. Для промежуточного представления и хранения извлекаемых логических правил (продукций) используется специальная обобщенная продукционная модель, которая позволяет абстрагироваться от особенностей определенных языков представления знаний. Средство ориентировано на непрограммирующего специалиста и обладает расширяемой архитектурой, за счет подключения программных модулей, в виде динамических библиотек, поддерживающих генерацию кода на других языках представления знаний. Тестирование разработанных баз знаний осуществляется путем подключения и запуска машин вывода. Приведены предварительные результаты тестирования (апробации) программного средства при решении различных учебных задач.

Ключевые слова: программная система, базы знаний, концептуальная модель, генерация кода, CLIPS, UML.

Введение

Сложность и трудоемкость процесса разработки экспертных систем (ЭС) обусловлена, главным образом, сложностью и трудоемкостью этапа разработки баз знаний (БЗ), который включает задачи по формализации предметных знаний и их описанию на определенном языке представления знаний (ЯПЗ) [1]. Повышение эффективности решения данных задач, путем их автоматизации, обуславливает необходимость разработки специализированных программных средств. Подобные программные средства в виде специализированных редакторов БЗ (например, Visual Expert System Designer, Expert System Designer, ES-Builder и др.), позволяют повысить эффективность процесса разработки за счет использования визуального моделирования, шаблонов представления знаний, автоматизации процесса верификации и генерации кода БЗ.

Одним из распространенных источников знаний являются концептуальные модели, создаваемые в процессе проектирования программного обеспечения, анализа и моделирования предметной области. Данные модели, представленные в форме диаграмм (DFD, IDEF0, IDEF5, UML и др.) (см. рис. 1), являясь результатом этапов идентификации и

концептуализации, в большинстве случаев не преобразовываются (или преобразование сильно ограничено) в программные коды БЗ.

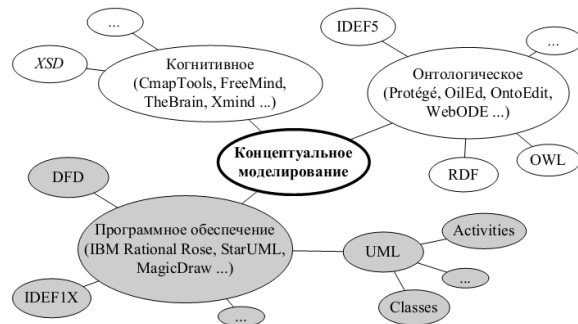


Рисунок 1 – Концепт-карта стандартов концептуального моделирования.

При наличии большого количества CASE-средств, обеспечивающих построение диаграмм (в частности, UML: IBM Rational Rose, StarUML, Enterprise Architect и др.) и синтез каркасных программных кодов на языках программирования общего назначения, не выявлено ни одного средства, генерирующего программные коды для БЗ ЭС. В тоже время в наиболее распространенных системах программирования продукционных БЗ (например, Exsys Corvid, ClipsWin и др.) отсутствует возможность

интеграции с CASE-средствами в части импорта концептуальных моделей. Что, в свою очередь, обуславливает постановку задачи создания специализированного алгоритмического и программного обеспечения, обеспечивающего анализ концептуальных моделей и синтез программных кодов БЗ производственного типа.

Таким образом, требуется разработать программную систему, обеспечивающую:

поддержку процесса проектирования БЗ производственного типа;

интеграцию с CASE-средствами в части импорта концептуальных моделей в форме диаграмм классов UML;

синтез и проверку программных кодов на CLIPS (Language Integrated Production System) [2];

поддержку RVML (Rule Visual Modelling Language) [3] – графической нотации для моделирования логических правил.

В качестве CASE-средства для интеграции предлагается использовать IBM Rational Rose [4].

Применение подобной программной системы позволит исключить ошибки, возникающие при ручном программировании, и обеспечит возможность быстрого прототипирования БЗ. Подробному описанию алгоритмического обеспечения, особенностям преобразования концептуальных моделей и краткому описанию RVML посвящены статьи [5, 6], поэтому в данной работе уделим основное внимание описанию программного средства и оценке эффективности его применения путем апробации на учебных (тестовых) примерах.

Предлагаемое решение

В результате решения поставленной задачи была разработана программная система – Personal Knowledge Base Designer, которая представляет собой специализированный редактор для разработки и тестирования производственных БЗ ЭС, в частности, для CLIPS [7].

Основными функциями редактора являются:

создание элементов производственных БЗ (шаблонов фактов и правил, а также фактов и правил) непрограммирующим пользователем, благодаря использованию либо набора подпрограмм-мастеров, либо предварительно подготовленных шаблонов фактов и правил;

поддержка авторской графической нотации – RVML [3] для визуального представления продукции. Что позволяет более наглядно отобразить причинно-следственные отношения и учесть некоторые особенности CLIPS (например, важность правил и коэффициенты уверенности);

интеграция с CASE-средствами IBM Rational Rose [4] и StarUML, в части импорта

концептуальных моделей (диаграмм классов UML), которые могут быть использованы для генерации элементов БЗ;

интеграция с CLIPS [2], в части синтеза отчуждаемого кода БЗ и его тестирования, благодаря включению в состав модулей программной системы машины вывода CLIPS;

формирование специализированных отчетов.

С целью реализации требований и функций, разработана архитектура [7], включающая следующие основные модули (см. рис. 2):

управления базами знаний – обеспечивает загрузку и сохранение БЗ в формате ЕКВ (XML-подобный формат программной системы для хранения знаний);

управления метамоделью – обеспечивает внутреннее представление производственной модели знаний, которое не зависит от определенного языка программирования БЗ, а также манипулирование (создание, редактирование, удаление) элементами этой модели;

управления модулями поддержки языков программирования – обеспечивает подключение и отключение модулей ЯПЗ, а также доступ к их функциям;

интеграции с концептуальными моделями – обеспечивает загрузку элементов из концептуальных моделей (диаграмм классов UML), построенных в CASE-средствах IBM Rational Rose и StarUML;

управления машинами вывода – обеспечивает использование машин вывода (в виде динамических библиотек) для тестирования БЗ, включая объяснение полученных результатов;

редактор RVML – обеспечивает представление элементов БЗ в виде графических примитивов, расширяющих UML;

графический пользовательский интерфейс – обеспечивает доступ к перечисленным функциям.

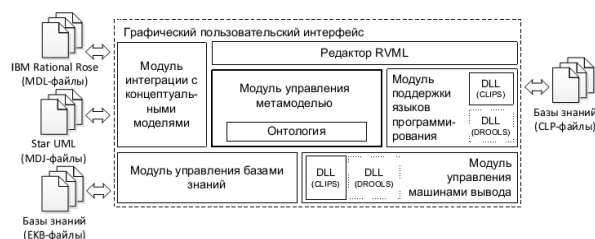


Рисунок 2 – Архитектура программной системы.

Разработанный редактор обладает простым и интуитивно понятным графическим интерфейсом, который представлен основным рабочим пространством и набором программ-мастеров, которые подставляют собой последовательность экранных форм, сегментирующих и упорядочивающих процессы ввода и редактирования элементов БЗ.

В частности, при создании шаблона факта пользователю последовательно предлагается задать: имя шаблона (используется для отображения в редакторе), описание и свойства (слоты). Подобные программы-мастера также применяются для создания и редактирования шаблонов правил.

На основе данных шаблонов, пользователь может вводить конкретные факты и правила.

Пример применения

Редактор позволяет быстро создавать и тестировать производционные БЗ, в том числе, на основе анализа и преобразование концептуальных моделей, в частности, диаграмм классов UML.

Рассмотрим простейший пример (использовался в учебном процессе) разработки ЭС для поддержки принятия решений при решении задач прогнозирования лесных пожаров. Основные понятия и отношения предметной области могут быть описаны в виде концептуальной модели (Рис. 3), анализ которой при импортировании в редактор позволяет сформировать описание шаблонов фактов и правил, используемых в дальнейшем для создания конкретных фактов и правил. При этом каждый элемент БЗ имеет графическое представление в виде RVML-схемы (Рис.4) на основе которого синтезируются программный код CLIPS.



Рисунок 3 – Пример UML-модели.

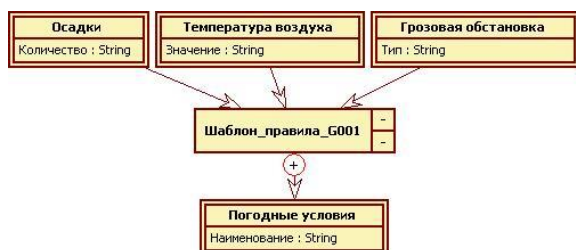


Рисунок 4 – Пример RVML-схемы.

Для проведения тестирования разработанной БЗ необходимо ввести начальные факты и активировать машину вывода (Рис. 5). В результате подобной проверки возможно ознакомиться с активированными правилами и

изменениями в рабочей памяти, например, добавленными фактами.

Проверенная экспертом БЗ может быть сохранена в формате CLIPS и использована в сторонних приложениях.

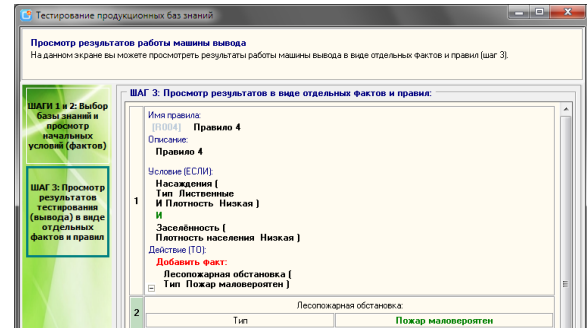


Рисунок 5 – Форма результатов проверки созданной базы знаний.

Оценка эффективности

Разработанное алгоритмическое и программное обеспечение проходило апробацию на базе Иркутского национального исследовательского технического университета (ИрНТУ). В апробации приняло участие 60 студентов Института кибернетики им. Е.И. Попова изучающих курсы «CASE-средства», «Инструментальные средства информационных технологий» и «Технологии программирования», группы АСУз-10, АСУбз-11, АСУб-12, ЭВМбзс-12. Студенты обладали знаниями в области проектирования программного обеспечения, UML и искусственного интеллекта.

Основная задача апробации состояла в том, чтобы на примере решения учебных задач оценить трудоемкость разработки БЗ в формате CLIPS с использованием разработанного редактора и без него (путем применения других средств).

Таким образом, студентам было предложено разработать статические ЭС для решения задач диагностирования или прогнозирования в определенной предметной области (в зависимости от варианта задания, Табл. 1) тремя различными способами:

C1 – путем построения диаграмм классов UML в IBM Rational Rose, с последующим их импортом в Personal Knowledge Base Designer (PKBD) и доработкой полученных структур;

C2 – путем построения диаграмм классов UML в IBM Rational Rose, с последующим их использованием в других средствах разработки БЗ CLIPS;

C3 – путем использования других средств разработки БЗ CLIPS без построения каких либо диаграмм.

В качестве «другого средства» для разработки БЗ CLIPS выбрана среда разработки –

ClipsWin [8].

При этом для обеспечения возможности неоднократного повторения процесса решения задач и их временной компактности выполнения были введены ограничения на варианты заданий:

число предметных сущностей: 5-10;

число свойств предметных сущностей: до 3;

число связей между сущностями: 5-10;

число причинно-следственных связей: 3-4;

число экземпляров причинно-следственных связей (возможных правил): 10-15.

Следует отметить, что первые 4 пункта представляют собой ограничения, налагаемые на этап концептуального моделирования (ограничения на элементы диаграммы классов UML). Последний пункт относится к этапу программирования (реализации) кода БЗ в среде разработки (ограничение на количество возможных правил в БЗ).

Таблица 1. Описание решаемых задач.

№	Предмет. сущности (ед.)	Связи (ед.)	Прич.-след. связи (ед.)	Правила (ед.)
1	6	5	3	10
2	5	6	3	10
3	8	5	3	10
4	5	8	4	11
5	8	7	3	12
6	9	5	3	10
7	5	6	3	14
8	8	7	4	14
9	6	5	3	15
10	7	10	3	12
11	5	6	3	11
12	5	6	3	12
13	7	7	3	14
14	8	5	3	11
15	7	6	3	18
16	6	8	3	14
17	6	5	3	11
18	8	7	3	12
19	7	8	3	10
20	5	7	3	12

Для оценки трудоемкости использовался временной критерий (затраты времени на выполнение отдельных этапов разработки ЭС). Оценка осуществлялась на следующих этапах [1, 9]:

Концептуализация:

формулировка базовых концепции и отношений между ними, включая: характеристику различных видов используемых данных, анализ информационных потоков и лежащих в их основе структур в предметной области в терминах, причинно-следственных связей, отношений частное/целое, постоянное/временное и т.п.;

построение концептуальной модели.

Формализация:

перевод ключевых понятий и отношений на некоторый формальный ЯПЗ;

оценка полноты и степени достоверности (неопределенности) информации и других ограничений, накладываемых на логическую интерпретацию данных, таких как зависимость от времени, надежность и полнота различных источников информации.

Реализация:

преобразование формализованных знаний в работающую программу (код БЗ).

Основным результатом этапов концептуализации и формализации являлась концептуальная модель предметной области, представленная в форме диаграмм классов UML. Основным результатом этапа реализации – синтаксически корректный программный код БЗ, проверенный на адекватность и непротиворечивость.

Результаты оценки временных затрат представлены в таблице 2, 3 и на рисунке 7. При этом выделены минимальные и максимальные процентные значения относительной разницы между С1 и С3, С1 и С2.

Отметим ряд особенностей способов:

С2: данный способ показал самые большие временные показатели, обусловленные тем, что построенные модели вручную переносились в среду разработки ClipsWin, так как у данного средства отсутствует поддержка возможности автоматической кодогенерации БЗ на основе созданных концептуальных моделей. При этом следует отметить, что другие программные средства, позволяющие синтезировать код БЗ в формате CLIPS (например, [10]), не удалось применить для этой задачи;

С3: Функциональные ограничения ClipsWin в части редактирования видимого программного кода обусловили применение дополнительного текстового редактора (Programmer's Notepad) при выполнении этапа кодирования. В частности, сначала осуществлялось описание кода БЗ во внешнем текстовом редакторе (используя возможности копирования и вставки отдельных блоков кода), а затем полученный код импортировался в ClipsWin, где осуществлялась проверка синтаксиса. На практике данная схема позволила снизить в 1,5 раза время на создание БЗ.

Анализ эффективности предлагаемого метода по временному критерию показал, что эффективность разработки баз знаний методом С1 может быть повышена, в среднем на 60.3% по сравнению с С2 и на 48.2% по сравнению с С3 за счет автоматической кодогенерации на основе визуальных моделей, что в свою очередь позволяет:

Таблица 2. Результаты оценки временных затрат.

№	Моделирование, мин.	PKBD, мин.	C1, мин.	C2, мин.	C3, мин.
1	10,89	7,2	18,09	41,29	30,4
2	8,36	7,1	15,46	32,86	24,5
3	8,58	8,3	16,88	36,46	27,88
4	9,36	5,83	15,19	26,82	17,46
5	11,25	5,52	16,77	64,41	53,16
6	10,78	4,6	15,38	43,8	33,02
7	6,6	15,82	22,42	68	61,4
8	10,95	7,56	18,51	57,23	46,28
9	7,37	7,2	14,57	54,71	47,34
10	12,58	6,6	19,18	42,7	30,12
11	8,69	5,5	14,19	38,01	29,32
12	8,36	6	14,36	45,22	36,86
13	10,64	7,42	18,06	44,31	33,67
14	10,66	10,23	20,89	50,57	39,91
15	10,01	7,56	17,57	55,5	45,49
16	10,92	8,96	19,88	49,42	38,5
17	8,14	8,36	16,5	45,59	37,45
18	11,55	18	29,55	47,43	35,88
19	11,85	5,2	17,05	43,28	31,43
20	9,12	7,44	16,56	40,95	31,83

Таблица 3. Результаты оценки временных затрат.

№	C3: ошибки програм., шт.	Относительная разница, % C1 и C3	Относительная разница, % C1 и C2
1	2	40,49	56,19
2	0	36,89	52,95
3	1	39,45	53,70
4	0	13,00	43,36
5	3	68,45	73,96
6	1	53,42	64,89
7	5	63,48	67,03
8	3	60,00	67,66
9	3	69,22	73,37
10	0	36,32	55,08
11	0	51,60	62,67
12	2	61,04	68,24
13	2	46,36	59,24
14	1	47,66	58,69
15	4	61,38	68,34
16	1	48,36	59,77
17	1	55,94	63,81
18	3	17,64	37,70
19	1	45,75	60,61
20	2	47,97	59,56
Итоговое значение разницы:	сред. относ.	48,2	60,3

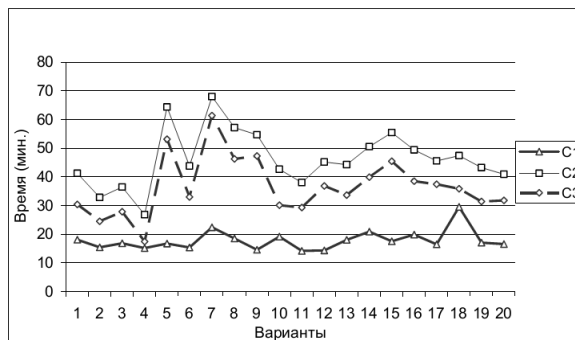


Рисунок 7 – Результаты оценки затрат времени на разработку баз знаний.

эффективно использовать результаты этапов концептуализации и формализации в форме диаграмм классов UML, рассматривая последние не как статические графические артефакты, а как основу для формирования программного кода в соответствии с идеологией модельно-управляемого подхода (Model Driven Architecture) [11];

снизить риск ошибок проектирования, за счет возможности быстрого прототипирования БЗ и получения их программного кода;

исключить ошибки программирования, за счет автоматического отображения элементов концептуальной модели в языковые конструкции CLIPS.

Заключение

Концептуальные модели остаются важным источником знаний при разработке БЗ ЭС, который в настоящее время достаточно ограниченно используется. Данные ограничения обусловлены недостатками существующего алгоритмического и программного обеспечения, что, в свою очередь, обуславливает необходимость создания специализированных программных систем в форме редакторов БЗ, способных интегрироваться с различными CASE-средствами в части импорта концептуальных моделей.

Примером подобного редактора является Personal Knowledge Base Designer [12], особенностями которого являются:

ориентация на непрограммирующего специалиста (свойство реализовано с помощью набора подпрограмм-мастеров, обеспечивающих описание знаний в виде продукции).

поддержка авторской графической нотации – Rule Visual Modeling Language (RVML) [3];

интеграция с CASE-средствами IBM Rational Rose [4] и StarUML, в части импорта и экспорта концептуальных моделей (диаграмм классов UML).

Программная система используется в учебном процессе в Иркутском национальном исследовательском техническом университете

(ИрНИТУ) при выполнении лабораторных работ по курсам «CASE-средства», «Инструментальные средства информационных систем» и «Технологии программирования».

Апробация редактора показала значительное снижение времени разработки прототипов БЗ продукционных ЭС. Кроме того, использование концептуальных моделей и Personal Knowledge Base Designer позволяет:

уменьшить риск ошибок проектирования (быстрое прототипирование позволяет проверять модели);

восстанавливать элементы БЗ на основе анализа концептуальных моделей;

исключить ошибки программирования, благодаря автоматической генерации кодов.

Необходимо отметить, что концептуальные модели, используемые в качестве основы для формирования БЗ продукционных ЭС, должны содержать описание причинно-следственных отношений. В противном случае результаты анализа моделей (осуществляемого при импорте) будут лишены содержательного смысла. Кроме того, текущая версия редактора поддерживает ограниченный набор основных конструкций языка, достаточный для автоматического синтеза исполнимых и синтаксически корректных программ. В частности, не поддерживаются: глобальные переменные, мультислоты, функции и сложные условия.

В дальнейшем планируется расширить поддержку форматов концептуальных моделей и языков программирования БЗ.

Работа выполнена при частичной финансовой поддержке РФФИ (проекты № 15-07-05641, 16-37-00122).

Литература

1. Гаврилова Т. А., Кудрявцев Д. В., Муромцев Д. И. Инженерия знаний. Модели и методы. – СПб.: Лань, 2016. – 324 с.

2. Частиков А.П., Гаврилова Т.А., Белов Д.Л. Разработка экспертных систем среда CLIPS. СПб: БХВ-Петербург, 2003. – 393 с.

3. Юрин А.Ю. Нотация для проектирования баз знаний продукционных экспертных систем // Объектные системы. – 2016. – №12. – С.48-54.

4. IBM Rational Rose Enterprise. Режим

доступа: <http://www-03.ibm.com/software/products/ru/enterprise> (дата обращения 27.10.2016).

5. Дородных Н.О., Юрин А.Ю. Использование диаграмм классов UML для формирования продукционных баз знаний // Программная инженерия. – 2015. – №4. – С.3-9.

6. Грищенко М.А., Дородных Н.О., Николайчук О.А., Юрин А.Ю. Применение модельно-управляемого подхода для создания продукционных экспертных систем и баз знаний // Искусственный интеллект и принятие решений. – 2016. – №2. – С. 16-29.

7. Юрин А.Ю., Грищенко М.А. Редактор баз знаний в формате CLIPS // Программные продукты и системы, 2012. – №4. – С. 83-87.

8. ClipsWin: CLIPS Rule Based Programming Language. Режим доступа: <https://sourceforge.net/p/clipsrules/news/2008/01/clipswin-6241/> (дата обращения 27.10.2016).

10. Джексон П. Введение в экспертные системы. Пер. с англ. – М: Вильямс, 2001. – 624 с.

11. Meditskos G., Bassiliades N. CLIPS-OWL: A framework for providing object-oriented extensional ontology queries in a production rule engine // Data & Knowledge Engineering, Vol. 70, No. 7, 2011. – p. 661-681.

12. MDA Guide rev. 2.0 // OMG Document ormsc/2014-06-01. Режим доступа: <http://www.omg.org/mda/specs.htm> (дата обращения 27.10.2016).

13. Personal Knowledge Base Designer (PKBD). Режим доступа: <http://www.knowledge-core.ru/index.php?p=pkbd> (дата обращения 27.10.2016).

Дородных Н.О., Юрин А.Ю. Формирование баз знаний продукционного типа на основе UML-моделей. В работе рассмотрено программное средство (Personal Knowledge Base Designer, PKBD) автоматизирующее разработку баз знаний продукционного типа. В качестве целевого языка представления знаний используется CLIPS (C Language Integrated Production System). Данное средство позволяет интегрироваться с CASE-средствами IBM Rational Rose и StarUML, в части импорта UML-моделей и генерации на их основе кода баз знаний в формате CLIPS. Для промежуточного представления и хранения извлекаемых логических правил (продукций) используется специальная обобщенная продукционная модель, которая позволяет абстрагироваться от особенностей определенных языков представления знаний. Средство ориентировано на непрограммирующего специалиста и обладает расширяемой архитектурой, за счет подключения программных модулей, в виде динамических библиотек, поддерживающих генерацию кода на других языках представления знаний. Тестирование разработанных баз знаний осуществляется путем подключения и запуска машин вывода. Приведены предварительные результаты тестирования (апробации) программного средства при решении различных учебных задач.

Ключевые слова: Программная система, базы знаний, концептуальная модель, генерация кода, CLIPS, UML.

Dorodnykh N.O., Yurin A.Yu. Building the rule knowledge bases on the basis of UML models. The paper considers software (Personal Knowledge Base Designer, PKBD) designed for development of rule knowledge bases. The C language production system (CLIPS) was selected as the targeted knowledge representation language. The software is integrated with CASE-tools (IBM Rational Rose and StarUML) in terms of UML-models import and code generation of knowledge bases in the CLIPS format. The special generalized rule-oriented model is used for the intermediate representation and storage of extracted logical rules. This model abstracts the specific features of knowledge representation languages. The software is focused on the non-programming specialists and has an extensible architecture that implemented the ability to connect the program modules (dynamic-link libraries) supporting the code generation in other knowledge representation languages. Testing developed knowledge bases is carried out by connecting and running the rule engines. The preliminary results of software approbation for solving a variety of learning tasks are presented.

Keywords: Software, knowledge bases, rules, conceptual model, code generation, CLIPS, UML.

Статья поступила в редакцию 08.12.2016

Рекомендована к публикации д-ром техн. наук О.А.Николайчуком

УДК 519.248

Математический анализ вероятности возникновения веерных отключений электросетей

¹Кузнецов П.А., ¹Юдин А.В., ²Масло С.В., ¹Клюковкин В.Р.
¹Рыбинский государственный авиационный технический университет
им. П.А. Соловьева
Кафедра электротехники и промышленной энергетики
²Донецкий национальный технический университет
Кафедра прикладной математики
E-mail: kuznetsovpa@inbox.ru

Кузнецов П.А., Юдин А.В., Масло С.В., Клюковкин В.Р. Математический анализ вероятности возникновения веерных отключений электросетей. В статье рассматривается вопрос возникновения и прогнозирования таких явлений, как веерные отключения и каскадные аварии, происходящие в электрических сетях. Проведен анализ веерных отключений на двух магистралях. Для углубленного анализа законов их распространения была выбрана соответствующая модель, показана тенденция к увеличению их количества. Составлен алгоритм протекания стандартного веерного отключения, причиной которого является единичное иницирующее событие. Проведен анализ потерь мощности в двух магистралях за длительные временные рамки методом скользящего окна. Представлены зависимости и подчинение каскадной аварии степенным законам.

Ключевые слова: Веерное отключение, математическая модель, анализ аварий, Minitab, статистический анализ, авария в электросети, математическое прогнозирование.

Введение

Развитие современной промышленности и рост численности населения земли приводит к тому, что с каждым днём в мире потребляется все большее количество электроэнергии. Многие сети, которые используются на данный момент для снабжения потребителей различных категорий, уже не справляются с объемом нагрузки, который на них возлагается, и требуют ремонта и модернизации. Так по данным [1] износ основных производственных и технологических активов отечественных электросетей оценивается в два раза выше, чем за рубежом. В связи с этим, все чаще как мировые, так и отечественные электросети страдают от каскадных аварий (веерных отключений), которые приводят не только к браку и материальным убыткам, но, даже к человеческим жертвам. Разработка методов и устройств диагностирования, предсказания развития аварий и их нейтрализация становятся все более актуальными задачами [2]. Основными причинами аварий в электрических сетях могут быть природные катаклизмы, сильные порывы ветра, человеческие или технические факторы, трудовые конфликты, терроризм, военные действия и т.д [3, 4]. А очень многие аварии могут быть вызваны комбинацией электрических,

компьютерных и человеческих ошибок и развиваться далее в лавинообразное веерное отключение, которые имеют огромное влияние для потребителей [5].

Идея работы

Основной задачей данной работы является математический анализ веерных отключений, происходящих в электрических сетях, на основе собранной статистики по потерям мощности. Помимо этого, часть исследования посвящена поиску первоисточника каскадных аварий и возможным методам их устранения.

Анализ последних исследований данного вопроса

Технические условия, которые могут приводить к возникновению аварийных ситуаций - это перегрузки, превышения крайних пределов уровня напряжений или частоты, нестабильность системы, отключение подстанций или источников генерации [5]. В трудах по изучению свойств электрических систем структурная сложность и высокая взаимная зависимость между компонентами системы усложняют изучение их поведения. Взаимная зависимость – это главная причина уязвимости электрических сетей и

относится к риску возникновения огромного веерного отключения.

Уязвимость системы объясняется чувствительностью системы к угрозам и опасностям [5]. Очевидно, что из-за наличия внутренних взаимосвязей, локальные перегрузки линии или выход локальных компонентов из строя могут привести к каскадному нарушению функционирования текущих узлов всей системы, что в дальнейшем приведет к её отключению. Терминологически этот процесс называется веерным отключением. Веерные отключения могут быть вызваны неспособностью электрической системы к адаптации изменению нагрузки или вследствие чрезмерного потребления электрического тока потребителем в размере, превышающем емкость сети.

Многие исследователи руководствовались поведением электрических сетей и возникновением ошибок в них в процессе передачи энергии. Сети состоят из генераторов, трансформаторов, линий электропередачи и электростанций [6]. Авария в этих системах означает, что, когда случается неисправность, по крайней мере, с одним из этих компонентов или внутри определенного элемента. Неисправность в компоненте системы может привести к серьезному веерному отключению, порождающему серьезный кризис в обществе. Однако, большинство крупнейших веерных отключений и лавинообразных аварий не были внезапными. Каскадные аварии – это главный механизм, согласно которому развиваются крупные веерные отключения. Таким образом, это – цепочка начальных явлений, которыми электрическая сеть вяло сигнализирует о реальном состоянии всех соседних звеньев [7, 8 - 10].

Каскадная неисправность определяется как последовательность зависимых неисправностей отдельных компонентов, которая постепенно ослабляет электрическую систему. Эта неисправность охватывает физические элементы системы, программное обеспечение, исполняющие функции, обслуживающий персонал и организации, которые разрабатывают, управляют и регулируют электрическую систему [11]. Каскадные отключения могут являться следствием состояния элементов системы, таких как вышедших из строя, или зависеть от характеристик передачи мощности, а также последствием ручных или автоматических команд [12]. Благодаря осложнениям полный перебор всех вариаций каскадных отключений не представляется возможным.

Исследования математической модели

Веерные отключения могут быть вызваны неспособностью электрической системы к

адаптации изменению нагрузки или вследствие чрезмерного потребления электрического тока потребителем в размере, превышающем емкость сети. В настоящее время существуют три основные концепции изучения веерного отключения в технической и научной литературе [8]. Первое учение описывает сеть и взаимодействие между её компонентами во времени, где взаимодействие выражает равенство и неравенство алгебраических систем с ограничениями в нелинейных дифференциальных уравнениях. Второе учение рассматривает поведение сети в устойчивом состоянии введением случайных флуктуаций при потреблении нагрузки, а третья концепция полагает, что электрическая сеть – это пример сложной системы, чье поведение глубоко зависит от её топологических свойств.

Модель сети на основе дифференциальных уравнений дает наиболее точное представление о физических процессах, происходящих в ней. Однако, решение, а зачастую даже и составление системы очень трудоемко, а, иногда даже и не представляется возможным. Третья модель не учитывает коммутационные процессы, постоянно влияющие на сеть. Поэтому для проведения расчетов была выбрана именно вторая теория.

Математический эксперимент

Т.к. каждая авария сопровождается потерями мощности, то для возможности анализа веерных отключений по имеющимся данным, было принято следующее упрощение. На основании второй теории о функционировании электрической сети она всегда находится в равновесном состоянии. Если в один момент времени мощность, отданная генератором (P_r) и полученная потребителем (P_n) равны, то аварии не происходит. Если же данные о генерации и потреблении мощности разнятся, то мы получаем потери (ΔP), которые представляют собой аварию.

Поскольку выбранная модель сети представляет исследуемый объект всегда находящимся в статическом состоянии и выходящим из равновесия только при коммутационных процессах, то это отразится и на описании модели аварии. Веерные отключения и каскадные аварии становятся все более частыми и разрушающими. Для подтверждения этой теории были отобраны данные [7] по двум магистралям – №1 (370 случаев) и №2 (230 случаев). Данные первой магистрали соответствуют периоду с 1990 по 2003 гг, а данные второй - с 1980 по 2005 гг. Анализ проводился в программе Minitab. На рис. 1 представлены результаты расчетов по данным потерь мощности. Ось абсцисс – это порядковый номер аварии. Они задокументированы и внесены в анализ в хронологической последовательности. По оси ординат расположены потери мощности,

которые случились в результате каскадной аварии. Для прогнозирования дальнейших вероятностей возникновения аварий были аппроксимированы линии тренда. Для данных из первой области кривая тренда соответствует степенному закону, описываемому формулой $8,2 + 0,175 \cdot x - 0,000169 \cdot x^2$. Из графика четко видно, что вероятность возникновения аварии в сети растет во времени по степенному закону и в будущем отключения будут происходить все чаще.

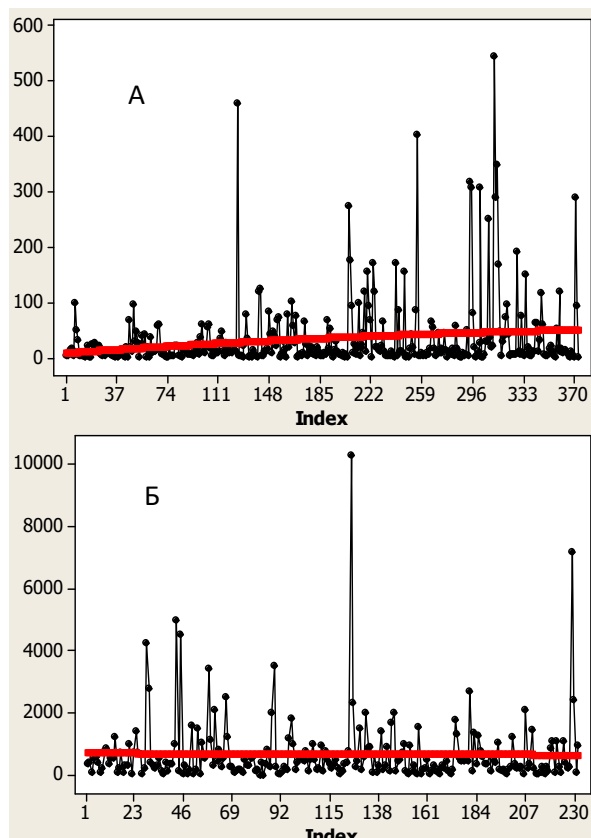


Рисунок 1 – Анализ тенденции протекания веерных отключений: а) для данных по магистрали №1; б) для данных по магистрали №2

Анализ линии № 2 показывает несколько иную картину. Линия тренда также подчиняется степенному закону, описываемому формулой $701 - 0,34 \cdot x + 0,0004 \cdot x^2$. Примечательно, что согласно данной кривой вероятность аварий будет снижаться. Это противоречие объясняется тем, что после масштабных аварий 1997-2001 годов, были приняты меры по модернизации сети и реконструкции ее мощностей.

Алгоритм протекания веерного отключения

Не смотря на значимость этого события и разрушающие последствия его протекания, исследованию каскадных аварий и веерных

отключений не уделяется много внимания. До недавнего времени вообще считалось, что основной причиной масштабных поломок электросетей являются только природные бедствия. Оказывается, что это не так. Многие исследователи ссылаются на то, что практически все масштабные аварии, которые происходили в последние десятилетия, начинались с какой-либо незначительной причины – так называемого иницирующего события. Оно сопоставимо с первым камнем, который тянет за собой лавину.

Анализ [5-7] показывает, что большинство веерных отключений, как локального, так и национального характера были следствием какой-либо единичной поломки. К примеру, веерное отключение 2003 года между США и Канадой, которое вывело из строя практически 11% нагрузки североамериканской сети и оставило без электричества более 50 млн человек, было вызвано изначально небольшой рассинхронизацией генераторов модуля 5 электростанции Eastlake. Она была вызвана наличием большого количества реактивной мощности в сети. Персонал не отреагировал вовремя и должным образом на показания системы автоматики, что и привело к каскадному отключению электроснабжения [8].

Исходя из этого, модель веерного отключения можно представить в виде алгоритма, изображенного на рис. 2. Как видно из диаграммы, в любом веерном отключении есть точка невозврата, после которого сеть будет неконтролируемо затухать, а есть и место для своевременной реакции и восстановления системы в номинальное состояние.

Очень часто эти первые сигналы не замечаются обслуживающим персоналом и даже игнорируются автоматикой. Однако, не каждое такое иницирующее событие может стать причиной веерного отключения. Возможно, что оно произойдет на локальном уровне и не вызовет цепную реакцию. Предсказание влияния таких аварий на целостность сети является важной задачей.

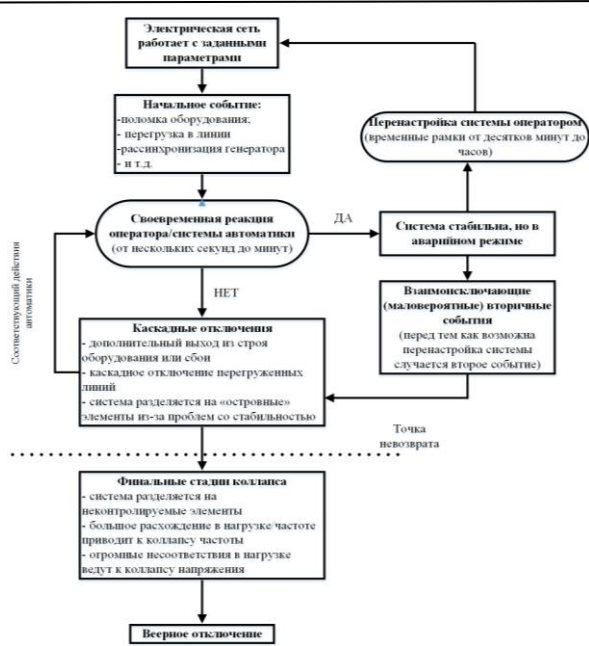


Рисунок 2 – Алгоритм протекания веерного отключения

Если течение каскадных аварий подчиняется степенным законам, то это должно подтверждаться реальными данными. Проверим это, проанализировав данные методом скользящего окна (МСО). Каждый набор данных был рассмотрен как группа временных отрезков X_t и детальное теоретическое статистическое отклонение было получено из [7].

$$X = \{X_t \quad t = 1, 2, 3, \dots, n\} \quad (1)$$

Броуновское движение может быть получено из (1):

$$Y = \{Y_t \quad t = 1, 2, 3, \dots, n\} \quad (2)$$

Где Y_t – это первообразная интеграла по времени X_t и записана в виде

$$Y_t = \sum_{i=1}^t X_i \quad (3)$$

Новый ряд $Y^{(m)}$ потом получается для ряда Y и $m=1, 2, 3, \dots, n$.

Отсюда

$$Y^{(m)} = \{Y_u^m, u = 1, 2, 3, \dots, n/m\} \quad (4)$$

Элементы этого ряда – это блоки, содержащие m элементов начального ряда Y . Это значит, что

$$Y_u^{(m)} = \{Y_{mu-m+1}, \dots, Y_{um}\} \quad (5)$$

Стандартное отклонение $\sigma_u^{(m)}$ в пределах каждого n/m блока и содержащее m элементов определяется как

$$\sigma_m^{(u)} = \sqrt{\sum_{m=1}^n \sum_{u=1}^{n/m} \frac{(Y_{mu-m+1} \dots Y_{um}) - \mu_m^{(u)}}{m}} \quad (6)$$

Где $\mu_m^{(u)}$ – это значение Броуновского движения для ряда из n/m блоков и m элементов. После всего этого среднее значение стандартного отклонения n/m блоков рассчитывается по формуле

$$\sigma_m = \frac{\sum_{u=1}^{n/m} \sigma_m^{(u)}}{n/m} \quad (7)$$

Таким образом, временной ряд X с определенной функцией автокорреляции может быть представлен как функция σ_m , имеющая фазу и шкалу степенного закона. Это - $\sigma_m \propto m^H$, где H – это экспонента Хёрста, определенная ранее. Или

$$\log(\sigma_m) = H \log(m) \quad (8)$$

Согласно [7], для $0,5 < H < 1$, ряд X_t имеет длительную временную зависимость. А для $0 < H < 0,5$ ряд имеет длительную временную антикорреляцию. Если же $H=1,0$, то ряд X_t детерминистский или выход результата предсказуем в то время как он некоррелирован, если $H=0,5$. Точный график расчета этой таблицы показан на рис. 3, 4 используя метод скользящего окна. Для второго случая результат показывает, что $H \approx 1,0$, что резко отражает наличие сильной корреляции или предсказуемой зависимости между последовательными веерными отключениями. Для первой сети точно так же прослеживается наличие длительной временной корреляции.

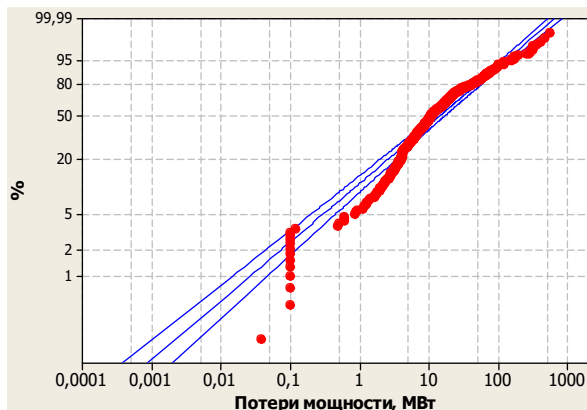


Рисунок 3 - Анализ веерных отключений первой магистрали МСО

На рис. 4 показан точечный график анализа потерь мощности во второй магистрали методом скользящего окна. Кривая представляет собой стандартное отклонение от временной зависимости для последовательных временных рядов веерных отключений. Было обнаружено, что стандартное отклонение может быть нормировано как степенной закон $\sigma_m \propto m^{0,93}$, где m представляет временное запаздывание последовательных временных рядов веерных отключений. Число $= 0,93$ – это приблизительное значение H . И, т.к. H меньше единицы, но более половины, то это значит, что существует длительная временная корреляция или зависимость между временными рядами веерных отключений.

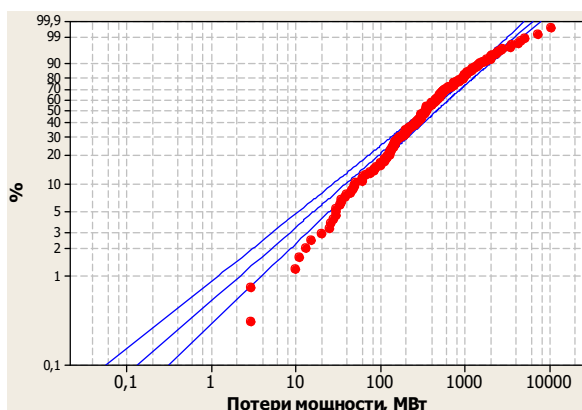


Рисунок 4 Анализ веерных отключений второй магистрали МСО

Точечный график на рис. 4 отражает анализ потерь мощности для второй сети МСО. Похоже, как и для первого случая, вторая система так же доказывает степенной закон по формуле $\sigma_m \propto m^{1,00}$ и масштабирующая экспонента $1,00$ – это аппроксимированная величина для H . Однако, в отличие от первого случая, здесь $H \approx 1,0$, что отражает детерминистскую природу временного

ряда веерного отключения или причину предшествующего события во временном ряде веерного отключения, делая его совершенно неизбежным.

Выводы

После анализа собранной статистики веерных отключений можно сделать следующие заключения.

1. Учитывая текущие состояния магистральных сетей и растущую на них нагрузку, количество веерных отключений будет только увеличиваться. Для их предотвращения необходимы защитные меры: введение прогнозирующей автоматики. Увеличение пропускной способности, эффективный менеджмент.

2. Большинство каскадных аварий происходят вследствие начального единичного инициирующего события, которое часто остается незамеченным и не анализируется. 85% крупных аварий могли бы быть предотвращены своевременной реакцией персонала на подобные единичные события.

3. Статистика каскадных аварий была обработана методом скользящего окна. Обе сети показали степенную зависимость.

Литература

1. Кобец Б. Б., Волкова И. О. Инновационное развитие электроэнергетики на базе концепции Smart Grid. — М.: ИАЦ Энергия, 2010. — 208 с.
2. Department of Energy USA. <http://www.energy.gov/statistics>.
3. World Energy Outlook 2015. International Energy Agency (IEA), Paris. — 2015. — 15 pp.
4. British Petroleum Energy Outlook 2016 Edition. British Petroleum, London – 2016. – 80pp.
5. Pouyan Pourbeik, Prabha Kundur. The anatomy of a power grid blackout – Root causes and dynamics of recent major blackouts. – Article in IEEE Magazine, Piscataway, New Jersey – 2016.
6. J. Blumschein, Y. Yelgin, M. Kereit. Blackout Prevention by Power Swing Detection and Out-of-Step Protection. – Journal of Power and Energy Engineering (JPPE), Irvine CA – 2014.
7. Friew Gebremedhin Abraha Statistics of Electric Power Blackouts: Data Analysis and Data Modelling. – Norwegian University of Science and Technology, Trondheim – 2013, 71 с.
8. North American Electric Reliability Council Technical Analysis of the August 14, 2003, Blackout: What Happened, Why, and What Did We Learn? – Report to the NERC Board of Trustees by the NERC Steering Group, New Jersey - 2004.
9. A. Farina, A. Graziano, F. Mariani, and F. Zirilli. Probabilistic analysis of failures in power transmission networks and phase transitions: Study case of a high-voltage power transmission network.

Journal of Optimization Theory and Applications, 139(1), 2008.

10. M. Vaiman, K. Bell, Y. Chen, B. Chowdhury, I. Dobson, P. Hines, M. Papic, S. Miller, and P. Zhang. Risk assessment of cascading outages: Methodologies and challenges. Power Systems, IEEE Transactions on, 27(2), 2012.

11. David L. Pepyne. Topology and cascading line outages in power grids. Journal of Science and Systems Engineering, 16(2), 2007.

12. Bei Gou, WeibiaoWu. Is the prediction of power system blackouts possible? In Power and Energy Society General Meeting – Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE,

2008.

13. Кузнецов П.А., Соленый С.В. Борьба с веерными отключениями в системах электроснабжения. – Завалишинские чтения: молодежная секция. Сб. докл. / СпБ.: ГУАП., 2016, 210 с.

14. Кузнецов П.А., Степанов О.А., Юдин А.В. Структура автоматической системы компенсации реактивной мощности промышленных предприятий. – Проблемы недропользования: сборник трудов международного форума-конкурса молодых ученых. Часть II/ Санкт-Петербургский горный университет. СпБ, 2016. 247 с.

Кузнецов П.А., Юдин А.В., Масло С.В., Клюковкин В.Р. Математический анализ вероятности возникновения веерных отключений электросетей. В статье рассматривается вопрос возникновения и прогнозирования таких явлений, как веерные отключения и каскадные аварии, происходящие в электрических сетях. Проведен анализ веерных отключений на двух магистралях. Для углубленного анализа законов их распространения была выбрана соответствующая модель, показана тенденция к увеличению их количества. Составлен алгоритм протекания стандартного веерного отключения, причиной которого является единичное инициирующее событие. Проведен анализ потерь мощности в двух магистралях за длительные временные рамки методом скользящего окна. Представлены зависимости и подчинение каскадной аварии степенным законам.

Ключевые слова: Веерное отключение, математическая модель, анализ аварий, Minitab, статистический анализ, авария в электросети.

P. Kuznetsov, A. Judin, S. Maslo, V. Kliukovkin. Mathematical analysis of appearance probability of blackouts. This article describes the problem of blackouts or cascade malfunctions which appears in electrical networks. There were analyzed blackout's data on the two power lines according to the collected information and there was chosen a model for the next profound analysis of their expansion laws. There is shown a tendency for their amount increase. There was developed an algorithm for standard blackout duration which reason can be a single initiating event. There was also held an analysis of power losses in two power grids for the long term time interval using scaled window variance method. There are shown dependencies which reveal the obedience of blackout's behavior according to the power law.

Keywords: blackout, mathematical model, outage analysis, Minitab, statistical analysis, emergency mode in power grid.

Статья поступила в редакцию 20.11.2016
Рекомендована к публикации д-ром техн. наук В.Н. Павлышом

УДК 658.8.027

Разработка имитационной модели для анализа динамики движения материалов и сырья с учетом случайных факторов

Д.И. Левченко (6 курс, каф. АСУ), В.А. Светличная, к.т.н., доц., К.К. Бабич, асс.
Донецкий национальный технический университет
devchonkadasha@yandex.ru

Левченко Д.И., Светличная В.А, Бабич К.К. Разработка имитационной модели для анализа динамики движения материалов и сырья с учетом рисков. Описаны общие факторы динамики движения сырья и материалов, влияющие на расходы предприятия. Предложена имитационная модель, учитывающая данные факторы и позволяющая определить такой объем хранимых на складах предприятия ресурсов, при котором возможно сократить затраты предприятия на хранение сырья и при этом полностью реализовать поставленный производственный план.

Ключевые слова: имитационное моделирование, расход сырья, управление поставками, сокращение запасов, риски.

Общая постановка задачи

Прежде чем готовая продукция попадет конечному потребителю, она проходит производственный цикл, начиная от сырья (полуфабриката) и заканчивая готовой продукцией. Учитывая, что ситуация на рынке носит непредсказуемый характер это отражается на производственном процессе.

Чтобы производство не остановилось из-за отсутствия запасов сырья, по причине увеличения резкого спроса или перебоев в снабжении, на предприятии создаются запасы. Если бы производственный процесс функционировал как единый конвейер, то потребности в запасах сводились к нулю. В реальной жизни этого не происходит.

Создание запасов всегда сопряжено с дополнительными финансовыми расходами.

Затраты, связанные с созданием и содержанием запасов можно разбить на несколько групп:

- отвлечение части финансовых средств на поддержание запасов;
- расходы на содержание специально оборудованных помещений (складов);
- оплата труда специального персонала;
- дополнительные налоги;
- постоянный риск порчи, наличия просроченного товара, хищения.

В свою очередь отсутствие необходимого объема запасов приводит также к расходам, которые можно определить в следующей форме потерь:

- потери от простоя производства,

- потеря от упущенной прибыли из-за отсутствия товара на складе в момент возникновения повышенного спроса,

- потери от закупки мелких партий товаров по более высоким ценам;

- потеря потенциальных покупателей и др.

Кроме финансовых расходов, связанных с формированием запасов, предприниматель должен руководствоваться конъюнктурой рынка.

По перечисленным выше причинам предприниматели, как в торговле, так и в промышленности, отдают предпочтение созданию запасов, так как в противном случае увеличиваются издержки обращения, т.е. уменьшается прибыль [1].

Постановка задачи исследования

Страховые запасы служат своего рода «аварийным» источником снабжения в тех случаях, когда на процесс производства и реализации начинают влиять случайные факторы. На практике спрос на товары удается точно спрогнозировать чрезвычайно редко. Это же относится и к точности предсказания сроков реализации заказов. Отсюда и необходимость в создании страховых товарно-материальных запасов.

Создание запасов требует дополнительных финансовых затрат. Поэтому возникает необходимость в сокращении этих финансовых затрат с помощью достижения оптимального баланса между объемом запаса с одной стороны, и финансовыми затратами с другой.

Этот баланс достигается выбором оптимального объема партий заказанных товаров или определением экономического (оптимального) размера заказа [2].

На крупных предприятиях, выпускающих большие объемы продукции различных наименований, целесообразным является использование компьютерных логистических систем.

Такая система включает в себя ряд достаточно объемных задач:

- учет выпуска продукции;
- учет продаж продукции и остатков;
- планирование производства;
- учет запасов;
- управление запасами.

Задача управления запасами проектируется для непрерывного обеспечения производства каким-либо видом материального ресурса. Реализация этой цели достигается решением следующих задач:

- учет текущего уровня запаса на складах различных уровней;
- определение размера гарантийного (страхового) запаса;
- расчет размера заказа;
- определение интервала времени между заказами.

Решению данной задачи посвящен целый ряд работ и исследований [1-7]. В основном в качестве решений предлагаются математико-экономические модели, с помощью которых по статическим выражениям рассчитываются требуемые параметры запасов и поставок. Эти модели не учитывают все случайные факторы, которые можно было бы учесть с помощью имитационной вероятностной модели. Исследование процесса динамики поставок и расхода сырья с помощью такой модели позволит оценить и учесть риски при определении модели поставок, а также определить рациональные объемы страховых запасов.

Основная часть

Для анализа динамики поступлений и расхода материалов и сырья с учетом случайных факторов, влияющих на ритмичность функционирования предприятия и, как следствие, на движение его финансовых потоков, была разработана имитационная модель.

Данная модель позволяет проследить зависимость затрат предприятия от гарантийного уровня запаса сырья на складах предприятия и на основании этой зависимости определить оптимальный уровень запасов сырья. При разработке имитационной модели были проанализированы особенности работы предприятия, а именно используемые модели

управления сырьем, особенности производства продукции, «поведение» при недостаточном количестве сырья и основные составляющие расходов предприятия.

Основная деятельность рассматриваемого предприятия связана с производством продукции, а именно пива и пивосодержащих напитков. Логично предположить, что поступление и расход сырья на этом предприятии напрямую связаны с составленным производственным графиком предприятия. При этом поступление продукции необходимо организовать так, чтобы производственный график был выполнен в полной мере и с минимальными задержками. Производственный график составляется при планировании работы предприятия на будущие периоды. При составлении производственного графика учитываются объемы продукции, которые нужно поставить по уже имеющимся договорам на поставку, а так же спрогнозированный объем продаж на планируемый период. Прогноз объемов продаж необходим, т.к. предприятием заключаются не только долгосрочные договора на поставку продукции, но и краткосрочные (сезонные), т.к. спрос на продукцию предприятия может колебаться в зависимости от времени [3].

В зависимости от объемов продукции, которые необходимо произвести, и особенностей производства составляется производственный график. Основным отличием данного производства является то, что продукция производится партиями, и нецелесообразно запускать производственные линии для слишком маленькой партии продукции, да и максимальный объем производимой в день продукции ограничен физическими возможностями оборудования и работников предприятия. Составленный производственный график служит основой для дальнейшего анализа влияния динамики движения сырья и материалов на складах предприятия на некоторые затраты предприятия (издержки).

При разработке модели были выделены следующие группы издержек:

- издержки производства – затраты на производство единицы продукции;
- издержки хранения – затраты на хранение единицы сырья;
- издержки доставки – плата за сырье и его доставку;
- издержки недостатка сырья – недополученная прибыль из-за простоя производства.

Приведенные группы издержек являются основными и общими для большого количества предприятий.

Управление запасами на рассмотренном предприятии выполняется с использованием системы управления запасами с фиксированным

объемом поставок. В этой системе акцент делается на размере заказа, он является основополагающим параметром и определяется в первую очередь. Размер заказа строго зафиксирован и не меняется ни при каких условиях. Заказ подается в момент, когда текущий запас достигает порогового уровня. Если поступивший заказ не пополняет запас до порогового уровня, то новый заказ производится в день поступления заказа.

Существует довольно много методов расчета оптимального размера заказа. Одним из наиболее распространенных методов является использование формулы Вилсона [6]. Однако каким бы ни был размер поставки сырья, его необходимо где-то хранить до потребности в нем. Для хранения сырья недостаточно арендовать помещение под склад, необходимо обеспечить и важно соблюдать в этом помещении определенные условия для надлежащего хранения сырья. Т.к. его повреждения или порча может привести к дополнительным затратам (в основном на заказ новой партии сырья).

Для расчета стоимости хранения сырья, специалисты советуют использовать следующую формулу [7]:

$$Z_{\text{хран.}} = ST_{\text{хран. уд.}} * T_{\text{оборот. запасов}} * V_{\text{исп. сырья}} \quad (1)$$

где $Z_{\text{хран.}}$ – затраты на хранение данного сырья;

$ST_{\text{хран. уд.}}$ – удельная стоимость хранения – это тот объем затрат, который осуществляется на единицу склада в единицу времени. Как правило, за единицу времени принимается 1 день. Единица измерения этого параметра – рубль на ед. складской мощности в один день.

$T_{\text{оборот. запасов}}$ – период оборота запасов – это промежуток времени с момента фактического прибытия партии на склад до момента использования сырья из этой партии, обычно измеряется в днях.

$V_{\text{исп. сырья}}$ – объем сырья, перемещенного со склада в единицах складской мощности.

Как было сказано выше, заказ на поставку сырья подается в момент, когда текущий запас достигает порогового уровня. Пороговый уровень запаса определяется следующим образом:

$$Z_{\text{пор}} = Z_{\text{гар}} + Z_{\text{ож.д.}}, \quad (2)$$

где $Z_{\text{гар}}$ – гарантийный уровень сырья, т.е. страховой запас сырья, который должен всегда быть в наличии на складе на случай задержки поставки;

$Z_{\text{ож.д.}}$ – ожидаемый объем потребления сырья за время поставки заказа.

Ожидаемый объем потребления сырья за время поставки заказа равен произведению ожидаемого объема потребления сырья в день и времени доставки. Т.к. объемы потребления сырья в день колеблются мало, ожидаемый расход сырья в день можно рассчитать как среднее арифметическое объемов потребления сырья в день за планируемый период.

Т.к. основной целью использования имитационной модели является сокращение затрат на хранение продукции без ущерба для прибыли предприятия и его авторитета перед покупателем, во время работы с моделью изменяется значение гарантийного запаса сырья, а пороговый его уровень определяется по выше приведенной формуле.

При моделировании работы предприятия акцент делается на текущем уровне запаса, суммарных затратах предприятия и выполнении составленного производственного графика. Период времени, за который моделируется работа предприятия, определяется пользователем. С учетом особенностей работы предприятия этот период колеблется от двух недель до месяца. Также пользователем определяется максимальное значение гарантийного запаса, объем переходящего запаса, минимальный и максимальный объем производства в день, объем поставки сырья и сроки его доставки. Т.к. доставка сырья может быть задержана, указывается не только время доставки продукции в срок, но и количество дней возможной ее задержки, а также вероятность задержки поставки. Вероятность задержки поставки определяется экспертом на основании опыта сотрудничества с данным поставщиком, либо отзывах о нем. Блок-схема работы модели в общем виде представлена на рисунке 1.

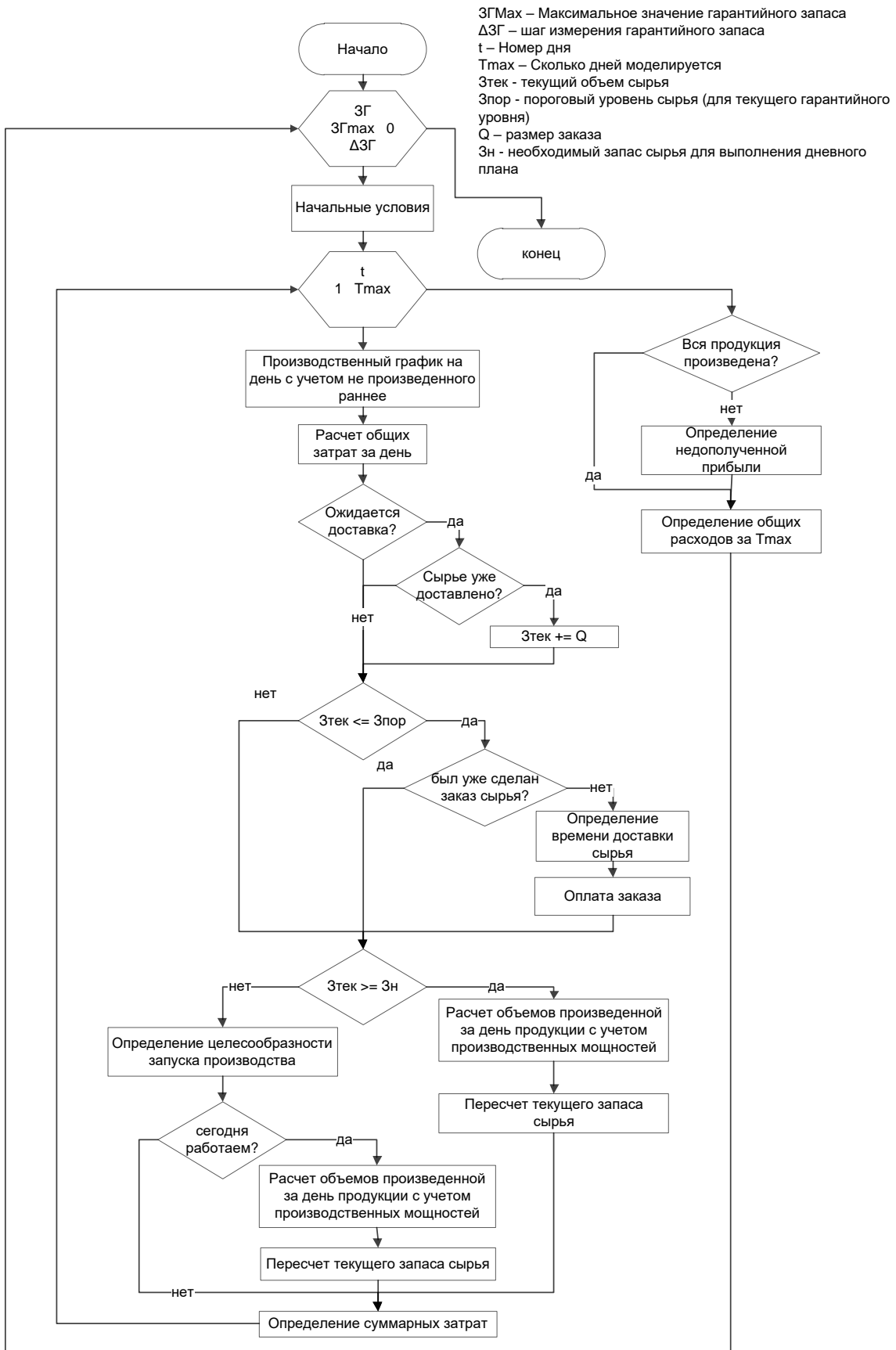


Рисунок 1 – Блок-схема работы имитационной модели

При определении начальных условий выполняется расчет порогового уровня запаса ($Z_{пор}$) для текущего значения гарантийного запаса ($ZГ$), устанавливается текущий объем запаса сырья ($Z_{тек}$), равный объему переходящего запаса. Как видно на блок-схеме, во время работы модели основными являются 2 цикла: цикл по уровню гарантийного запаса и цикл по дням (для оценки деятельности предприятия за определенный период времени). Ежедневно на предприятии в начале рабочего дня определяется производственный план (блок «производственный график на день с учетом непроизведенного ранее»). Делается это для того, чтобы учесть возможное отставание от производственного плана в предыдущие дни (в основном из-за нехватки сырья). В этом блоке к плановому объему производства прибавляется объем непроизведенной ранее продукции и определяется суммарный объем сырья, необходимый для выполнения нового дневного плана. Далее выполняется расчет общих затрат за день. В состав этих затрат входит заработная плата сотрудникам (среднее значение за день), издержки на хозяйственные нужды (электричество, вода и др.), а также стоимость хранения сырья.

Поскольку поставки сырья выполняются в большинстве случаев в начале рабочего дня, далее выполняется проверка, не ожидается ли «сегодня» новой поставки. Если сырье доставлено, текущий запас сырья ($Z_{тек}$) увеличивается на размер поставки (Q). Также в начале рабочего дня определяется достаточно ли сырья на предприятии для дальнейшей его работы. Эта проверка выполняется в два этапа. Первый этап – долгосрочный, в нем сравниваются значения текущего запаса сырья и его порогового уровня. Если текущее значение запаса ниже или равно пороговому уровню, и заказ на поставку продукции еще не был сделан, то оформляется заказ на поставку продукции и сразу же оплачивается. Время получения заказа определяется на основании времени поставки товара, времени его возможной задержки и вероятности задержки товара, которые вводятся пользователем. Таким образом, модель учитывает и случайные факторы, влияющие на работу предприятия.

На втором этапе проверка выполняется для определения достаточности текущего запаса для выполнения намеченного производственного плана. На этом этапе работы модели очень важно

учитывать особенности производства, а именно минимальный и максимальный уровни производимой за день продукции. Если объемов сырья недостаточно для выполнения производственного плана, проверяется целесообразность запуска производственных линий, т.е. сравнивается значение объема выпуска по производственному плану со значением минимального объема производства за день. Если объем производства по плану ниже, производство продукции откладывается на следующие периоды времени. Блок схема расчета произведенного объема продукции представлена на рисунке 2.

Если объема сырья достаточно для производства продукции, то «производство» выполняется по производственному плану, но не более максимально возможного значения в день, и выполняется пересчет текущего запаса сырья.

Блок схема расчета произведенного объема продукции представлена на рисунке 3.

После того, как выполнено моделирование работы предприятия на указанный пользователем период, необходимо проверить был ли выполнен в полной мере намеченный производственный план. Если нет, необходимо определить объем потерь от дефицита продукции, который будет следствием полученной в результате моделирования работы предприятия [5]:

$$C_{потерь} = V_{дефицита} + R_{продаж} + T_{дефицита} \cdot (3)$$

где $C_{потерь}$ – сумма потерь от дефицита товара
 $V_{дефицита}$ – объем дефицита товара
 $R_{продаж}$ – рентабельность продажи товара
 $T_{дефицита}$ – период дефицита товара
 Т.к. в момент, на который выполняется расчет потерь от дефицита работа модели для текущего гарантийного уровня запаса ($ZГ$) считается оконченной, время дефицита товара равно единице, т.е. учитываются потери от дефицита товара только за день.

После определения величины потерь от дефицита товара, в модели суммируются все затраты, связанные с текущим гарантийным уровнем запаса, т.е. затраты на заработную плату сотрудникам, хозяйственные нужды, за поставку сырья и его хранение, потери от недостаточного уровня запаса сырья. Полученные суммарные затраты для текущего уровня гарантийного запаса запоминаются и выполняются аналогичные расчеты для следующего уровня гарантированного запаса.

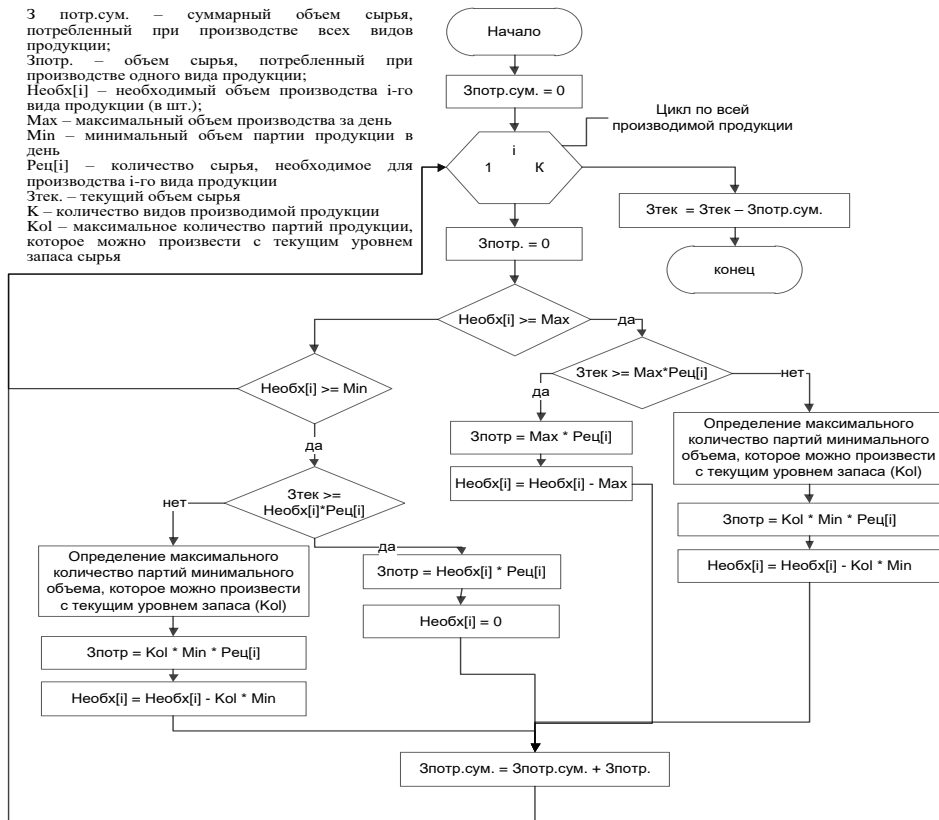


Рисунок 2 - Блок схема определения произведенного объема продукции при недостаточном текущем объеме запасов сырья

З потр.сум. – суммарный объем сырья, потребленный при производстве всех видов продукции;
Зпотр. – объем сырья, потребленный при производстве одного вида продукции;
Необх[i] – необходимый объем производства i-го вида продукции (в шт.);
Мах – максимальный объем производства за день;
Мин – минимальный объем партии продукции в день;
Рец[i] – количество сырья, необходимое для производства i-го вида продукции;
Зтек. – текущий объем сырья;
К – количество видов производимой продукции

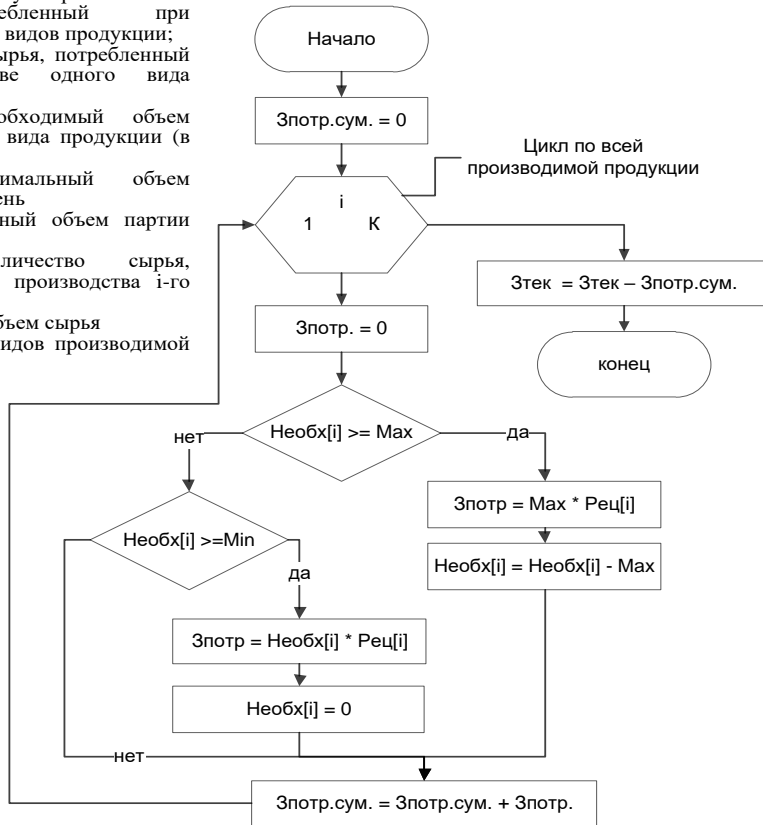


Рисунок 3 - Блок схема определения произведенного объема продукции при достаточном текущем объеме запасов сырья

Результатом работы имитационной модели является история расходов сырья и его поставок, график расхода сырья и график зависимости общих расходов от гарантийного уровня запасов. Окно программы с результатами работы имитационной модели представлено на рисунке 4. График расхода сырья отображается для гарантийного уровня запаса, при котором затраты были минимальными.

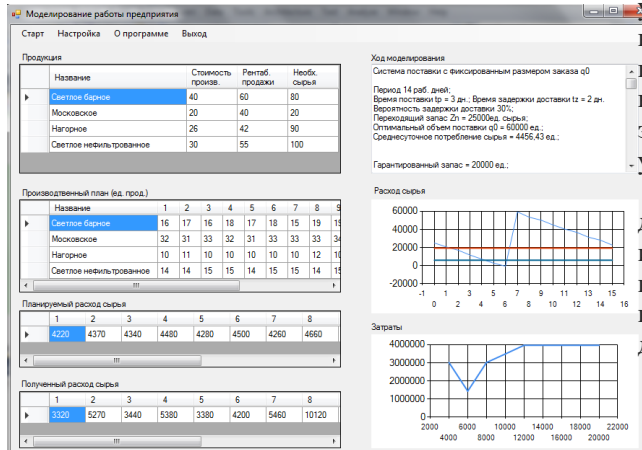


Рисунок 4 – Результат работы имитационной модели

Основным результатом работы является зависимость общих расходов от гарантийного уровня запасов (рисунок 5). На этом графике ниспадающая часть отображает затраты, по большей части состоящие из затрат на хранение сырья, текущих затрат (зарплатная плата и хозяйственные затраты) и издержек недостатка сырья. Восходящая же часть отображает затраты, в большей степени состоящие из затрат на хранение сырья и производство новой продукции.

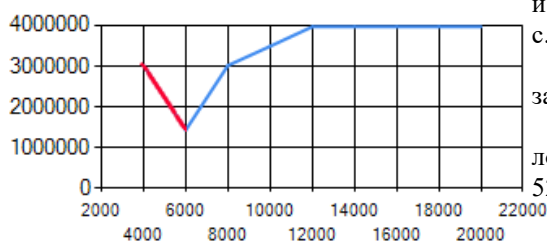


Рисунок 5 – График зависимости объемов затрат от гарантийного уровня запаса сырья

На графике (рисунок 5) видно, что для данного производственного плана оптимальным уровнем гарантийного запаса является значение близкое к 6000 ед., т.к. в этой точке сведены к минимуму затраты на хранение сырья и сохраняется прибыль от произведенной продукции, а также авторитет предприятия перед покупателем. Результаты работы предложенной имитационной модели могут использоваться в

качестве исходных данных для системы поддержки принятия решений о выборе системы управления запасами, объема и сроков поставок сырья и материалов.

Выводы

В данной работе представлена имитационная модель для анализа динамики поступлений и расхода материалов и сырья с учетом случайных факторов, результатом работы которой является история расходов сырья и его поставок. Результаты могут быть представлены как в текстовом виде, так и в виде графика зависимости общих расходов от гарантийного уровня запасов.

Полученные с помощью модели данные дадут возможность используя методы поддержки и принятия решения определять наиболее приемлемые параметры запасов, хранения и поставок сырья и материалов в условиях динамично меняющегося производства.

Литература

1. Логистика в малом бизнесе: Электронное учебное пособие [Электронный ресурс] // URL: <http://www.dist-cons.ru/modules/logistic/section7.html>;
2. Системы управления материальными запасами [Электронный ресурс] // URL: <http://logistic-info.ru/upravlenie-zapasami.html>;
3. Д.И. Левченко, В.А. Светличная Выбор метода прогнозирования для компьютеризированной подсистемы управления запасами сырья пищевого предприятия // Компьютерная и программная инженерия (КПИ-2015) / Материалы международной научно-технической конференции студентов, аспирантов и молодых ученых. — Донецк, ДонНТУ — 2015, с. 238-241.;
4. Зеваков А.М. Логистика материальных запасов и финансовых активов: Питер – 2005г.;
5. Моисеева Н.К. Экономические основы логистики: Учебник. – М74М.: ИНФРА-М, 2008. – 528 с. – (Высшее образование);
6. Стерлигова А.Н. Оптимальный размер заказа, или Загадочная формула Вильсона // Логистик & система. - №2. - С. 64-69. - №3. - С. 62-71. – 2005;
7. БОРГЕР Как рассчитать затраты на хранение товара [Электронный ресурс] // URL: <http://bor-ger.ru/kak-rasschitat-zatraty-na-hranenie-tovara/>

Левченко Д.И., Светличная В.А, Бабич К.К. Разработка имитационной модели для анализа динамики движения материалов и сырья с учетом рисков. Описаны общие факторы динамики движения сырья и материалов, влияющие на расходы предприятия. Предложена имитационная модель, учитывающая данные факторы и позволяющая определить такой объем хранимых на складах предприятия ресурсов, при котором возможно сократить затраты предприятия на хранение сырья и при этом полностью реализовать поставленный производственный план.

Ключевые слова: имитационное моделирование, расход сырья, управление поставками, сокращение запасов, риски.

Levchenko D., Svitlychna V., Babich K. Development of the risk-based simulation model to analyze dynamics of the income and consumption of raw materials. The general factors of raw materials and materials movement that affecting costs of the company were described. A simulation model that takes into account these factors and allows to define the best volume of raw materials in the warehouses of the enterprise was offered. Use of simulation results will allow to lower costs of the enterprise of raw materials storage and completely realize the delivered production plan

Keywords: simulation, consumption of raw materials, supply chain management, inventory reduction, risks.

Статья поступила в редакцию 20.11.2016
Рекомендована к публикации д-ром техн. наук В.Н. Павлышом

УДК 004.94

Моделирование внутренних операций процессорных элементов

Р.В. Мальчева, Т.В. Завадская
Донецкий национальный технический университет
raisa_malcheva@donntu.org

Мальчева Р.В., Завадская Т.В. Моделирование внутренних операций процессорных элементов. Выполнен анализ необходимости разработки демонстрационных, обучающих и имитационных моделей внутри процессорных операций. Приведены примеры моделей различного уровня. Выполнено проектирование устройств с использованием языков описания аппаратуры HDL и Verilog.

Ключевые слова: демонстрационная модель, средства обучения, проектирование, HDL, Verilog, исследования.

Введение

Стремительное развитие компьютерных и, в особенности, микропроцессорных технологий привело к отсутствию доступа к большинству компонентов вычислительных устройств (регистров, сумматоров, вентиляльных схем и т. п.), расположенных внутри одной или нескольких больших интегральных схем (БИС), что, соответственно, усложнило изучение внутренних процессов и операций.

В связи с этим все более актуальной становится задача разработки специальных моделей: физических или функциональных [1], позволяющих как демонстрировать внутреннее представление данных и выполнение операций, так и поддерживать различные стадии процесса проектирования процессорных элементов.

Подходы к обеспечению процесса обучения

Обеспечение образовательного процесса по дисциплинам, связанным с изучением архитектуры компьютеров, предусматривает создание учебных лабораторных комплексов на базе микропроцессорных комплектов или создание компьютерных моделей процессорных и других цифровых устройств [1]. Первый путь позволял проводить физическое исследование цифровых устройств, но требовал периодической замены лабораторной базы. Так за прошедшие годы на кафедре были введены в строй учебные лабораторные комплексы на микропроцессорных комплектах K580, K1804 и др., на которые истрачено достаточно много средств. Второй путь предполагал создание программной модели исследуемого процессора, цифрового устройства или вычислительной машины в целом, реализуемой в некоторой инструментальной среде. Такой моделью стала персональная

вычислительная машины, на которой легко изменять как структуру, так и программу изучения функционирования процессора и любого цифрового устройства [1].

На кафедре компьютерной инженерии ДонНТУ уже более 30 лет применяется и второй подход - проектирование гипотетических вычислительных машин для изучения с их помощью архитектуры компьютеров, а также использования самого процесса их разработки в качестве средства обучения.

Демонстрация процессорных операций

В последние десятилетия с появлением мультимедийных технологий изменились и презентационные свойства моделей [2]. Современные средства существенно облегчают процесс обучения через реализацию одного из фундаментальных его методов – наглядности. Наиболее удобной для создания демонстраций является технология Flash, которая поддерживает не только все презентационные возможности, но и позволяет демонстрировать работу некоторого процесса шаг за шагом с заданной частотой смены кадра [3]. Как показывает практика подготовки специалистов направления компьютерной инженерии, особое затруднение у студентов вызывает понимание машинного (внутреннего) представления данных и их обработки. Для повышения эффективности преподавания дисциплин, связанных с изучением основ информатики и особенностей выполнения машинных команд, был подготовлен значительный объем демонстрационных роликов [4]. В качестве примера на рис. 1 приведены слайды демонстрационного ролика выбора внутренних регистров процессора и формирования результата при выполнении операции целочисленного умножения для 8-мибитных и 16-тиразрядных операндов.

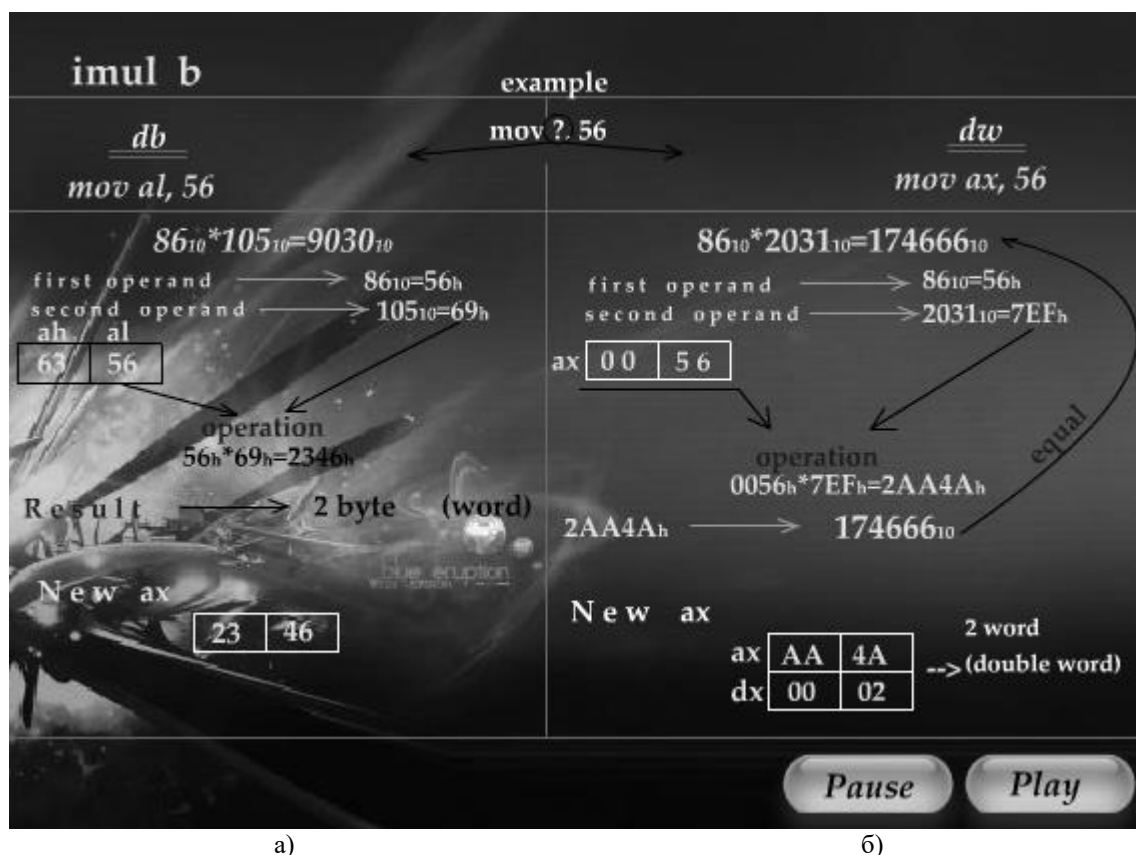


Рисунок 1 - Демонстрация выбора внутренних регистров процессора и формирования результата при выполнении операции целочисленного умножения для 8-мибитных (а) и 16-тиразрядных (б) операндов

Анимационные ролики используются не только во время чтения лекций. Разработаны Web-версии учебных дисциплин, со страниц которых студенты могут просматривать демонстрационные ролики, а также проходить самотестирование изученного материала.

Разработка моделей функционирования устройств

Как и во многих университетах [1, 5], в качестве заданий по курсовому проектированию применяется разработка студентами моделей (рис. 2) гипотетических компьютеров - упрощенных архитектур наиболее популярных промышленных процессоров (рис. 3).

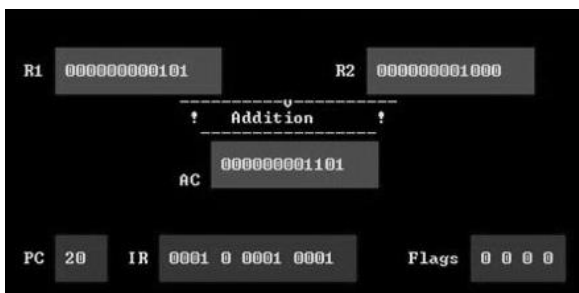


Рисунок 2 – Простейшая программная модель выполнения команды «сложение» в одноадресной гипотетической машине

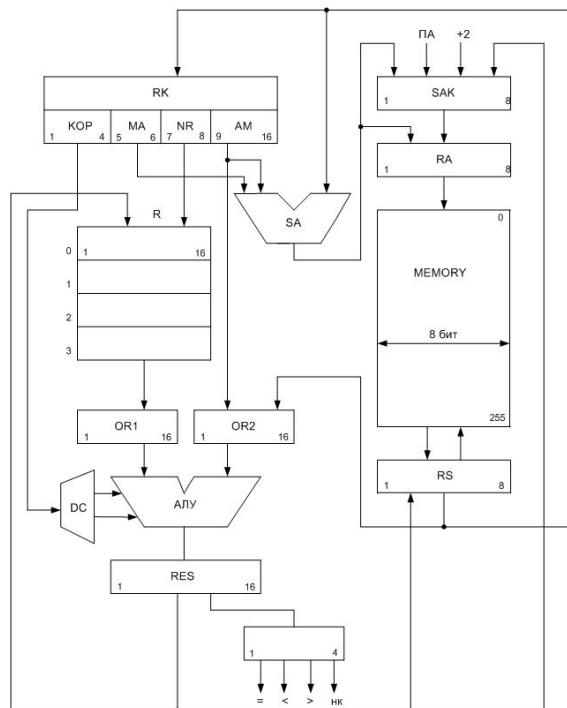


Рисунок 3 – Вариант архитектуры одноадресной гипотетической машины

Для ознакомления студентов с принципом работы одно- двух- и трехадресной вычислительной машины созданы специальные 66

Flash-фильмы [6]. Разработка программных моделей применяется и для изучения функционирования различных аналого-цифровых преобразователей. На рис. 4 приведен пример моделирования логики работы аналого-цифрового преобразователя поразрядного уравнивания.

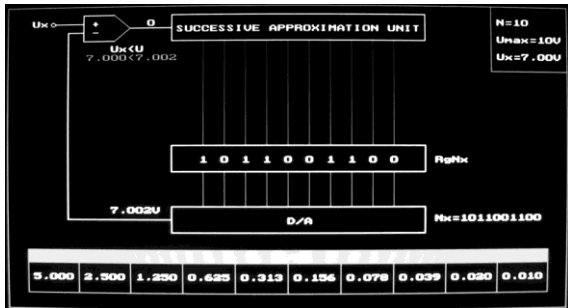


Рисунок 4 – Модель АЦП поразрядного уравнивания

Примеры проектирования устройств

Проектирование устройств выполняется с использованием таких языков описания аппаратуры, как HDL, Verilog и др. [7-10].

Рассмотрим пример разработки процессорного элемента для вычисления скалярного произведения *r* двух векторов в соответствии с математическим выражением (1):

$$r = x_1 \cdot x_2 + y_1 \cdot y_2 + z_1 \cdot z_2 , \quad (1)$$

где { *x*₁, *y*₁, *z*₁ } - компоненты 1-го вектора;

{ *x*₂, *y*₂, *z*₂ } - компоненты 2-го вектора.

Приведем код для вычисления скалярного произведения на языке Verilog:

```
module top( //выходы элемента
    output wire[15:0] r,
    output wire vis, // флаг видимости
    // входные данные
    input wire[15:0] x1,
    input wire[15:0] y1,
    input wire[15:0] z1,
    input wire[15:0] x2,
    input wire[15:0] y2,
    input wire[15:0] z2 );
// вычисление скалярного произведения
    assign r = x1*x2 + y1*y2 + z1*z2;
Endmodule
```

Вычисление формулы в программе автоматически разбивается средой на отдельные действия (*x1*x2*, *y1*y2*, ...), которые выполняются параллельно. Схема полученного устройства изображена на рис. 5. Чтобы показать параллельность вычислений, в код модели были добавлены промежуточные переменные *m1*, *m2*, *m3*, *s1* и внесены временные задержки следующим образом:

```
wire[15:0] m1;
wire[15:0] m2;
wire[15:0] m3;
wire[15:0] s1;
assign #20 m1 = x1*x2;
assign #20 m2 = y1*y2;
assign #20 m3 = z1*z2;
assign #5 s1 = m1+m2;
assign #5 r = s1+m3;
```

При моделировании получены следующие временные диаграммы (рис. 6).

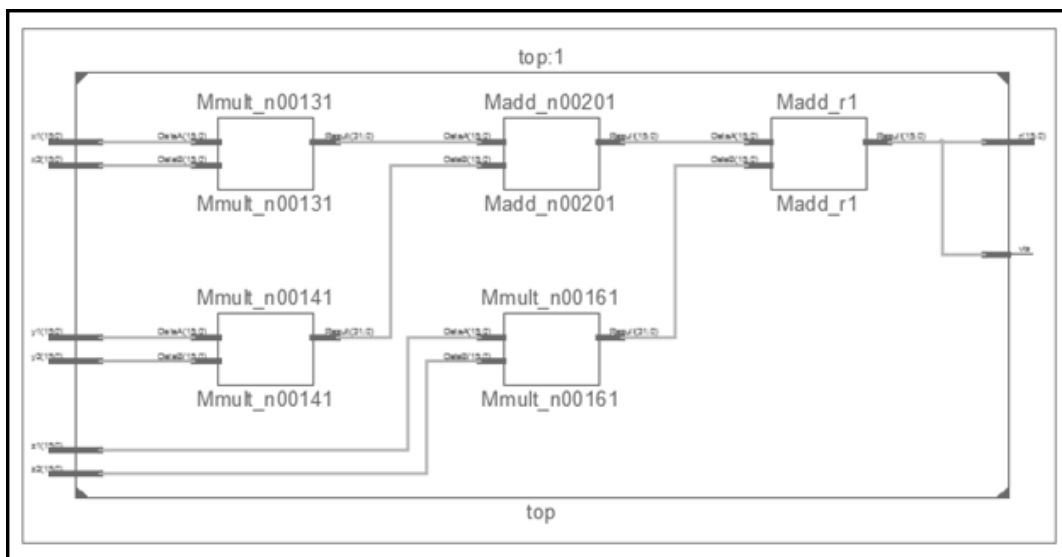


Рисунок 5 – Функциональная схема устройства для вычисления скалярного произведения

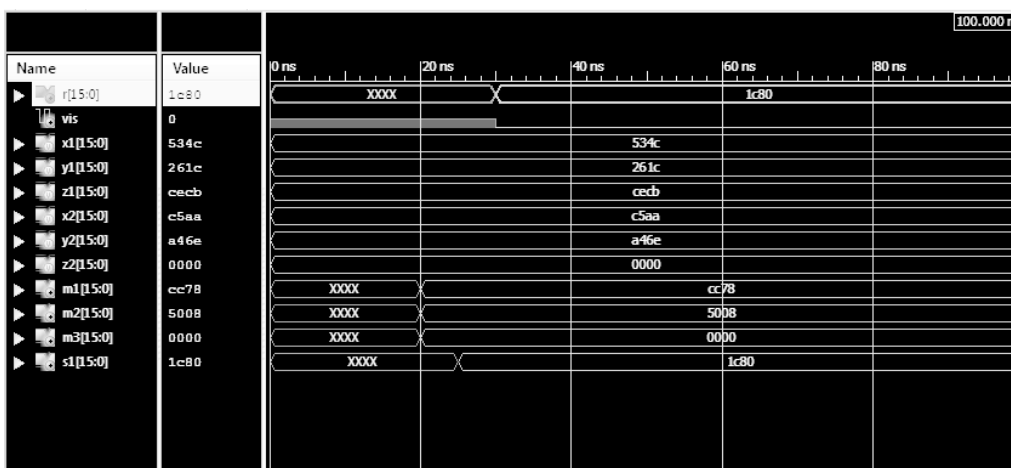


Рисунок 6 – Результаты моделирования устройства для вычисления скалярного произведения

Для описания более детального проектирования рассмотрим пример реализации в виде VHDL модели устройства умножения. Оно (рис. 7) состоит из умножителя, элемента И-2-ИЛИ, D-триггера, DC-триггера. При нажатии кнопки Reset, логический «0» поступает на DC триггер и на блок умножения, происходит обнуления регистра AC, записывается множимое и множитель, происходит умножение. D-триггер срабатывает по сигналу ready умножителя, он необходим для буферизации данных.

Умножитель (рис. 8) включает:

MC - 8 разрядный регистр множимого. Загрузка множимого осуществляется по команде Cload_mc с управляющего автомата.

MR - 8 разрядный регистр множителя. По команде Cload_mr в регистр загружается множитель, по команде Cshift_right_mr происходит логический сдвиг вправо. Нулевой разряд является признаком коррекции произведения.

AC - 16 разрядный регистр результата. Имеет логический сдвигатель вправо

(Cshift_right_ac), команду сброса (Creset), команду загрузки результата с АЛУ (Calu).

ALU - 8 разрядное арифметико-логическое устройство, содержит сумматор (Cadd/sub := 1) и вычитатель (Cadd/sub := 0).

СТ - счетчик, предназначенный для подсчета операций. Имеет признак переполнения CTF. Команда Creset_ct сбрасывает счетчик в исходное состояние, а Cinc - инкрементирует содержимое счетчика.

Control Unit - управляющий автомат.
VHDL модель MC:

```

process (CLK)
begin
//если восходящий вход CLK
if (rising_edge(CLK)) then
//и если сигнал Enable
if (EN='1') then
//то записать данные
Q <= D;
end if;
end if;
end process;
    
```

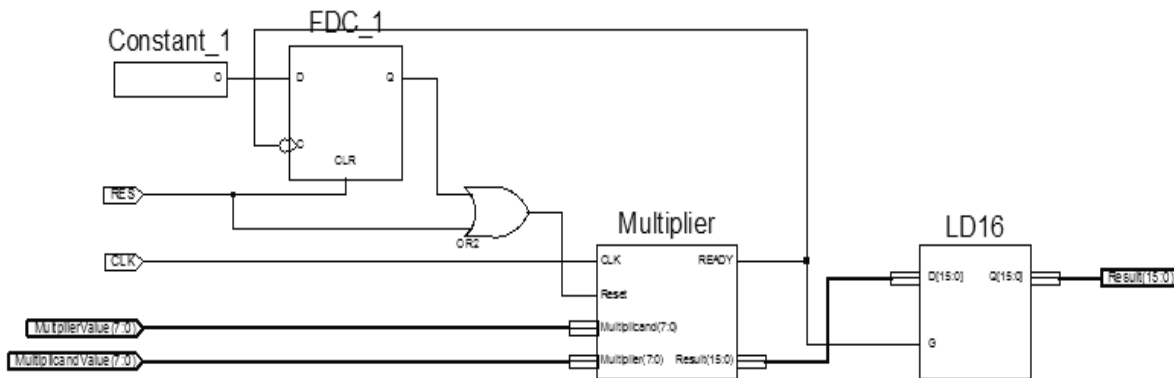


Рисунок 7 – VHDL модель устройства умножения

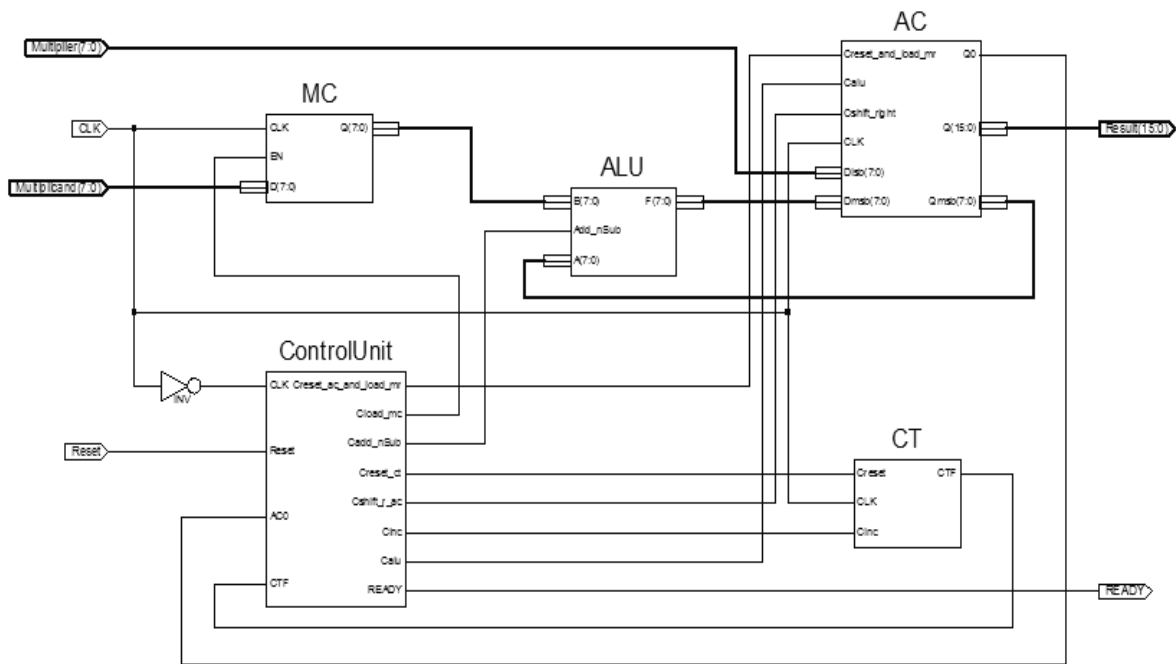


Рисунок 8 – VHDL модель умножителя

Блок АЛУ включает в себя:

- инкрементор:

```
process (A)
begin
    I <= A+1;
end process;
```

- инвертор;

- сумматор:

```
process (A,B)
variable S:STD_LOGIC_VECTOR (7 downto 0);
begin
    S := A + B;
    ModS <= S(6 downto 0);
    SignS <= S(7);
end process;
```

Для формирования произведения используется обычный сумматор, а после инвертируется знаковый разряд. Для выполнения операции вычитания необходимо поменять знак у второго операнда (B) и провести сложение. Смена знака производится путем инверсии операнда, а затем инкрементации полученного значения. Счетчик, реализованный в виде VHDL модели, включает в себя 4-х разрядный счетчик и элемент 3И и реализует следующую сущность:

```
entity CB4CE_HXILINX_CT is
port (
    Q0 : out STD_LOGIC;
    Q1 : out STD_LOGIC;
    Q2 : out STD_LOGIC;
    Q3 : out STD_LOGIC;
    C : in STD_LOGIC;
    CE : in STD_LOGIC;
    CLR: in STD_LOGIC
);
end CB4CE_HXILINX_CT;
```

Реализация счетчика выглядит следующим образом:

```
architecture Behavioral of CB4CE_HXILINX_CT
is
    signal COUNT : STD_LOGIC_VECTOR(3 downto 0)
    := (others => '0');
    constant TERMINAL_COUNT :
    STD_LOGIC_VECTOR(3 downto 0) := (others =>
    '1');
begin
    process(C, CLR)
    begin
        if (CLR='1') then
            COUNT <= (others => '0');
        elsif (C'event and C = '1') then
            if (CE='1') then
                COUNT <= COUNT+1;
            end if;
        end if;
    end process;
    Q3 <= COUNT(3); Q2 <= COUNT(2);
    Q1 <= COUNT(1); Q0 <= COUNT(0);
end Behavioral;
```

По сигналу CLR значение счетчика сбрасывается в исходное состояние. Нарастивание счетчика происходит по фронту сигнала CE. Для данной задачи необходим 3-х разрядный счетчик, поэтому сигнал переполнения добавляется логическим элементом 3И.

Регистр AC реализован в виде следующего кода:

```
process (CLK,RESULT)
begin
    if (rising_edge(CLK)) then
        if (Creset_and_load_mr = '1') then
            RESULT(15 downto 8) <= "10000000";
            //Сброс старшей части
            RESULT(7 downto 0) <= Dlsb;
            //Записываем множимое в младшую часть
        end if;
```

```

if (Calu = '1') then
    RESULT(15 downto 8) <= Dmsb;
end if;
if (Cshift_right = '1') then
    //логический сдвиг
    RESULT(15) <= RESULT (15);
    RESULT(14) <= not (RESULT(15));
    RESULT(13 downto 0) <= RESULT(14
downto 1);
end if;
end if;
Qmsb <= RESULT (15 downto 8);
//Множимое
Q0 <= RESULT (0);
//Младший бит
Q <= RESULT;
// Запись результата
end process;
    
```

```

when LAST SHIFT => ...
//состояние готовности
when RDY => ...
end case;
end if;
end process;
    
```

Плата устройства умножения разработана с использованием комплексной системы проектирования высокоскоростных устройств altium designer 16 [11]. Трассировка платы приведена на рис. 9. Для наглядности спроектирована трехмерная модель платы, показанная на рис. 10.

Блок Control Unit представлен в виде VHDL-модели и реализует алгоритм операции умножения в положительном нуле. В общем виде модель выглядит следующим образом:

```

process (CLK,Reset)
begin
    if (Reset = '1') then //Сигнал сброса
        state <= RST;
        //Если фронт CLK
        elsif rising edge (CLK) then
            case state is
                //Проверка текущего состояния
                when RST => ... //состояние сброса
                //проверка нулевого бита множителя
                when BIT CHECK => ...
                //сдвиг регистра AC
                when SHIFT => ...
                //проверка переполнения и проверка
                //знакового бита, после переполнения
                when CTF AND SIGNBIT CHECK => ...
                //сдвиг, если знаковый бит равен «1»
            end case;
        end if;
    end if;
end process;
    
```

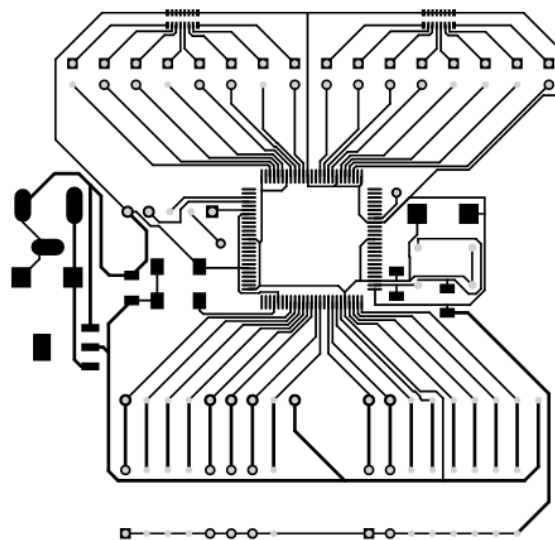
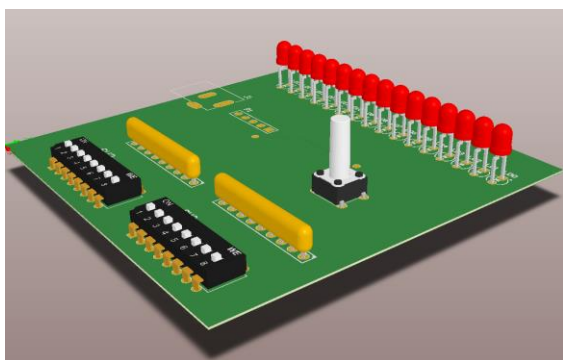
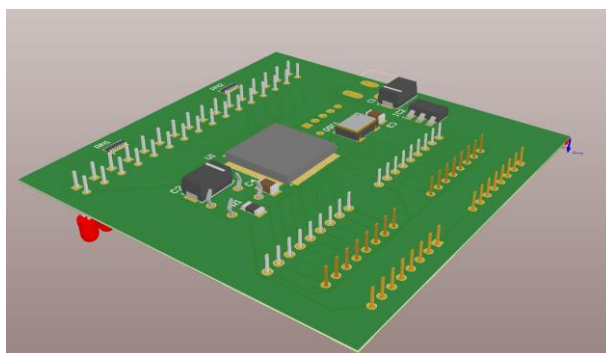


Рисунок 9 – Трассировка платы устройства умножения



а)



б)

Рисунок 9 – Трехмерная модель платы устройства умножения: а) вид сверху; б) нижняя панель платы

Т.о. кратко прослежены основные стадии процесса проектирования устройства.

Заключение

Для обеспечения дисциплин, связанных с изучением архитектуры и составных элементов компьютеров, необходимо применять широкий спектр демонстрационных и обучающих средств. Для их создания используются различные

современные технологии, такие как Power Point и Flash.

Для изучения основ проектирования элементов вычислительных систем и выполнения соответствующих курсовых и дипломных проектов необходимо использовать языки описания аппаратуры, такие как HDL, Verilog и др. Модели, созданные на языках проектирования VHDL или Verilog, можно не только запускать в средах проектирования, но и

загрузить в FPGA плату и подключить к нему устройства ввода и вывода. Существует множество отладочных FPGA стендов, которые уже имеют встроенные устройства ввода, такие, как кнопки и переключатели; и вывода – световые индикаторы, экран. Также в таких стендах имеются различные порты ввода/вывода, к которым можно подключить периферийные устройства.

Особое место в процессе организации учебного процесса занимает создание простых гипотетических машин, обладающих типичными чертами многих конкретных процессоров. Знание принципов построения и функционирования таких машин является хорошей базой для дальнейшего изучения и применения вычислительных систем.

Список использованной литературы

1. Кириллов В.В. Архитектура базовой ЭВМ. – СПб: СПбГУ ИТМО, 2010. – 144 с.
2. Malcheva R. Applying Internet technologies to improve the perception of lectures // Proceedings of 3d Congress EE. - Glasgow, 2002. - PP. 348-349.
3. Macromedia Flash MX 2004 ActionScript 2.0. Справочник разработчика: Пер. с англ. – М.: Издательский дом «Вильямс», 2005. - 896 с.
4. Мальчева Р.В. Использование видеороликов и игр при обучении иностранных студентов направления "Компьютерная инженерия" // Збірка праць V науково-методичної конференції "Проблеми і шляхи вдосконалення науково-методичної та

навчально-виховної роботи в ДонНТУ". – Донецьк: ДонНТУ, 2013.

5. Baranov S. Digital System Design. In book "Design of Digital Systems and Devices". Series: Lecture Notes in Electrical Engineering. – Springer-Verlag Berlin Heidelberg, Vol. 79, 2011. – P. 3- 41.

6. Мальчева Р.В., Сбитнева М. С. Интерактивная анимация как метод повышения эффективности изучения архитектуры компьютеров // Інформатика та комп'ютерні технології. Матеріали VI міжнародної науково-технічної конференції студентів, аспірантів та молодих науковців. - Донецьк, ДонНТУ, 2010.- Т.2. - С. 57–61.

7. Rawski M., Tomaszewicz P., Borowik G. Logic Synthesis Method of Digital Circuits Designed for Implementation with Embedded Memory Blocks of FPGAs. In book "Design of Digital Systems and Devices". Series: Lecture Notes in Electrical Engineering. - Springer-Verlag Berlin Heidelberg, Vol.79, 2011.– P. 121-144.

8. Хаханов В.И., Хаханова И.В., Литвинова Е.И., Гузь О.А. Проектирование и верификация цифровых систем на кристаллах. Verilog & System Verilog. – Харьков: ХНУРЭ, 2010. – 528 с.

9. Malcheva R., Naaem H. Development of the Data Transferring System Using SoC // European Scientific Journal, 2014. - Vol. 10. - N 7. - PP. 168-172.

10. Grout I. Digital Systems Design with FPGAs and CPLDs. – Amsterdam: Elsevier, 2008. – 406 pp.

11. <http://dugtor.ru/programmy/raznoe/25293-altium-designer-16111-build-255-multi-ru.html>

Поступила в редколлегию 12.2016

Malcheva R.V., Zavadskaya T.V. Simulation of the internal operations of the processing elements. The need for demonstration, training and simulation of the internal operations of the processing elements is analyzed. Examples of models at different functional levels are demonstrated. The processing devices are implemented using hardware description languages HDL and Verilog.

Keywords: model for demonstration, learning tools, designing, HDL, Verilog, investigation.

Мальчева Р.В., Завадська Т.В. Моделювання внутрішніх операцій процесорних елементів. Виконаний аналіз необхідності розробці моделей, що демонструють, навчають та імітують виконання операцій процесорів. Наведені приклади моделей різного рівня. Виконано проектування пристроїв з використанням мов опису апаратури HDL и Verilog.

Ключові слова: демонстраційна модель, засоби навчання, проектування, HDL, Verilog, дослідження.

Статья поступила в редакцию 20.11.2016

Рекомендована к публикации д-ром техн. наук В.Н. Павлышом

УДК 681.3.014

К вопросу робастности средств JPWL в условиях пакетных ошибок

С. В. Скороход, О. А. Касьянов
Южный федеральный университет
sss64@mail.ru

Скороход С. В., Касьянов О. А. К вопросу робастности средств JPWL в условиях пакетных ошибок. Рассматривается задача передачи видео в формате jpeg 2000 с применением средств помехоустойчивого кодирования JPWL по зашумленному каналу в условиях возникновения пакетных ошибок. Основная проблема – восстановление потерянного пакета данных средствами JPWL. Поставлена задача проведения экспериментального исследования робастности средств JPWL, под которой понимается способность корректировать пакетные ошибки. Методика исследования заключается в моделировании многократной передачи защищенного средствами JPWL видео кадра и подсчете статистики декодирования принятого кодового потока. Варьируемыми параметрами исследования являются: применяемые для защиты коды Рида-Соломона, количество тайлов изображения, процент потерь RTP (Real-time Transport Protocol) пакетов. Рассматриваются два способа защиты: стандартный и комбинация стандартного варианта с методом внутрикадрового чередования. Исследуемыми параметрами эксперимента являются средние значения процента полностью и частично восстановленных тайлов от общего количества тайлов в кодовом потоке. Описан разработанный для целей исследования программный комплекс, в который включаются кодер и декодер JPWL, средства разбиения кодового потока на RTP пакеты, средства моделирования пакетных потерь, средства сборки кадров из RTP пакетов. Описаны укрупненная схема функционирования кодера и декодера JPWL. Предложен метод внутрикадрового чередования кодового потока, предназначенный для повышения его устойчивости к пакетным ошибкам. В процессе исследования выдвинуты две гипотезы, которые были подтверждены полученными экспериментальными результатами. Гипотеза 1: стандартные средства JPWL не способны корректировать пакетные ошибки. Гипотеза 2: стандартные средства JPWL, применяемые совместно с алгоритмом чередования, могут восстанавливать пакетные ошибки. Побочным результатом исследования является вывод о некотором улучшении восстанавливаемости тайлов с ростом количества тайлов в кодовом потоке.

Ключевые слова: : jpeg 2000, JPWL, пакетные ошибки, помехоустойчивое кодирование.

Введение

В работе рассматривается общая проблема передачи видео в формате jpeg 2000 [1] по зашумленному каналу, в частности, по беспроводным сетям. Передача связана с пересылкой сетевых пакетов, несущих фрагменты кодированного видео потока. В результате зашумления канала некоторые пакеты могут быть искажены и отброшены. Следствием таких пакетных ошибок является отсутствие смежных фрагментов видео данных в приемнике, что либо затрудняет их декодирование, либо делает декодирование невозможным. Для защиты кодового потока jpeg 2000 при передаче по беспроводным сетям разработана спецификация ITUT.810 [2]. В ней описан набор средств, применяемых для одиночного изображения, называемых JPWL. Поскольку видео формата

jpeg 2000 представляет собой последовательность независимых друг от друга изображений [3], JPWL используется также для защиты видео данных.

Целью данной работы является проведение экспериментальных исследований способности средств JPWL корректировать пакетные ошибки. Под пакетной ошибкой понимается потеря как минимум одного сетевого пакета с данными при передаче от источника к приемнику [4]. Для проведения такого исследования разработан специализированный программный комплекс, который способен выполнять кодирование и декодирование JPWL, разбиение кодового потока JPWL на последовательность сетевых пакетов и обратную сборку кодового потока JPWL из сетевых пакетов с моделированием их потерь. В работе также рассматривается метод чередования, который может существенно повысить способность стандартных средств корректировать пакетные

ошибки, что также является предметом экспериментального исследования.

Исследуемый кодовый поток JPEG 2000

Методика исследования опирается на набор средств защиты JPWL, предназначенных для защиты кодового потока видео кадра (Скороход, Артюхова, 2014). При кодировании в формате jpeg 2000 кадр может быть разбит на несколько прямоугольных фрагментов (тайлов), каждый из которых кодируется отдельно. Структура кодового потока изображена на рис. 1 [5].

SOC	Основной заголовок	Заголовок тайла 1	Данные тайла 1	...	Заголовок тайла N	Данные тайла N	EOC
-----	--------------------	-------------------	----------------	-----	-------------------	----------------	-----

Рисунок 1 – Кодовый поток jpeg 2000

Маркеры SOC и EOC обозначают начало и конец конкретного кадра. Основной заголовок содержит данные об изображении в целом: размер, цветовая схема, количество тайлов и др. Заголовок тайла включает информацию о прямоугольном фрагменте изображения, в том числе о его привязке к координатной сетке. Данные тайла – это собственно закодированный фрагмент изображения.

Средства JPWL могут использовать два способа защиты [2]:

- контрольные суммы CRC-16 или CRC-32 позволяют только обнаружить присутствие ошибок в данных при использовании минимального объема кодов четности;

- коды Рида-Соломона [6] $RS(n,k)$ (RS -коды) предназначены не только для обнаружения, но и для исправления $(n-k)/2$ искаженных данных. Здесь n – длина кодового слова в байтах, состоящего из защищаемого фрагмента данных и кодов четности, а k – количество байт защищаемых данных. Код $RS(n,k)$ позволяет исправить до $(n-k)/2$ искаженных байт в кодовом слове. Спецификацией ITUT.810 предусмотрено использование целого семейства RS -кодов от наиболее «слабых» $RS(37,32)$ до наиболее «сильных» $RS(128,32)$ и $RS(160,64)$.

Поскольку контрольные суммы не могут использоваться для защиты от пакетных ошибок, в настоящей работе рассматривается использование только RS -кодов.

Защита кодового потока выполняется добавлением в него специальных сегментов. Предусмотрены четыре вида сегментов.

- EPC – возможность защиты от ошибок. Вставляется только в основной заголовок. Его задача – сообщить декодеру, что в кодовом потоке присутствуют средства JPWL. Содержит описание использованных методов защиты и, при необходимости, их параметры.

- EPB – блок защиты от ошибок. Собственно реализует защиту данных с

использованием RS -кодов. Вставляется в основной заголовок и заголовки тайлов. Содержит параметры применяемой защиты и избыточные коды четности.

- ESD – дескриптор чувствительности. Задает степень чувствительности к ошибкам участков кодового потока. Чем больший вклад в формируемое изображение вносит фрагмент кодового потока, тем больше его чувствительность.

- RED – дескриптор остаточной ошибки. Используется только в заголовках декодированного кодового потока. Описывает участки, которые либо полностью восстановлены, либо не были восстановлены декодером JPWL и содержат остаточные ошибки.

Сегмент ESD используется только кодером JPWL с целью реализации более надежной защиты наиболее ценных участков кодового потока. Для декодера JPWL он не несет никакой полезной информации, существенно увеличивая при этом размеры заголовков и всего кодового потока в целом. В связи с этим при передаче видео целесообразно отказаться от использования сегмента ESD [7] и в настоящем исследовании этот сегмент не используется.

План исследования

Методика исследования заключается в моделировании передачи защищенного средствами JPWL видео, состоящего из 1000 кадров одного и того же изображения размером 1024x768 пикселей, предварительно закодированного кодером jpeg 2000 системы OpenJpeg 2.1.0 [8]. Полученный кодовый поток загружается в разработанную автором исследовательскую программную систему, в которой он последовательно проходит следующие этапы обработки:

- кодирование JPWL;
- разбиение на RTP-пакеты, соответствующие спецификации RFC 5371 [9, 10];
- моделирование потерь пакетов по заданному проценту потерь;
- сборка кодового потока из оставшихся пакетов;
- декодирование JPWL и подсчет статистики декодирования.

Неизменными параметрами в исследовании являются способ защиты заголовков и размер RTP пакета. Для защиты основного заголовка и заголовков тайлов применяются RS -коды, предопределенные в спецификации T.810. Размер RTP пакета равен 1024 байт.

Переменными параметрами исследования являются виды RS -кодов, применяемых для защиты данных, процент потери пакетов и количество тайлов, на которые разбивается

изображение (определяется выбранным размером тайла). Программа исследования приведена в табл. 1.

Таблица 1. Программа исследования

Количество тайлов (размер тайла)	RS-коды для защиты данных	Процент потерь RTP пакетов
12 (256x256) 48 (128x128) 192 (64x64)	RS(37,32), RS(64,32), RS(96,32), RS(128,32)	1, 3, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50

Статистическими результатами декодирования выступают количество полностью и частично восстановленных тайлов кодового потока.

Следует отметить, что наиболее важной частью кодового потока является основной заголовок. Поэтому для его дополнительной защиты обычно применяются специальные меры, такие как избыточное дублирование пакетов, содержащих его фрагменты, в связи с чем, в данном исследовании пакетные ошибки не будут применяться к пакетам основного заголовка.

Исследовательский программный комплекс

Для данного исследования был разработан специализированный программный комплекс, состоящий из двух модулей: подготовительного и моделирующего. Схема функционирования комплекса изображена на рис. 2.

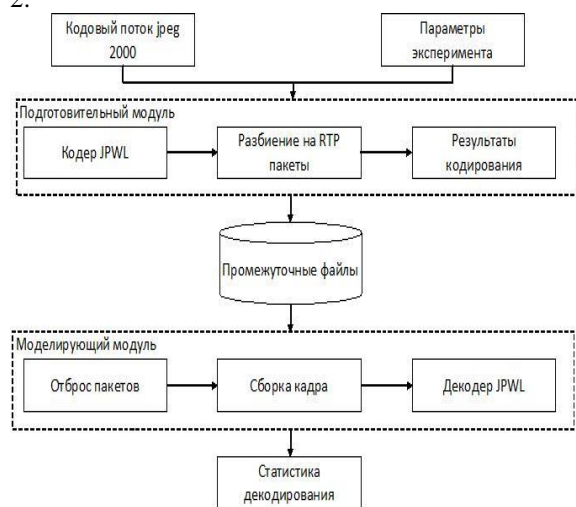


Рисунок 2 – Структура программного комплекса

Исходными данными для подготовительного модуля является кодовый

поток jpeg 2000, предварительно создаваемый при помощи OpenJpeg 2.1.0, содержащий 12, 48 или 192 тайла в соответствии с программой исследования, а также набор параметров исследования, в которых задается режим кодирования (с использованием чередования или без) и процент отбрасываемых пакетов. Подготовительный модуль выполняет кодирование JPWL, при необходимости – чередование, разбивает полученный кодовый поток на RTP пакеты в соответствии со спецификацией RFC 5371 и выполняет выгрузку в промежуточные файлы полученную последовательность пакетов, их параметры (количество, длины и т.д.) и параметры исследования.

Моделирующий модуль загружает из промежуточных файлов подготовленную последовательность пакетов и параметры исследования. Далее выполняются циклические итерации отброса пакетов, сборки кадра и его обработки декодером JPWL. Декодер подсчитывает количество полностью и частично восстановленных тайлов. После выполнения 1000 подобных итераций вычисляется среднее арифметическое этих величин, характеризующее способность декодера восстановить кодовый поток при заданных параметрах кодирования и заданном уровне пакетных ошибок.

Главными звеньями данного комплекса являются кодер и декодер JPWL. В данном исследовании используются библиотеки кодера и декодера, разработанные автором для системы видеотрансляции [11, 12].

Обобщенная схема работы кодера JPWL приведена на рис. 3.

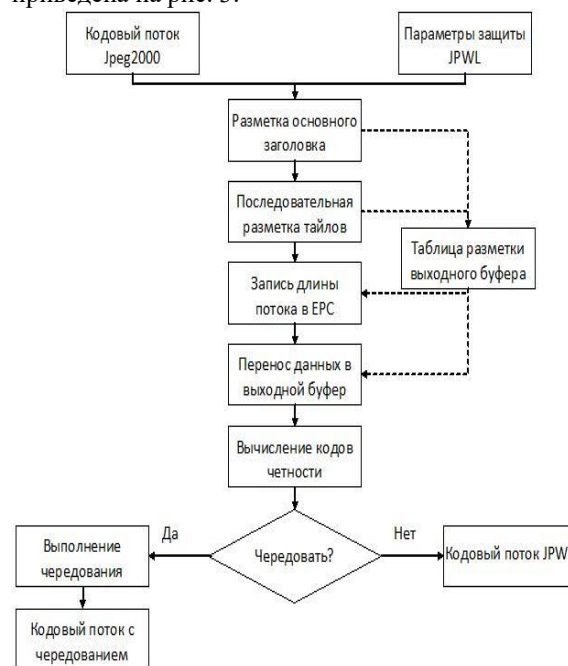


Рисунок 3 –Кодер JPWL

Входными данными являются собственно кодový поток `jpeg 2000` и параметры используемых средств защиты JPWL. Кодер выполняет разметку основного заголовка и каждого последующего тайла в выходном буфере. При этом данные о параметрах маркеров JPWL и их местоположении заносятся в таблицу разметки выходного буфера. После окончания прохода входного буфера вычисляется длина выходного потока как сумма длины входного потока и длин всех добавляемых в него маркеров. Полученное значение длины помещается в таблицу разметки маркера EPC. Далее выполняется копирование на свои места в выходном потоке исходных данных и параметров маркеров JPWL. Последним шагом кодирования является вычисление и вставка в выходной поток кодов четности маркеров EPB. После этого шага формируется кодový поток JPWL, удовлетворяющий требованию обратной совместимости, т.е. декодер `jpeg 2000` способен декодировать данный поток, если в нем нет искажений. При применении алгоритма чередования дополнительно выполняется чередование, результатом которого является кодový поток JPWL, удовлетворяющий требованию обратной совместимости с расширением: декодер `jpeg 2000` способен декодировать такой поток только при наличии декодера JPWL.

Обобщенная схема работы декодера JPWL изображена на рис. 4.

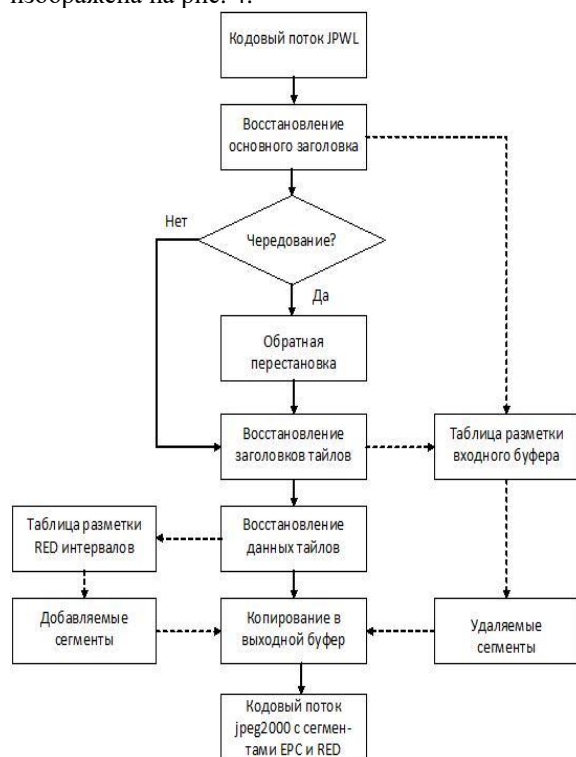


Рисунок 4 – Декодер JPWL

Декодер JPWL выполняет декодирование предварительно собранного кадра, последовательно анализируя основной заголовок и заголовки тайлов. При невозможности восстановления основного заголовка весь кадр отбрасывается. Если основной заголовок успешно восстановлен, то анализ сегмента EPC позволяет определить применяется чередование кодového потока или нет.

Если чередование применяется, декодер сначала выполняет обратную перестановку, и только после этого переходит к анализу тайлов. При невозможности восстановления заголовка тайла пропускается весь тайл и ищется заголовок следующего тайла. При этом пропущенный фрагмент помечается в таблице разметки входного буфера как испорченный. В случае успешного восстановления основного заголовка или заголовка тайла в таблицу разметки заносятся параметры и местоположение всех найденных маркеров JPWL. Восстановление данных тайла выполняется только в случае успешного восстановления его заголовка. При этом часть данных тайла может оказаться скорректированной, а часть – нет. В этом случае создаются записи в таблице RED интервалов, которые описывают начало и конец как исправленных, так и не исправленных фрагментов данных.

По завершении просмотра входного буфера выполняется копирование данных в выходной буфер. При этом выполняется пропуск всех входных маркеров JPWL. При наличии записей в таблице RED интервалов в основной заголовок вставляется сегмент EPC, сигнализирующий о присутствии в кодovém потоке сегментов RED, а в заголовки частично восстановленных тайлов добавляются сегменты RED, описывающие адреса восстановленных и не восстановленных участков.

Подсчет статистики декодера JPWL выполняется на основе результатов восстановления заголовков и данных тайлов. Если после декодирования заголовок тайла не содержит сегмента RED, такой тайл считается полностью восстановленным. Если сегмент RED присутствует – тайл восстановлен частично. Наличие сегмента RED ничего не говорит о качестве данных в частично восстановленном тайле, поэтому в данном исследовании этот вопрос не затрагивается.

Внутрикадровое чередование

Кодер и декодер JPWL поддерживают как стандартный способ защиты, соответствующий спецификации T.810, так и алгоритм чередования, не относящийся к этой спецификации [13]. Сущность алгоритма заключается в том, что основной заголовок остается без изменений, а остальная часть кодového потока подвергается

перестановке. Основной заголовок содержит сегмент EPC, в котором записана общая длина кодового потока. Разность общей и длины основного заголовка определяет длину данных, которые должны быть переставлены. Эти данные рассматриваются как прямоугольная матрица, заполненная «по строкам». Данные в этой матрице переставляются так, чтобы она оказалась заполнена «по столбцам».

Пусть L – общая длина кодового потока, M – длина основного заголовка. Тогда согласно [14] количество столбцов и строк матрицы вычисляется по формулам:

$$N_C = \lceil \sqrt{L-M} \rceil, N_R = \left\lceil \frac{L-M}{N_C} \right\rceil.$$

Следует отметить, что в результате такой перестановки фактическая длина кодового потока несколько увеличивается на стороне кодера, поскольку в матрицу может быть включено до $N_R - 1$ «лишних» элементов. Кодер и декодер JPWL должны учитывать этот факт.

Отличие предлагаемого способа чередования от описанного в [14] заключается в том, что в данном случае чередованию подвергаются весь кодовый поток за исключением основного заголовка. В [14] предлагается перестановка только кодов четности, записанных в сегментах EPB.

Анализ результатов исследования

В исследовании проверяются две основные гипотезы.

Гипотеза 1. Стандартные средства JPWL, соответствующие спецификации T.810, не способны корректировать пакетные ошибки. Критерием, подтверждающим эту гипотезу, является обратная линейная зависимость процента пакетных потерь и суммарного количества полностью и частично восстановленных тайлов.

Проанализируем стандартный способ защиты с применением RS -кодов. Кодовое слово $RS(n,k)$ состоит из двух частей расположенных в разных частях кодового потока. Первую часть составляют избыточные коды четности длиной $n-k$ байт, которые расположены в основном заголовке или заголовках тайлов. Вторая часть – k информационных байт, защищаемых от искажения. При потере пакета возможны три варианта [13].

1. Обе части кодового слова потеряны – восстановление невозможно.

2. Потеряны избыточные коды четности – восстановление невозможно.

3. Потеряны только информационные байты – восстановление возможно только если $k \leq (n-k)/2$. Этому условию соответствуют только коды $RS(160,64)$, $RS(80,25)$, $RS(40,13)$, применяемые для защиты заголовков тайлов, и $RS(96,32)$,

$RS(112,32)$, $RS(128,32)$, применяемые для защиты данных тайлов. Остальные 14 предопределенных стандартом кодов не способны произвести восстановление.

Анализируемые в исследовании способы защиты кодами $RS(37,32)$, $RS(64,32)$ не способны скорректировать данные ни в одном из перечисленных вариантов. При применении $RS(96,32)$ и $RS(128,32)$ возможна коррекция данных в варианте 3, однако расплатой за это является значительное увеличение заголовков тайлов, что делает их гораздо уязвимее и может нивелировать возможный положительный эффект.

Гипотеза 2. Стандартные средства JPWL, применяемые совместно с алгоритмом чередования, могут восстанавливать пакетные ошибки. Способность восстановления зависит от используемых RS -кодов и процента пакетных потерь. Критерием, подтверждающим эту гипотезу, является нелинейная зависимость процента пакетных потерь и суммарного количества полностью и частично восстановленных тайлов, при которой до определенного уровня потерь наблюдается стопроцентное восстановление тайлов.

Для проверки гипотезы 1 проведем эксперимент 1. Эксперимент проводится над кодовым потоком jpeg 2000, содержащем 48 тайлов, к которому применяется стандартная защита. Применяемые RS -коды и моделируемые проценты потерь RTP пакетов заданы в программе исследований. Результаты эксперимента приведены в табл. 2 с округлением до целых величин. Для каждого вида RS -кодов указан размер кодового потока JPWL, средний процент полностью и частично восстановленных тайлов.

Таблица 2. Результаты эксперимента 1

Потери пакета в %	RS(37,32) 1025691 байт		RS(64,32) 1767653 байт		RS(96,32) 2647205 байт		RS(128,32) 3526937 байт	
	Полн	Част	Полн	Част	Полн	Част	Полн	Част
1	81	18	71	28	69	30	58	40
3	53	44	34	63	35	61	21	76
5	35	60	18	77	16	79	8	85
10	12	78	4	86	4	86	1	87
15	5	81	1	84	1	84	0	81
20	2	78	0	81	0	80	0	76
25	1	75	0	76	0	75	0	71
30	0	69	0	70	0	70	0	65
35	0	64	0	65	0	64	0	60
40	0	59	0	59	0	61	0	54
45	0	55	0	55	0	56	0	50
50	0	50	0	49	0	51	0	43

Данный результат полностью подтверждает гипотезу 1. Более того, надежды на то, что RS -коды $RS(96,32)$ и $RS(128,32)$ дадут лучший результат по сравнению с $RS(37,32)$ и

RS(64,32) не оправдались по причине чрезмерного увеличения заголовков тайлов при применении этих кодов.

На рис. 5 изображены графики зависимости суммарного процента полностью и частично восстановленных тайлов (ордината) от процента пакетных ошибок (абсцисса) для эксперимента 1.

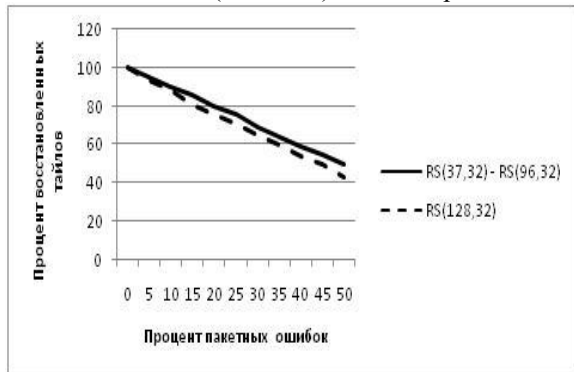


Рисунок 6 –Зависимость для эксперимента 1

В рамках проверки гипотезы 1 проводились также эксперименты 2 и 3, в которых использовался кодовый поток jpeg 2000, состоящий соответственно из 12 и 192 тайлов. Остальные условия соответствовали эксперименту 1. Результаты экспериментов 2 и 3 практически аналогичны приведенным в табл.2, с незначительным улучшением при большем количестве тайлов и незначительном ухудшении – при меньшем.

Полученные зависимости полностью соответствуют критерию, подтверждающему гипотезу 1 во всех экспериментах. Кроме того, можно сделать вывод о том, что с увеличением количества тайлов увеличивается процент их восстановления. Это объясняется тем, что чем больше тайлов, тем меньше их размер, что уменьшает вероятность повреждения конкретного тайла при пакетной ошибке. Наоборот, чем меньше тайлов, тем больше их размер и более уязвимыми для пакетной ошибки они становятся.

Для проверки гипотезы 2 проведем эксперимент 4. Исходные данные для него соответствуют эксперименту 1, но стандартный способ защиты комбинируется с алгоритмом чередования. Результаты эксперимента приведены в табл. 3.

Таблица 3. Результаты эксперимента 4

Потери пакета в %	RS(37,32)		RS(64,32)		RS(96,32)		RS(128,32)	
	Полн	Част	Полн	Част	Полн	Част	Полн	Част
1	39	61	100	0	100	0	100	0
3	0	100	100	0	100	0	100	0
5	0	100	100	0	100	0	100	0
10	0	100	98	2	100	0	100	0

15	0	100	28	72	100	0	100	0
20	0	100	0	100	85	15	100	0
25	0	98	0	99	7	91	83	15
30	0	92	0	92	0	92	7	84
35	0	72	0	73	0	72	0	70
40	0	37	0	39	0	39	0	35
45	0	11	0	11	0	11	0	9
50	0	1	0	1	0	1	0	1

Анализ табл. 3 подтверждает гипотезу 2. Действительно, при уровне пакетных потерь до 25% любые RS-коды восстанавливают практически все тайлы, изменяется лишь распределение полностью и частично восстановленных тайлов. С дальнейшим ростом пакетных потерь процент восстановленных тайлов начинает снижаться. В эксперименте 4 проявилась еще одна явная зависимость: при применении более «длинных» RS-кодов качество восстановления улучшается, прежде всего за счет большего процента полностью восстановленных тайлов.

На рис. 6 изображен график зависимости суммарного процента полностью и частично восстановленных тайлов (ордината) от процента пакетных ошибок (абсцисса) для эксперимента 4. Полученные зависимости полностью соответствуют критерию, подтверждающему гипотезу 2.

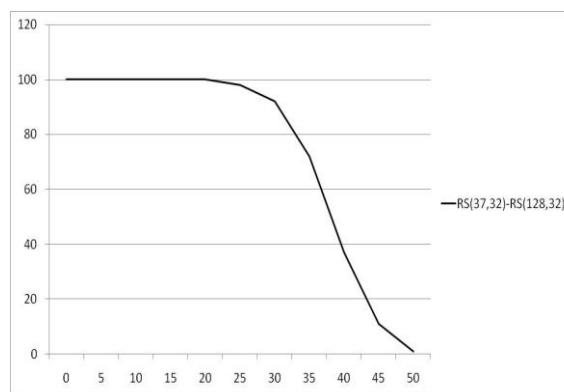


Рисунок 6 –Зависимость для эксперимента 4

В рамках данного исследования были проведены эксперименты 5 и 6, условия которых совпадали с условиями эксперимента 4, но был использован кодовый поток jpeg 2000, содержащий соответственно 12 и 192 тайла. Результаты этих экспериментов во многом совпадают с таблицей 5 и рисунком 4 с некоторым ухудшением при 12 тайлах и улучшением при 192.

Заключение

Данное исследование проведено на программном комплексе, разработанном на языке Си в системе программирования Microsoft Visual Studio 2008. Полученные результаты могут быть

использованы для выбора оптимальных параметров JPWL в зависимости от характеристик канала передачи данных.

По результатам проведенных экспериментов можно сделать следующие выводы.

1. Набор стандартных средств JPWL, соответствующих спецификации T.810 является недостаточным для эффективной коррекции пакетных ошибок.

2. Сочетание стандартных средств JPWL с предложенным в работе алгоритмом чередования позволяет успешно корректировать пакетные ошибки.

3. Увеличение количества тайлов незначительно повышает устойчивость кодового потока к пакетным ошибкам.

Направлением дальнейших исследований является экспериментальное исследование влияния пакетных ошибок на качество декодированного изображения в формате jpeg 2000. Для проведения этого исследования требуется разработать декодер jpeg 2000, способный декодировать частично восстановленный кодовый поток, в котором могут отсутствовать некоторые тайлы, а в заголовках имеющихся тайлов присутствовать сегменты остаточной ошибки RED. К сожалению, известные доступные декодеры, такие как JasPer, OpenJpeg не могут декодировать такой кодовый поток.

Работа выполнена в рамках выполнения базовой части государственного задания в сфере научной деятельности (проект № 3442 "Информационно-алгоритмическое обеспечение систем цифрового управления, автономной высокоточной навигации и технического зрения для перспективных летательных аппаратов: разработка теоретических основ проектирования, алгоритмов, способов эффективной и надежной программной реализации, использование высокопроизводительной вычислительной инфраструктуры для экспериментального моделирования").

Литература

1. Taubman, D.S., Marcellin M.W. JPEG2000: Image Compression Fundamentals, Standards and Practice. – Boston: Kluwer Academic Publishers, 2002. – 773 p.

2. ITU-T Recommendation T.810. Information Technology JPEG2000 Image Coding System: Wireless. – Geneva: ITU, 2007. – 60 p.

3. Acharya T., Tsai P. JPEG 2000 standard for image compression: concepts, algorithms and VLSI architectures.– Hoboken : John Wiley & Sons, Inc., 2005. – 274 p.

4. Скороход С. В., Степанова А. П. Некоторые аспекты технологии сетевого взаимодействия между передатчиком и приемником в процессе видео трансляции // Известия ЮФУ. Технические науки.– №4 (165), .2015.– с. 171–181.

5. ITU-T Recommendation T.800. Information Technology JPEG2000 Image Coding System: Core Coding System. – Geneva: ITU-T Series T: Terminal for Telematic Services, 2003. – 212 p.

6. Морелос-Сагоса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение.– М: Изд-во Техносфера, 2005.– 320 с.

7. Skorokhod S.V., Kravchenko P.P., Khusainov N.S. Some aspects of JPWL implementation for streaming video // International Conference «Engineering & Telecommunication En&T 2014». November 26-28, 2014. Book of Abstracts. – Moscow– Dolgoprudny : MIPT, 2014. p. 135-136.

8. An open-source JPEG 2000 codec written in C. Retrieved April 6, 2016, from <http://www.openjpeg.org/>.

9. Network Working Group RFC 5371. RTP Payload Format for JPEG 2000 Video Streams. – The Internet Security (IETF), 2006. – 31 p.

10. Network Working Group RFC 3550. RTP: A Transport Protocol for Real-Time Applications.— Schulzrinne: BlueCoatSystemsInc., 2003.– 89 p.

11. Дроздов С.Н., Жиглатый А.А., Кравченко П.П., Скороход С.В., Хусаинов Н.Ш. Об опыте реализации системы видеотрансляции в формате jpeg2000 и перспективах применения стандарта jpeg2000 для передачи видео и мультиспектральных данных с борта бпла// Известия ЮФУ. Технические науки.– 2014, №7 (156).– с. 161–170.

12. Дроздов С.Н., Жиглатый А. Н., Кравченко П. П., Лутай В. Н., Скороход С. В., Хусаинов Н. Ш. Стандарт JPEG 2000: базовые алгоритмы, примеры реализации и перспективы применения.– Ростов-на-Дону, Изд-во ЮФУ, 2014.– 255 с.

13. Skorokhod S. V. The experimental studies of the jpwL tools ability to correct burst errors in a noisy channel when transmitting video in a jpeg 2000 format // Journal of Theoretical and Applied Information TechNology, 10th November, Vol. 81, № 1, 2015. – p. 34-42.

14. ITU-T Recommendation T.810. Information Technology JPEG 2000 image coding system: Wireless. Amendment 1: IP based wireless networks.— Geneva: ITU-T Series T: Terminal for Telematic Services, 2012.–6 p.

Скоруход С. В., Касьянов О. А. К вопросу робастности средств JPWL в условиях пакетных ошибок. Рассматривается задача передачи видео в формате jpeg 2000 с применением средств помехоустойчивого кодирования JPWL по зашумленному каналу в условиях возникновения пакетных ошибок. Основная проблема – восстановление потерянного пакета данных средствами JPWL. Поставлена задача проведения экспериментального исследования робастности средств JPWL, под которой понимается способность корректировать пакетные ошибки. Методика исследования заключается в моделировании многократной передачи защищенного средствами JPWL видео кадра и подсчете статистики декодирования принятого кодового потока. Варьируемыми параметрами исследования являются: применяемые для защиты коды Рида-Соломона, количество тайлов изображения, процент потерь RTP (Real-time Transport Protocol) пакетов. Рассматриваются два способа защиты: стандартный и комбинация стандартного варианта с методом внутрикадрового чередования. Исследуемыми параметрами эксперимента являются средние значения процента полностью и частично восстановленных тайлов от общего количества тайлов в кодовом потоке. Описан разработанный для целей исследования программный комплекс, в который включаются кодер и декодер JPWL, средства разбиения кодового потока на RTP пакеты, средства моделирования пакетных потерь, средства сборки кадров из RTP пакетов. Описаны укрупненная схема функционирования кодера и декодера JPWL. Предложен метод внутрикадрового чередования кодового потока, предназначенный для повышения его устойчивости к пакетным ошибкам. В процессе исследования выдвинуты две гипотезы, которые были подтверждены полученными экспериментальными результатами. Гипотеза 1: стандартные средства JPWL не способны корректировать пакетные ошибки. Гипотеза 2: стандартные средства JPWL, применяемые совместно с алгоритмом чередования, могут восстанавливать пакетные ошибки. Побочным результатом исследования является вывод о некотором улучшении восстанавливаемости тайлов с ростом количества тайлов в кодовом потоке.

Ключевые слова: : jpeg 2000, JPWL, пакетные ошибки, помехоустойчивое кодирование.

Skorokhod S. V., Kasyanov O. A. The experimental studies of the jpwl tools ability to correct burst errors in a noisy channel when transmitting video in a jpeg 2000 format. The discussion is made on the issue of transmitting video in a JPEG 2000 format using the JPWL tools (Wireless JPEG 2000) for noise-immunity coding through a noisy channel under conditions of burst errors occurrence. It is set out to conduct an experimental study on the ability of JPWL tools to correct burst errors. The methodology of the study consists in modeling a transmission of the JPWL protected video composed of 1,000 frames of one and the same image with a size of 1,024x768 pixels. The variable parameters of the study are as follows: the Reed-Solomon codes applied for protection, the number of image tiles, and the RTP (Real-time Transport Protocol) packet loss ratio. A standard variant of protection and a combination of the standard variant with an interleaving algorithm are considered. The final results of the experiment are the average values of the percentage of fully and partially restored tiles of the number of tiles in a code stream. A software system developed for the conduction of study is described, which includes the JPWL encoder and decoder, the tools for partitioning code streams into RTP packets, the tools for packet loss modeling, and the tools for frame assembly from the RTP packets. A macro flowchart of the JPWL encoder and decoder functioning is described. A method for the code stream interleaving is suggested, which is intended for increasing its resistance to burst errors. During the study process, the two hypotheses were put forward, which were confirmed by the experimental results obtained. The first hypothesis is that the standard JPWL tools are not able to correct burst errors. The second hypothesis is that the standard JPWL tools used in conjunction with the interleaving algorithm can recover burst errors. A side effect of the study is the conclusion on a slight improvement of the recoverability of tiles with the increasing number of tiles in a code stream.

Keywords: JPEG 2000, JPWL, burst errors, noise-immunity coding.

Статья поступила в редакцию 20.11.2016
Рекомендована к публикации д-ром техн. наук В.Н. Павлышом

УДК 621.1.016.4: 519.63

Численное моделирование и управление процессами теплопереноса в противоточных теплообменных аппаратах

В. К. Толстых, К. А. Пшеничный
Донецкий национальный университет
mail@tolstykh.com, kirillpch@mail.ru

Толстых В.К., Пшеничный К.А. Численное моделирование и управление процессами теплопереноса в противоточных теплообменных аппаратах.

Предложена упрощённая модель описания конвективных, нестационарных, одномерных процессов теплопереноса для теплообменников с противоточными теплоносителями, содержащая модельный коэффициент теплообмена между теплоносителями, включающий в себя как физический коэффициент теплопередачи, так и внутренние геометрические особенности теплообменника. Описана конечно-разностная схема расчёта тепловых процессов. Приведены расчёты с данными промышленного теплообменника (кислородная станция Енакиевского металлургического завода) для оценочных значений коэффициента теплообмена. Сформулирована задача идентификации коэффициента теплообмена. Поставлена задача оптимального управления теплообменным аппаратом с целью стабилизации температурного режима работы ректификационной колонны.

Ключевые слова: теплообменный аппарат, теплообменник, идентификация, оптимальное управление, численное моделирование.

Введение

Теплообменные аппараты используются во многих отраслях промышленности, таких как металлургия, энергетика, химическое производство. От качественной работы теплообменников в значительной степени зависит работа других объектов-потребителей различных ресурсов с заданными температурными характеристиками.

На рис. 1 приведена часть технологической схемы воздухоразделительной установки ректификационной колонны Енакиевского металлургического завода. Воздух (сплошная жирная линия) проходит основной теплообменник Е3118, после которого поступает в нижнюю ректификационную колонну Т3211. После процесса разделения воздуха из верхней колонны Т3212 отбирается некоторое количество жидкого кислорода (пунктирная линия) и посредством управляемого насоса Р3568 подаётся в основной теплообменник Е3118 для противоточного охлаждения воздуха.

Для качественной работы ректификационной колонны в любое время суток и года, экономии энергоресурсов и обеспечения сохранности оборудования в случае непредвиденных сбоев, необходимо адекватное управление теплообменником для удержания воздушного потока в колонну при номинальной температуре.

Поэтому важно уметь с достаточной точностью моделировать работу теплообменника для обеспечения оптимальных режимов работы ректификационной колонны.

В данной работе рассматриваются конвективные, нестационарные, одномерные процессы теплопереноса для теплообменников с противоточными теплоносителями. Одна среда теплоносителя – это охладитель жидкий кислород, вторая – охлаждаемый воздух. Рассматриваемая принципиальная математическая модель данных процессов для различных теплообменных аппаратов предполагает, что конструктивные и технологические особенности конкретных аппаратов должны учитываться некоторым адекватным модельным коэффициентом теплообмена между противоточными средами. Очевидно, что для достаточно точного численного моделирования работы теплообменника по предлагаемой модели необходимо решать задачу идентификации данного коэффициента.

В первом разделе настоящей работы описывается математическая модель процессов теплопереноса между двумя противоточными средами с учётом гидродинамических особенностей потоков. Второй раздел посвящён численному моделированию конвективных тепловых процессов по конечно-разностным схемам типа «уголок». Далее обсуждаются результаты расчётов численного моделирования потоков обеих сред и влияние на тепловые

процессы модельного коэффициента теплообмена. Предпоследние разделы посвящены обсуждению полученных результатов и возможного алгоритма идентификации коэффициента теплообмена, а также – задаче оптимального управления теплообменным аппаратом, по идентифицированной модели, с целью стабилизации температурного режима работы ректификационной колонны.

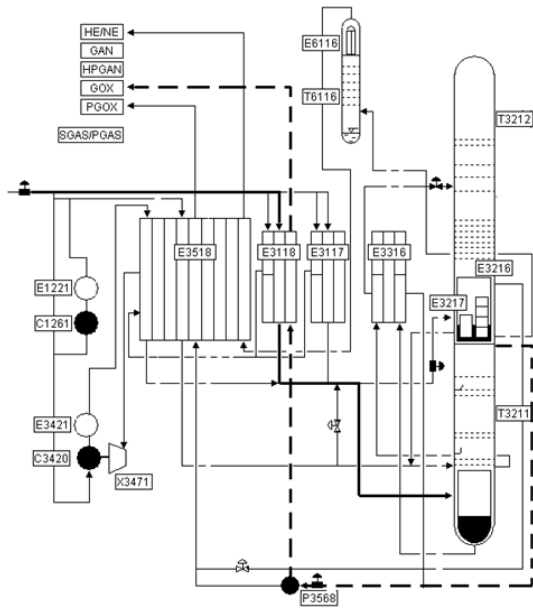


Рисунок 1. Схема воздуходелительной установки.

Математическая модель процессов теплопереноса

На рис. 2 показана принципиальная схема работы противоточного теплообменника в котором прямым потоком является воздух, а обратным – кислород. В одномерной постановке конвективные нестационарные тепловые процессы для указанного теплообменника можно описать следующей системой дифференциальных уравнений [1, 2]:

$$\begin{cases} \frac{\partial T_1}{\partial \tau} + v_1 \frac{\partial T_1}{\partial x} = \frac{1}{c_{p1} \rho_1} K(T_2 - T_1), \\ \frac{\partial T_2}{\partial \tau} - v_2 \frac{\partial T_2}{\partial x} = \frac{1}{c_{p2} \rho_2} K(T_1 - T_2), \end{cases} \quad (1)$$

где $T_1(\tau, x)$ и $T_2(\tau, x)$ температуры первого и второго теплоносителя (соответственно – воздух и жидкий кислород), движущихся со скоростями $v_1(\tau, x)$ и $v_2(\tau, x)$, K – модельный коэффициент, определяющий теплообмен между теплоносителями, включающий в себя

как физический коэффициент теплопередачи, так и внутренние геометрические особенности теплообменника, $\tau \in [0, \theta]$ – время, $x \in [x_a, x_b]$ – пространственная координата, c_p – удельная теплоемкость при постоянном давлении, $\rho(T)$ – плотность теплоносителей.

Граничные условия для системы (1) это:

$$\begin{aligned} T_1(\tau, x_a) &= T_{1a}(\tau), \\ T_2(\tau, x_b) &= T_{2b}(\tau). \end{aligned} \quad (2)$$

Как правило, начальная температура теплоносителей вдоль всего теплообменника неизвестна. Зададим в качестве начального приближения линейное изменение температуры вдоль теплообменника при заданных граничных значениях:

$$\begin{aligned} T_1(0, x) &= T_1(0, x_a) - \\ & [T_1(0, x_a) - T_1(0, x_b)] \frac{x - x_a}{x_b - x_a}, \\ T_2(0, x) &= T_2(0, x_b) - \\ & [T_2(0, x_b) - T_2(0, x_a)] \frac{x - x_b}{x_a - x_b}. \end{aligned} \quad (3)$$

Такое приближение будет приводить к искусственным переходным тепловым процессам в начале расчётов до некоторого момента $\tau_0 \in (0, \theta)$. После τ_0 начальные условия уже не будут влиять на дальнейшие расчёты, и полученное распределение температур можно будет считать новым начальным условием для дальнейших расчётов.

Для определения τ_0 будем задавать стационарные граничные условия как тепловые, так и гидродинамические. Момент τ_0 будет определяться как момент стабилизации искусственных переходных тепловых процессов.

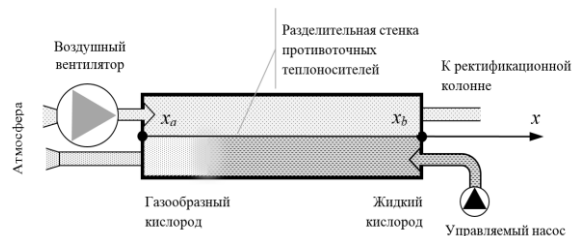


Рисунок 2. Схема противоточного теплообменного аппарата первой (высокотемпературной) ступени

Поскольку плотность теплоносителей существенно зависит от температуры [3], то и скорость течений теплоносителей будет

различной по длине теплообменника. Волновыми процессами в промышленных теплообменниках пренебрегают из-за их быстротечности. Поэтому гидродинамические течения теплоносителей будем принимать стационарными для каждого расчётного временного слоя $\tau^j \in [0, \theta]$. Тогда из уравнения неразрывности $div(\rho v) = 0$ мы получаем уравнения для расчёта скоростей течений теплоносителей:

$$\begin{aligned} \rho_1^j(T) v_1^j(x) &= C_1^j, \\ \rho_2^j(T) v_2^j(x) &= C_2^j, \end{aligned} \quad (4)$$

где C_1^j, C_2^j – набор констант, которые определяются через известную скорость на границах потоков: $C_1^j = \rho_{1a}^j v_{1a}^j, C_2^j = \rho_{2b}^j v_{2b}^j$. Плотность и температура вещества связаны уравнением состояния. В нашей постановке мы использовали уравнение Менделеева-Клапейрона для моля вещества: $PV_\mu = RT$, $\rho = \frac{m_\mu}{V_\mu}$, где m_μ – молярная масса, V_μ – молярный объем.

Таким образом, уравнения (1)-(4) образуют математическую модель для расчёта тепловых процессов рассматриваемой принципиальной схемы противоточного теплообменника. Зная приблизительное распределение температур в теплоносителях на некотором временном слое, мы можем найти из (4) распределение скоростей потоков теплоносителей, а из (1) уточнить новые значения температур.

Численное решение задачи для процессов теплопереноса

Для численного решения системы двух уравнений (1) с (2), (3) проводилась конечно-разностная аппроксимация каждого уравнения на одной и той же равномерной сетке по схемам типа «уголок» [4], которые показаны на рис. 3. Причем для прямого хода теплоносителя – это будет левый уголок с x_{i-1}, x_i , а для обратного потока теплоносителя – эта же схема приобретает вид правого уголка с x_i, x_{i+1} .

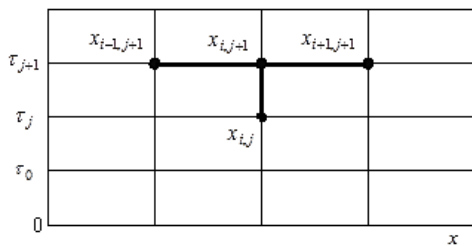


Рисунок 3. Конечно-разностная схема с двумя уголками для противоточных сред

Данная аппроксимация дифференциальных уравнений (1) приводит к следующим алгебраическим уравнениям:

$$\begin{cases} \frac{T_{1,i}^{j+1} - T_{1,i}^j}{\Delta \tau} + v_{1,i}^j \frac{T_{1,i}^{j+1} - T_{1,i-1}^{j+1}}{\Delta x} = \\ \frac{1}{c_{p1} \rho_{1,i}^j} K (T_{2,i}^j - T_{1,i}^j), \\ \frac{T_{2,i}^{j+1} - T_{2,i}^j}{\Delta \tau} - v_{2,i}^j \frac{T_{2,i}^{j+1} - T_{2,i-1}^{j+1}}{\Delta x} = \\ \frac{1}{c_{p2} \rho_{2,i}^j} K (T_{1,i}^j - T_{2,i}^j). \end{cases}$$

Откуда можно найти

$$\begin{cases} T_{1,i}^{j+1} = \frac{\gamma}{\gamma + 1} T_{1,i-1}^{j+1} + \frac{1}{\gamma + 1} T_{1,i}^j + \\ \frac{\Delta \tau}{1 + \gamma} \frac{1}{c_{p1} \rho_{1,i}^j} K (T_{2,i}^j - T_{1,i}^j), \\ T_{2,i}^{j+1} = \frac{\gamma}{\gamma + 1} T_{2,i+1}^{j+1} + \frac{1}{\gamma + 1} T_{2,i}^j + \\ \frac{\Delta \tau}{1 + \gamma} \frac{1}{c_{p2} \rho_{2,i}^j} K (T_{1,i}^j - T_{2,i}^j) \end{cases} \quad (5)$$

при условии:

$$\gamma = \frac{v_{\max} \Delta \tau}{\Delta x} \leq 1,$$

где $v_{\max} = \max_{\tau,x}(v_1, v_2)$. Расчёт решения для первой среды находится слева-наперво, а для второй – справа-налево на каждом временном слое τ^j .

Результаты расчётов и оценка влияния коэффициента теплообмена

Модель (1)-(4) можно будет считать приемлемой для какого-нибудь теплообменника, только если коэффициент K можно будет задать таковым, что модельная температура будет мало отличаться от экспериментально наблюдаемой температуры для этого же теплообменника. Оценим значение и влияние модельного коэффициента K на тепловые процессы в условиях промышленного теплообменника (кислородный цех Енакиевского металлургического завода).

На рис. 4 представлены результаты расчётов распределения температуры в обеих теплоносителях для двух различных коэффициентов теплообмена $K_1 = 10^4 \frac{\text{Вт}}{\text{м}^3\text{К}}$ и

$K_2 = 10^5 \text{ Вт/м}^2\text{К}$. Расчёты проводились при следующих данных. Длина теплообменного аппарата $L = 10 \text{ м}$. Приближения для начальных условий температуры рассчитывались по (3) при $T_1(0, x_b) = 315,15 \text{ К}$, $T_2(0, x_a) = 303 \text{ К}$. Температура на границах теплообменника задавалась постоянной и равной $T_1(\tau, x_a) = 315,15 \text{ К}$, $T_2(\tau, x_b) = 90,2 \text{ К}$. Давления в потоках принимались $p_1 = 5 \cdot 10^6 \text{ Па}$, $p_2 = 2,4 \cdot 10^6 \text{ Па}$. Граничные скорости потоков $v_1(\tau, x_a) = 76 \text{ м/с}$, $v_2(\tau, x_b) = 79 \text{ м/с}$. Расчётная сетка задавалась с ячейками по пространству $\Delta x = 1 \text{ м}$ и шагами по времени $\Delta \tau = 0,01 \text{ с}$.

Как показали расчёты, переходные процессы стабилизировались к моменту $\tau_0 \approx 20 \text{ с}$. После этого времени нестационарность процессов, вносимая приближительными начальными условиями (3), исчезала. Система переходила в установившееся состояние. Других источников нестационарности в рассматриваемом примере не было. Полученное время стабилизации совпадает со временем переходных процессов теплообменного аппарата Енакиевского металлургического завода.

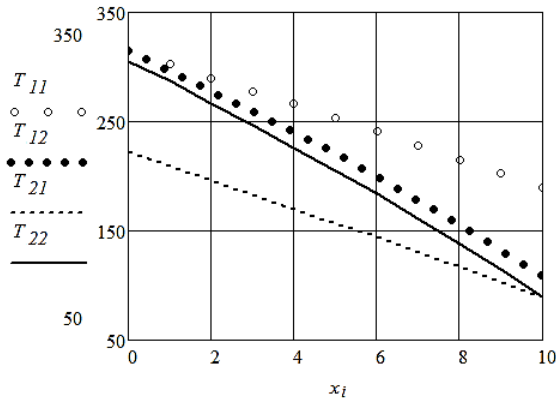


Рисунок 4. Распределение температур в воздухе T_{11}, T_{12} и в кислороде T_{21}, T_{22} при K_1 и K_2 соответственно

Из приведенных двух расчётов со значениями K_1 и K_2 видно, что коэффициент K существенно влияет на тепловые процессы теплообменника. При большем коэффициенте теплообмена K_2 температуры обеих сред в большей степени сближаются, что иллюстрируют кривые T_{12} и T_{22} . При меньшей теплообмена (коэффициент K_1) температуры противоточных сред слабо приближаются друг к другу, что видно по кривым T_{11} и T_{21} .

Распределение температур T_{12} и T_{22} близко к реальным рабочим температурам теплообменника Енакиевского металлургического завода. Это свидетельствует о корректности приведенного численного моделирования противоточных процессов и, что коэффициент теплообмена K_1 был выбран близким к правильному модельному значению.

Задача идентификации

Математическая модель (1)-(4) содержит в уравнении (1) заранее неизвестный коэффициент K . В нашем случае — это модельный, а не физический коэффициент, включающий в себя погрешности всех допущений модели: неопределённость точной геометрии теплообменника, значительность толщины разделительной стенки и отсутствие учёта в ней тепловых процессов, приближённое гидродинамическое моделирование и т.д. Очевидно, что коэффициент K не поддаётся непосредственному измерению и для каждой конкретной установки может быть найден только с использованием оптимизационных методов идентификации [5, 6, 7] на основе экспериментальных данных, например, по результатам измерений температуры потоков.

Очевидно, что удобнее (проще) проводить идентификацию при установившейся работе теплообменника. В этом случае задача идентификации будет формулироваться следующим образом. Необходимо найти коэффициент K модели (1)-(4) доставляющий минимум функционалу:

$$J(K) = \int_{x_a}^{x_b} [T_1(\tau_0, x) - T_e(\tau_0, x)]^2 dx \rightarrow \min \quad (6)$$

где $J(K)$ — критерий качества идентификации, который в нашем случае представляет собой неявную одномерную функцию модельного коэффициента теплообмена K , T_e — экспериментально наблюдаемая температура воздуха в теплообменнике. Коэффициент K , найденный из условия минимума J , будет содержать в себе все погрешности математической модели (1)-(4) таким образом, чтобы температура, рассчитываемая в дальнейшем по этой модели, была максимально приближена к реальной рабочей температуре теплообменника.

Функционал (6) может быть «облегчён» с учётом того, что мы рассматриваем установившиеся процессы, а сам коэффициент K одинаков по всей длине теплообменника. Здесь расхождение модельной и реальной температур можно контролировать только на выходе x_b охлаждённого воздуха из теплообменника. В

этом случае критерий качества идентификации принимает вид:

$$J(K) = [T_{1b}(\tau_0) - T_e(\tau_0)]^2 \rightarrow \min \quad (7)$$

Для решения задач идентификации (6), (7) могут применяться различные методы одномерной минимизации, например, метод золотого сечения, как метод поиска экстремума функции одной переменной на заданном отрезке. Мы имеем $J(K): [K_a, K_b] \rightarrow R$. Отрезок $[K_a, K_b]$ делится в пропорции золотого сечения, то есть выбираются две точки K_1, K_2 из условия:

$$\begin{aligned} K_1 &\approx K_a + 0,4(K_b - K_a), \\ K_2 &\approx K_a + 0,6(K_b - K_a). \end{aligned} \quad (8)$$

Далее в этих точках оцениваются значения критерия качества. Если $J(K_1) \leq J(K_2)$, тогда принимают $K_b \leftarrow K_2$ иначе $K_a \leftarrow K_1$ и все расчёты повторяют заново до уменьшения длины отрезка $[K_a, K_b]$ до желаемых минимальных размеров. После чего определяется решение задачи (7) как $K = (K_b + K_a)/2$.

Задача оптимального управления теплообменником

Для стабилизации температурного режима на входе в ректификационную колонну воздухоразделительной установки (см. рис. 1) необходимо «правильно» управлять насосом нагнетания жидкого кислорода в теплообменник (см. рис. 2). Правильная работа насоса должна обеспечить требуемую температуру воздуха $T_{1b}(\tau)$ на выходе их теплообменника независимо от температуры атмосферного воздуха и возможной некачественной работы компрессора на входе x_a в теплообменник. Это означает, что нашим управлением $u(\tau)$ будет скорость подачи охладителя $v_2(\tau, x_b)$ в теплообменник, которая фигурирует в граничном условии уравнений (5). Таким образом, мы получаем множество управляющих воздействий $\{u^j \equiv v_{2b}^j\}$, $j = 1, 2, \dots$, определяющих температуру воздуха для ректификационной колонны в граничном условии (4):

$$C_2^j = \rho_{2b}^j u^j. \quad (9)$$

Задача стабилизации температурного режима на входе в ректификационную колонну воздухоразделительной установки будет формулироваться следующим образом. Необходимо найти оптимальное управление

модели (1)-(4), (9) которое доставляет минимум функционалу

$$J_0 = \int_{\tau} [T_{1b} - T_*]^2 d\tau \rightarrow \min, \quad (10)$$

где T_* – требуемая номинальная температура воздуха на входе в ректификационную колонну воздухоразделительной установки.

Поскольку решение уравнений теплопереноса (1) аппроксимируется конечными разностями на тех же временных слоях $\{\tau^j\}$, что и решение гидродинамической задачи, то целевой функционал (10) примет вид:

$$J_0 = \sum_j [T_{1b}^j - T_*]^2 \rightarrow \min \quad j = 1, 2, 3, \dots \quad (11)$$

Для решения подобной задачи оптимизации могут применяться, например, методы динамического программирования [8,9] или прямая минимизация целевого функционала [10,11].

Выводы

Полученные результаты расчётов течений противоточных сред (рис. 4) для общей упрощённой модели теплообменника (1)-(4) с конечно-разностной схемой (5) показали физически обоснованные распределения температур в теплообменнике. Для использования полученной модели на практике необходимо решать задачу идентификации коэффициента теплообмена K приведенной модели, что можно делать по упрощённому критерию (7) методом золотого сечения. Идентифицированная модель может использоваться для расчёта оптимальных режимов работы теплообменника для стабилизации температурного режима ректификационной колонны по критерию (11).

Литература

1. Девятов, Б. Н. Теория переходных процессов в технологических аппаратах с точки зрения задач управления. – Новосибирск: Наука, 1964. – 324 с.
2. Девятов, Б. Н. Теория и методы анализа управляемых распределенных процессов / Б.Н. Девятов, Н.Д. Демиденко. – Новосибирск: Наука, 1983. – 310 с.
3. Ландау, Л.Д., Лифшиц, Е.М. Теоретическая физика гидродинамика. 3-е изд., испр. – М.: Наука. Гл. ред. физ.-мат. лит., 1986. – 736 с.
4. Самарский, А. А. Теория разностных схем. М.: Наука, 1977. – 656 с.
5. Толстых В.К., Недопекин Ф.В., Бодряга В.Е. Идентификация теплофизических

параметров в виде полиномов, зависящих от температуры. – Технічна теплофізика та промислова теплоенергетика, N1. – 2009. – С. 193-199.

6. Tolstykh V.K. Optimality Conditions and Algorithms for Direct Optimizing the Partial Differential Equations, Engineering. Scientific Research Publishing, Vol. 4, N7. – 2012, pp. 390-393.

7. Tolstykh V.K. Optimization for Systems Governed by Partial Differential Equations, The Electronic International Journal "Advanced Modeling and Optimization", Vol. 14, N3. – 2012, pp. 703-716.

8. Гладких, Б. А. Методы оптимизации и исследование операций для бакалавров информатики. Ч. 2. Нелинейное и динамическое программирование: учебное пособие. – Томск :

Изд-во НТЛ, 2011. – 264 с. – ISBN 978-5-89503-483-5.

9. Васильев, Ф. П. Методы оптимизации: В 2-х кн. Новое изд., перераб. и доп. – М : МЦНМО, 2011. Кн. 2. – 433 с. – ISBN 978-5-94057-708-9.

10. Толстых В.К. Прямой экстремальный подход для оптимизации систем с распределенными параметрами. – Донецк: "Юго-Восток", 1997. – 178 с.

11. Огурцов А.П., Недопекин Ф.В., Толстых В.К., Володин Н.А. Прямая оптимизация теплофизических процессов. – Днепродзержинск: ДГТУ-"Юго-Восток". – 1997. – 150 с.

Толстых В.К., Пшеничный К.А. Численное моделирование и управление процессами теплопереноса в противоточных теплообменных аппаратах.

Предложена упрощенная модель описания конвективных, нестационарных, одномерных процессов теплопереноса для теплообменников с противоточными теплоносителями, содержащая модельный коэффициент теплообмена между теплоносителями, включающий в себя как физический коэффициент теплопередачи, так и внутренние геометрические особенности теплообменника. Описана конечно-разностная схема расчёта тепловых процессов. Приведены расчёты с данными промышленного теплообменника (кислородная станция Енакиевского металлургического завода) для оценочных значений коэффициента теплообмена. Сформулирована задача идентификации коэффициента теплообмена. Поставлена задача оптимального управления теплообменным аппаратом с целью стабилизации температурного режима работы ректификационной колонны.

Ключевые слова: теплообменный аппарат, теплообменник, идентификация, оптимизация, численное моделирование.

Tolstykh V.K., Pshenichny K.A Numerical simulation and process control of heat transfer in counterflow heat exchangers. A simplified model for the description of convection, unsteady, one-dimensional heat transfer processes for heat exchangers with counterflow heat transfer fluids is proposed. It contains a model heat transfer coefficient between the heat-transfer agents, which includes both physical heat transfer coefficient, and internal geometrical features of heat exchanger. It is described the finite-difference scheme for calculating thermal processes. Results calculations with the data of industrial heat exchanger for some values of heat transfer coefficient were described. There was formulated a problem of identification of heat transfer coefficient. There was formulated a task of optimal control of the heat exchanger in order to stabilize the temperature regime of operation of the distillation column.

Keywords: heat exchanger, identification, optimization, numerical simulation.

Статья поступила в редакцию 20.11.2016
Рекомендована к публикации д-ром техн. наук В.Н. Павлышом

УДК 004.432.2

Проектирование и разработка компилятора C-подобного языка программирования общего назначения с поддержкой исполнения кода на этапе компиляции

Н.М. Ткачёв, А.И. Андрюхин
Донецкий национальный технический университет
m4dbrat@gmail.com, alexandruckin@rambler.ru

Ткачёв Н.М., Андрюхин А.И. Проектирование и разработка компилятора C-подобного языка программирования общего назначения с поддержкой исполнения кода на этапе компиляции. В статье представлен краткий обзор современных принципов проектирования языков программирования и сформированы требования к проектируемому языку. Спроектирована начальная итерация языка программирования общего назначения и кратко описана структура разрабатываемого компилятора. Перспективами развития языка является расширение системы типов и внедрение в компилятор комплексных средств анализа кода для предотвращения ошибок и исполнения кода на этапе компиляции.

Введение

Задача проектирования новых языков программирования в наше время является как никогда актуальной, поскольку широкое распространение компьютерных технологий ставит задачи максимальной оптимизации процесса разработки и написания кода в различных предметных областях. Именно из-за широкого использования тех или иных технологий становятся более заметными их недостатки, потому новые средства призваны их исправлять [1].

Высокоуровневые средства программирования значительно ускоряют решение типичных задач, но зачастую их гибкости может не хватать, поскольку обстоятельства решаемой задачи требуют более точного контроля над ситуацией. В частности, наиболее распространённой является проблема производительности, когда управление абстракциями приводит к вычислениям, не влияющим на конечный результат. Другой проблемой являются «протекающие абстракции» - явление, при котором программисту приходится отвлекаться от решения задачи на попытки понять внутреннее устройство используемых им средств. Как правило, такая проблема возникает при недостаточном внимании при проектировании средств разработки или несовместимости парадигм среды разработки, языка и библиотек. Например, Win32 API пытается сочетать объектно-ориентированную модель на основе обмена сообщениями вместе с низкоуровневым процедурным стилем языка C, из-за чего программисту приходится выполнять множество неочевидных действий и увеличиваются

временные затраты на отладку.

Язык программирования является нижним слоем, на котором держится процесс разработки, потому важно убедиться в том, чтобы разработчики могли чётко выражать в нём свои намерения (абстракции и алгоритмы, оперирующие ими), по минимуму отвлекаясь на языковые особенности («бороться» с языком).

Цель работы: исследование современных принципов проектирования языков программирования и их проблем, проектирование языка на их основе полученных сведений, а также разработка и развитие компилятора этого языка.

1 Цели при проектировании языков программирования

Ввиду развития и распространения множества различных парадигм программирования были выделены следующие требования к языкам, выполнение которых позволяет оптимизировать процесс разработки:

1. Язык программирования должен иметь синтаксис и модель работы, которые позволяют максимально просто записывать некоторые манипуляции и в то же время позволяют чётко их видеть при чтении кода. Такие языки, как Java, могут требовать написания сотен строк повторяющегося кода для решения задач, которые языки наподобие Haskell могут решить в одну строку. Однако необходимо помнить, что более краткий код может использовать более сложные понятия, потому на его понимание может уйти гораздо больше времени, чем на чтение аналогичного, но более длинного кода. Баланс между этими сторонами должен соблюдаться как

программистами, так и проектировщиками языка. Наиболее распространённой формой языка в этом плане являются языки с C-подобным синтаксисом и объектно-ориентированной моделью с некоторыми элементами функционального программирования (например, lambda-выражениями и замыканиями).

2. Язык, спроектированный для определённой предметной области, должен позволять прозрачно манипулировать понятиями данной области. Язык общего назначения должен быть достаточно гибким для формирования абстракций различных предметных областей, а также давать возможность построения связей между этими областями.

3. Язык должен предотвращать типичные для языков более низкого уровня логические ошибки, превращая их в семантические. Обычно к этой цели стремятся языки с гибкой системой типов, позволяя программистам описывать допустимые и недопустимые связи между объектами в определённых общих понятиях.

4. Абстракции, встроенные в язык, должны нести минимальную нагрузку, необходимую для их применения. Язык C++ работает на принципе «платишь только за то, чем пользуешься», позволяя, например, отказаться от таблицы виртуальных функций при использовании классов [2], если программисту не нужен полиморфизм, а только наследование и инкапсуляция. Язык Rust создаёт новую модель управления памятью, которая предотвращает типичные ошибки в языках с ручной моделью (такие как попытки доступа к преждевременно уничтоженным объектам), при этом не используя сборки мусора или автоматического подсчёта ссылок. Используемая в этих целях абстракция *lifetime* [3] (время жизни объекта) является *zero-cost abstraction* (абстракция с нулевой стоимостью), поскольку с её помощью такие моменты, как оптимальное время удаления объекта, полностью рассчитываются на этапе компиляции без влияния на производительность конечной программы, в то же время увеличивая производительность разработки за счёт сокращения этапа отладки.

Как видно по примеру с Rust, последние два пункта тесно связаны с вычислениями на этапе компиляции. Перенос решений как можно большего количества задач с конечной программы на компилятор улучшает производительность как самой программы, так и процесса отладки, потому является перспективным направлением исследований.

2 Структура разрабатываемого компилятора

Общая структура программы описана на рис. 1. Каждый компилятор имеет 3 основные фазы [4]:

1. Разбор исходного кода. Данная фаза состоит из двух процессов: лексический и синтаксический анализ. Лексический анализатор разбивает исходную программу на поток токенов (последовательность идентификаторов, ключевых слов, операторов и констант), после чего синтаксический анализатор проверяет соответствие образуемых конструкций грамматике языка. При отсутствии ошибок на этом этапе возможно формирование однозначной информации об описанных объектах и осуществляемых над ними действиях. Разрабатываемый компилятор использует связку генераторов Flex + GNU Bison для автоматизации разработки этих анализаторов, поскольку они позволяют описать грамматику языка в декларативной форме.

2. Семантический анализ. На этой стадии осуществляются такие операции, как построение промежуточного представления, сверка с таблицей символов, проверка типов, анализ потока управления, предварительная оптимизация и прочее. Основные цели: убедиться, что описанные действия верны в понятиях описанных абстракций, и определить корректные действия в понятиях генерируемой программы. Эти операции производятся с помощью таблицы символов путём анализа деревьев выражений. Каждый описанный тип данных в анализируемой программе сохраняется как объект подклассов *Type* (рис. 2). Аналогично хранятся инструкции (*Statement*), выражения (*Expression*) и прочие элементы.

3. Генерация кода программы. Действия в понятиях программы переводятся в понятия платформы, на которой будет работать целевая программа. На этой фазе могут осуществляться оптимизации, специфичные для данной ОС и/или архитектуры процессора. Традиционные компиляторы генерируют программы в ассемблерных инструкциях целевой платформы, затем используют программу-ассемблер для сборки рабочей программы. Перспективным направлением является компиляция кода в ANSI C, поскольку компиляторы языка C доступны на большом количестве платформ, а их встроенные оптимизаторы позволяют получить производительность, близкую к написанному вручную ассемблерному коду.

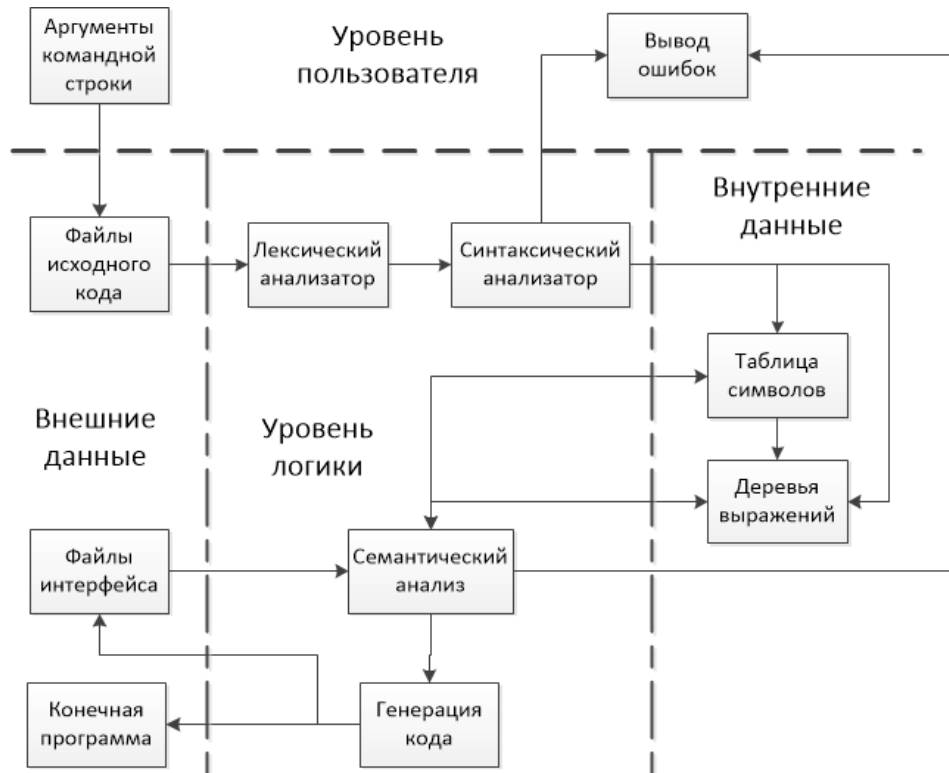


Рисунок 1 – Схема архитектуры компилятора

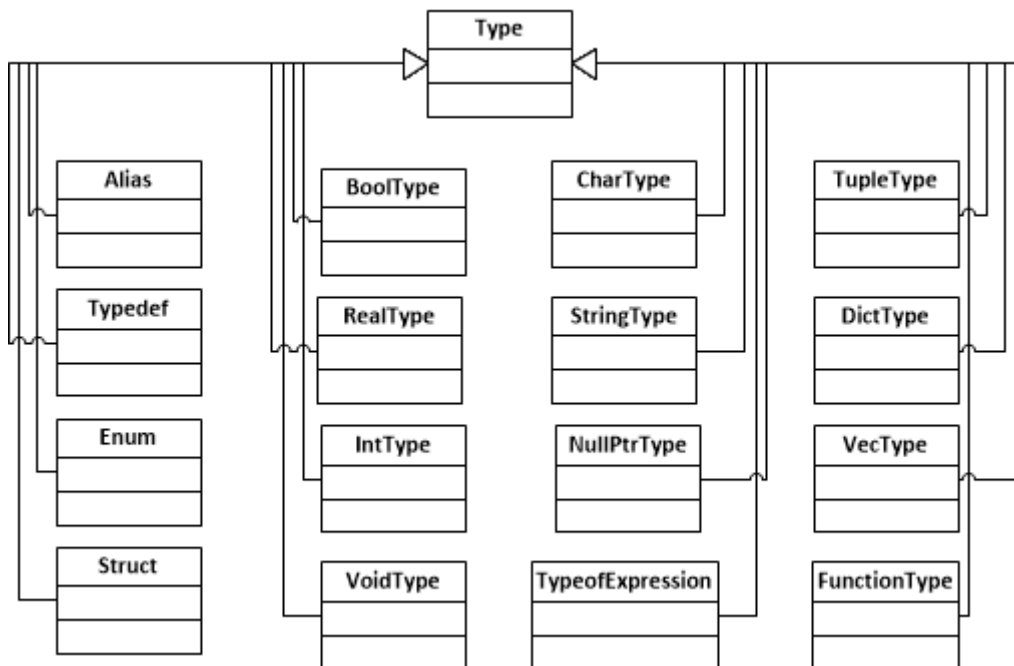


Рисунок 2 – Потомки класса Type, представляющие собой различные стандартные типы и способы построения пользовательских типов

3 Примеры конструкций проектируемого языка

На рис. 3 рассмотрен пример исходного кода. Синтаксис и многие конструкции языка являются производными от C-подобных языков, но с некоторыми структурными отличиями. Из данных примеров видны следующие:

1. Директива `import` указывает компилятору путь к модулям, символы которых необходимо добавить в глобальное пространство имён. Директива позволяет перечислять символы в иерархическом порядке, таким образом разрешая указать несколько символов в единожды указанном модуле. При импорте компилятор проходит соответствующий путь в файловой системе, пытаясь найти файл интерфейса указанного модуля. Файлы интерфейса генерируются при компиляции вместе с исходным кодом на C.

2. Типы-эnumерации (`enum`) являются алгебраическими типами данных, или помеченными объединениями (`type-tagged union`). Каждый вариант является не только некоторым значением, но ещё и сопровождающей это значение структурой данных. Все варианты могут иметь параметры разных типов. При

использовании объектов таких типов в кратком синтаксисе скрывается множество операций по проверке истинного типа объекта и компилятор гарантирует, что все операции над этими объектами типобезопасны.

3. Условные переходы, циклы и блоки кода вместо инструкций являются выражениями. Т. е. они имеют некоторое значение, вычисляемое кодом внутри них. Это позволяет: записывать некоторые операции более лаконично, сохранять чистоту внешней области видимости от временных переменных и обеспечивать константность переменных, требующих сложный процесс инициализации.

4. Цикл `for` имеет более простую форму записи, которая указывает имена индексной переменной и итератора, а также диапазон обхода без излишней информации о внутренних условиях работы цикла.

Кроме того, имеются ссылочные типы (в качестве абстракции над указателями) и чёткое разделение между типами `nullable` (которые могут иметь нулевой указатель в качестве значения) и `non-nullable`. Компилятор будет требовать проверку на `null` в случае попыток операций над объектами `nullable`-типов.

```
// Исходный текст программы начинается с директив import, добавляющих
// в таблицу символов определения из других модулей.
import std:(io, stdlib.malloc);

// Определить новый тип на основе одного предыдущего
typedef Integer = Int;

// Определение структуры
struct Point3D {
    x, y, z : Real; // Поля
}

// Эnumерация
enum Color {
    Red, Green, Blue, // Обычные варианты
    RGB(red : Int, green : Int, blue : Int) // Параметризованный
    вариант
}
// Определение функции
func main(args : Vec[String]) : Int {
    var eight = 3 + 5; // Автоматически принимает тип Int
    var nine : Real = eight + 1; // Преобразуется к Real из Int

    // Конструкция if может использоваться как выражение
    // Доступ к элементам массива происходит через "("
    // Оператор "." для доступа к свойствам
    var condInt = if (args(1).isEmpty) eight else nine;

    for (i,v in 1..10) { // аналог for (int i=0, v=1; v <= 10; i++,
v++)...
        print(i, " : ", v, "\n");
    };

    var point : Point3D = (3, 4, 0.2);
    var gray = RGB(128, 128, 128); // Тип - Color

    return 0;
}
```

Рисунок 3 – Пример исходного кода на проектируемом языке

4 Перспективы развития

Язык создан с намерением его расширения функциональными и объектно-ориентированными возможностями с целью исследования возможностей вычислений на этапе компиляции.

Среди планируемых возможностей являются такие распространённые возможности современных языков, как интерфейсы, полноценная поддержка шаблонов (в примере кода виден стандартный шаблон `Vec`, который является абстракцией над статическими и динамическими массивами), `lambda`-выражения (как минимум в форме `inline`-операций), контрактное программирование и выполнение произвольного кода на этапе компиляции.

Объектная модель, в которой в основе будут лежать интерфейсы, сходна с моделью языка Rust [5]. В языке Rust отдельно описываются структуры данных (`struct`), интерфейсы (`trait`) и реализации (`impl`), что отличается от традиционного объектно-ориентированного подхода, в котором классы объединяют в себе все три элемента. Возможность описывать реализации различных интерфейсов для различных структур данных открывает возможности для полиморфизма без явных признаков наследования («приоритет композиции вместо наследования» как один из принципов ООП подсвечивает одну из проблем этой парадигмы). При этом возможен больший фокус на статическом полиморфизме вместо динамического (компилятор может «развернуть» абстракцию и подобрать правильные операции без использования таблиц виртуальных функций, и таким образом предотвратить лишние ошибки динамического приведения типов).

Выполнение произвольного кода на этапе компиляции (`compile-time function execution`, `CTFE` [6]) также позволит повысить производительность конечной программы и обнаружить многие ошибки до её запуска. Задачи, которые можно выполнить единожды на машине разработчика, можно убрать с каждого запуска программы пользователем. Часть кода, которая требует вызова функций операционной системы для вычисления необходимых для компиляции данных, может быть обозначена специальным атрибутом. В остальных случаях компилятор может использовать анализ потока управления [7], чтобы определить возможность исполнения при компиляции конкретных участков кода. То есть ставится вопрос: требуются ли для исполнения данного кода данные от пользователя или его среды. Если нет, то в конечную программу попадает только результат его вычисления. На деле эта задача сводится к выделению в исходном коде комплекса метапрограмм, иерархически зависящих от выходных данных друг друга (рис. 5). При этом входные данные верхнего уровня прописаны в коде в качестве констант-аргументов для некоторых используемых библиотек и некоторые результаты работы среды разработчика, а выходные данные нижнего уровня напрямую попадают в конечную программу и взаимодействуют с системой пользователя без предварительных инициализаций. Такой подход также позволит очистить исполняемый файл от «мёртвого» кода и даже от «мёртвых» полей данных, которые существуют для поддержки обычно полезной функциональности, но не влияющей на работу данной конкретной программы.



Рисунок 4 – Общая схема работы будущего компилятора с возможностями исполнения кода

Модель состояний типов (`typestate`) [8] может обеспечить корректность применения

последовательностей операций в каждой точке программы на основе описанных условий.

Анализируя различные инструкции и условные переходы, компилятор может предсказать определённые наборы значений, которые переменные могут принять в отдельных ветвях программы, и предотвратить использование ошибочных значений в отдельных операциях либо неверной последовательности действий над некоторым объектом. В сочетании с контрактным программированием она даёт возможность предотвратить многие исключительные ситуации на этапе компиляции, поскольку создатели библиотек смогут установить особые ограничения на входные данные, и компилятор будет вынуждать программистов, использующих их библиотеки, выполнить соответствующие проверки или гарантировать корректность каким-либо другим образом (например, последовательностью преобразований).

Выводы

В ходе выполнения данной работы были изучены основные современные подходы к проектированию языков программирования и рассмотрена модель работы типичного компилятора. С учётом данных принципов был спроектирован C-подобный язык программирования общего назначения и компилятор для него.

Дальнейшее развитие проекта состоит в расширении системы типов и внедрении комплексных средств анализа кода для обеспечения улучшения качества процесса

разработки ПО.

Литература

1. Исполнение кода и диагностика ошибок программ на этапе компиляции. Н.М. Ткачёв, А.И. Андрияхин, Компьютерная и программная инженерия - 2015 год: - Донецк: ДонНТУ, 2015. – С.105-107.
2. Interview With Bjarne Stroustrup [электронный ресурс]. – Режим доступа: <http://www.stroustrup.com/devXinterview.html>.
3. Rust Programming Language. Lifetimes [электронный ресурс]. – Режим доступа: <https://doc.rust-lang.org/book/lifetimes.html>.
4. В.Э. Карпов "Классическая теория компиляторов", ISBN 5–230–16344–5, 2011. - С. 7-10.
5. Rust Programming Language. Traits [электронный ресурс]. – Режим доступа: <https://doc.rust-lang.org/book/traits.html>.
6. Wikipedia. Compile-time function execution [электронный ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/Compile_time_function_execution.
7. Википедия. Анализ потока управления [электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Анализ_потока_управления.
8. Wikipedia. Typestate analysis [электронный ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/Typestate_analysis.

Ткачёв Н.М., Андрияхин А.И. Проектирование и разработка компилятора C-подобного языка программирования общего назначения с поддержкой исполнения кода на этапе компиляции. В статье представлен краткий обзор современных принципов проектирования языков программирования и сформулированы требования к проектируемому языку. Спроектирована начальная итерация языка программирования общего назначения и кратко описана структура разрабатываемого компилятора. Перспективами развития языка является расширение системы типов и внедрение в компилятор комплексных средств анализа кода для предотвращения ошибок и исполнения кода на этапе компиляции.

Ключевые слова: компилятор, C, Flex, GNU Bison, генерация кода, вычисления на этапе компиляции, система типов.

Tkachev Nickolay, Andryukhin Alexander Designing a C-like general purpose programming language with CTFE capabilities and building a compiler. The article provides a brief overview of modern principles of programming language design and forms a set of requirements for a designed language. It showcases initial iteration of designed general purpose language and briefly describes architecture of its compiler. In the future language will have its type system expanded, the compiler will be extended with complex code analysis measures for error prevention and compile-time code execution.

Key words: compiler, C, Flex, GNU Bison, code generation, compile-time function execution, type system.

УДК 004.0- 519.854.001

Компьютерные расчеты и визуализации характеристик устойчивости систем: проблема центра-фокуса и гармонические функции

Филер З.Е.¹, Андрюхин А.И.²¹Кировоградский национальный педагогический университет,²Донецкий национальный технический университет

filier@ramber.ru, alexandruckin@ramber.ru

Филер З.Е., Андрюхин А.И. Компьютерные расчеты и визуализации характеристик устойчивости систем: проблема центра-фокуса и аналитические функции. Эта работа относится к известной проблеме определения параметров устойчивости систем. Теоретические обоснования решения этой задачи рассматриваются для случая центра. Решения для линейных и гармонических функций, а также однородных многочленов высших степеней обсуждаются. Результаты компьютерных расчетов приведены. В расчетах использовался пакет Mathematica.

Ключевые слова: проблема центра-фокуса, устойчивость системы, визуализация, компьютер.

Введение.

Общеизвестно влияние результатов А. Пуанкаре по качественному исследованию систем. В частности они послужили толчком для работ Ляпунова по устойчивости систем и следовательно на развитие общей теории управления, теории автоматического управления, теории систем.

С 1880 г. начался цикл работ А. Пуанкаре «О кривых, определяемых дифференциальными уравнениями». В своём «Аналитическом резюме» [1] он в 1-ом разделе «Дифференциальные уравнения» в подразделе Y он даёт анализ своих работ в этом направлении. На с. 597-598 он пишет: «... я начал свои исследования ... с изучения кривых, определяемых уравнениями вида

$$\frac{dx}{X} = \frac{dy}{Y},$$

где X, Y - ... многочлены от x и y ... [оно] привело к различению 4 видов особых точек ... Я дал [им] ... названия: 1) седла... 2) узлы ... 3) фокусы ... наподобие логарифмической спирали...4) центры, где все кривые, ... замкнуты, наподобие кривых уровня топографического плана лишь в крайне редких случаях). Далее, на с. 599, он пишет: «Имеется частный случай... .. необходимо бесконечное число условий ... это чаще всего ... доказывається, что все условия удовлетворены, ... когда ...

$$\frac{\partial X}{\partial x} + \frac{\partial Y}{\partial y} = 0 \text{ »}.$$

Здесь нет ни одной «картинки» - в докомпьютерную эру построение графиков решений было весьма трудоёмкой работой.

В нашей статье мы попытаемся обобщить эти результаты и использовать современные средства, предоставляемые ЭВМ.

1 Случай линейных функций X, Y .

Он рассматривается практически во всех пособиях по дифференциальным уравнениям [2-5]. Иногда это изложение занимает много

страниц. Заменяя уравнение $\frac{dx}{X} = \frac{dy}{Y}$ системой

уравнений $\dot{x} = X, \dot{y} = Y$, получим для

$X = ax + by, Y = cx + ey$ характеристическое уравнение $\det(A - \lambda E) = 0$. Оно является

квадратным относительно λ . След матрицы

$$A = \begin{pmatrix} a & b \\ c & e \end{pmatrix} \quad \text{Sp}A = a+e \quad \text{и её определитель}$$

$\det(A) = ae - bc$ определяют тип особой точки $O(0, 0)$.

Центр будет при чисто мнимых корнях характеристического уравнения, когда

$\text{Sp}A = a+e=0$, а $\det(A) = ae - bc = \omega^2 > 0$. Тогда x и

y будут периодическими функциями от t частоты

ω (с периодом $2\pi/\omega$). При комплексных корнях

этого уравнения, когда $\text{Sp}A \neq 0$ и

$\det(A) = ae - bc > 0$ налицо узел. Все кривые

будут подобными ввиду однородности уравнения.

В других случаях будут узлы и седла. Если

записать это уравнение в дифференциалах

$Ydx - Xdy = 0$, то сказанное Пуанкаре условие

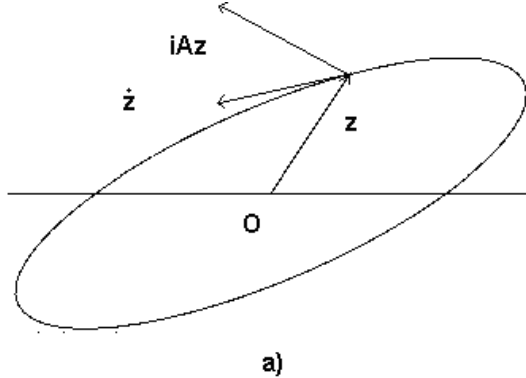
$X_y + Y_x = 0$ является условием полного

дифференциала левой части уравнения. Таким

образом, уравнение имеет полный интеграл

$cx^2 + 2axy + by^2 = f$ при постоянных действительных коэффициентах. Если $bc > a^2, b > 0, f > 0$, то оно описывает эллипс. Точка O является центром.

Переход к комплексной форме для функции $\dot{z} = (a + bi)z + iAz$ при



$\sqrt{a^2 + b^2} < A$ даёт центр при любых a, b, A . У этой формы есть простой механический смысл. Вектор касательной к траектории при $a = b = 0$ перпендикулярен к радиусу - вектору каждой её точки. Такой линией является окружность. При $a \neq 0, b \neq 0$ траекторией является эллипс.

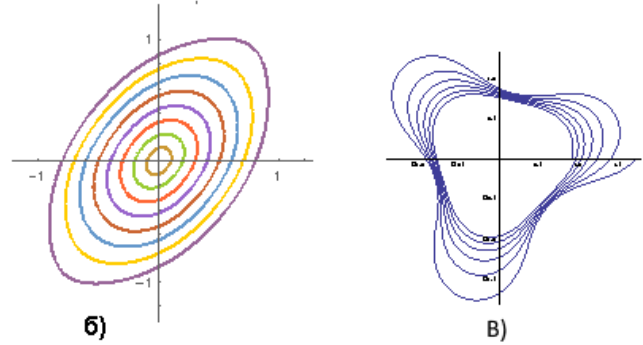


Рисунок.1--а) Механическое истолкование центра - скорость движения точки z под действием сил iAz и az ; б) Центр системы концентрических эллипсов в) фокус при $a=1, b=0.8, A=1.68, r_0=0.9, B=0.01$.

В [5, с.128-129] по поводу центра читаем: «... случай центра: при малом изменении элементов матрицы ... центр перейдёт в ... фокус. Включение этого вырожденного случая ... в основной текст ... объясняется его важностью». Периодические решения уравнения 2-го порядка изображаются замкнутыми линиями на фазовой плоскости.

2. Случай однородных многочленов высших степеней.

Для системы $\dot{x} = X, \dot{y} = Y$ удобнее в этом случае перейти к полярным координатам $x = r \cos \varphi, y = r \sin \varphi$. Если многочлены X, Y гармоничны, то есть лапласиан от них равен нулю, правые части являются линейными комбинациями косинусов и синусов аргумента $n\varphi$, умноженных на r^n . Умножая уравнения на x и y соответственно и складывая, получим систему $\dot{r} = (xX + yY)/r, \dot{\varphi} = (xY - yX)/r^2$. Она равносильна уравнению

$$\frac{dr}{d\varphi} = r \frac{Y \sin \varphi + X \cos \varphi}{Y \cos \varphi - X \sin \varphi},$$

$$X = r^n (a \cdot \cos(n\varphi) + b \cdot \sin(n\varphi)),$$

$$Y = r^n (c \cdot \cos(n\varphi) + e \cdot \sin(n\varphi))$$

Правая часть уравнения имеет точки бесконечного разрыва, в которых тригонометрическое уравнение

$Y \cos \varphi - X \sin \varphi = 0$ имеет корни. Добавив в знаменатель число A , что эквивалентно добавлению в правые части соответственно членов $-Ay$ и Ax , получим при выборе постоянной $A > \max[\sqrt{a_k^2 + b_k^2}], k = 1, 2$, где a_k, b_k - коэффициенты указанных линейных комбинаций.

В этом случае при использовании комплексной переменной $z = x + iy$ получим уравнение $\dot{z} = \alpha_n z^n + iAz$. Комплексный коэффициент $\alpha_n = a_n + ib_n$ выбирается так, чтобы выполнялись условия центра: $|\alpha_k| < A$. Соответствующее уравнение будет

$$\frac{dr}{d\varphi} = r \frac{Y \sin \varphi + X \cos \varphi}{Y \cos \varphi - X \sin \varphi + A},$$

$$X = r^n (a \cdot \cos(n\varphi) - b \cdot \sin(n\varphi)),$$

$$Y = r^n (b \cdot \cos(n\varphi) + a \cdot \sin(n\varphi))$$

Оно содержит лишь 3 произвольных постоянных a, b, A , которые могут изменяться и не мало, но согласованно.

В декартовых координатах в левой и правой точках с вертикальными касательными приходится переходить от уравнения $\frac{dy}{dx} = \frac{Y}{X}$ к

уравнению $\frac{dx}{dy} = \frac{X}{Y}$. Поэтому удобно для

построения траекторий переходить к *полярным* координатам.

На рис. 2-4 представлены центры с различными n . Нижние рисунки показывают изменение радиуса $r(\varphi) - r(0)$.

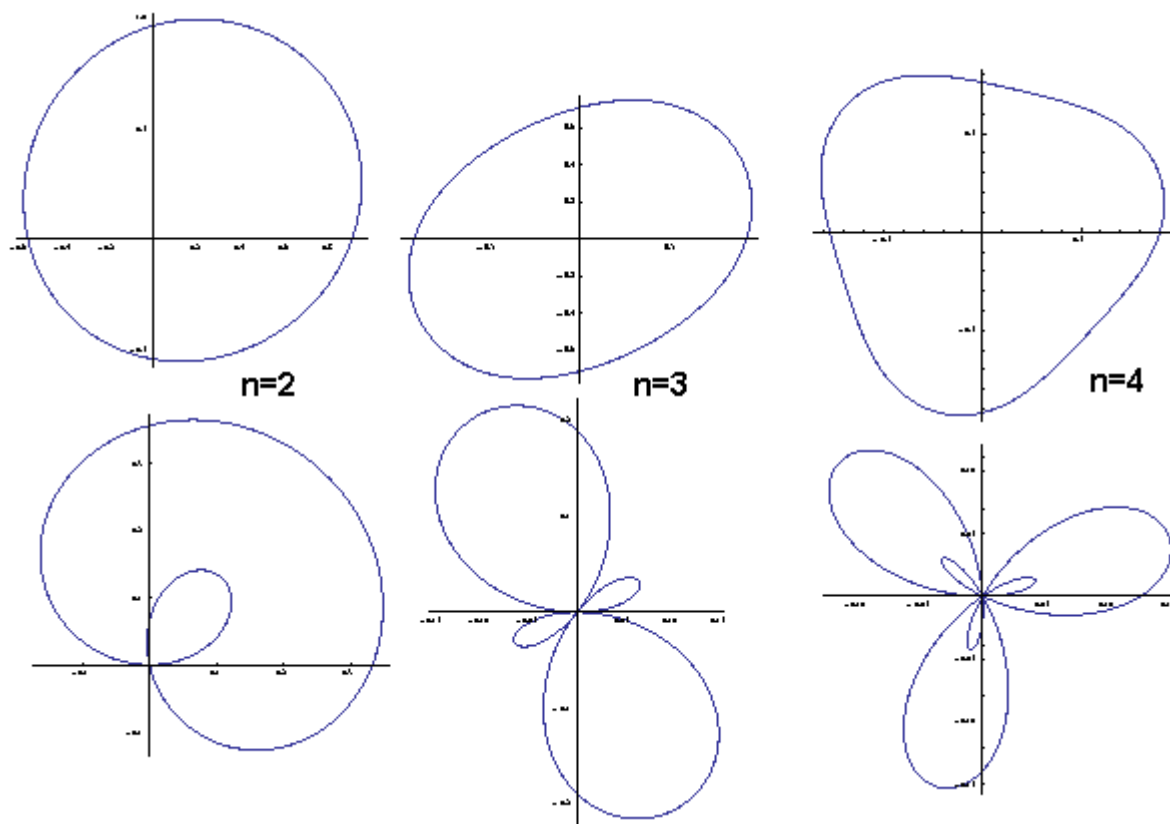


Рисунок 2-- Коэффициенты имеет значения: $a = 1, b = 8, A = 1.68$.

«Атомарные» гармонические многочлены являются «кирпичиками», из которых строятся функции, обеспечивающие появление центра.

Примерами их при $n = 3$ являются 2 конкретных многочлена 3-ей степени:
 $X_3 = x^3 - 3xy^2, Y_3 = 3x^2y - y^3$;

при $n = 4$ имеем
 $X_4 = x^4 - 6x^2y^2 + y^4, Y_4 = 4x^3y - 4xy^3$. Они являются действительными и мнимыми частями степени $z^n = (x + iy)^n$ при n , соответственно равном 3 и 4.

3. Случай гармоничных функций

Для комплексной аналитической функции $f(z) = \sum_{k=0}^{\infty} c_k z^k$ центр находится в точке O при $c_0=0$; член $c_1 z$ должен иметь вид $(a + bi)z + iAz$. Коэффициенты

$a, b, A, a_k, b_k, k \geq 2$ произвольны, но сумма модулей коэффициентов *сходящегося* ряда должна быть меньше A .

Для известных гармонических функций можно пользоваться привычными формулами, например, для функции $f(z) = e^z - 1 - z$ будем иметь центр в точке O , если взять $(a + ib)f(z) + iAz$ с достаточно большим $A \gg \sqrt{a^2 + b^2}$. Действительной частью этой функции является $X = e^x \cos(y) - 1 - x$, мнимой – функция $Y = e^x \sin(y) - y$.

Добавление в состав одного из слагаемых, которые не входят в состав гармонических членов, делает траекторию отличной от центра, ибо в числителе появляется постоянная составляющая. ведущая к росту модуля производной.

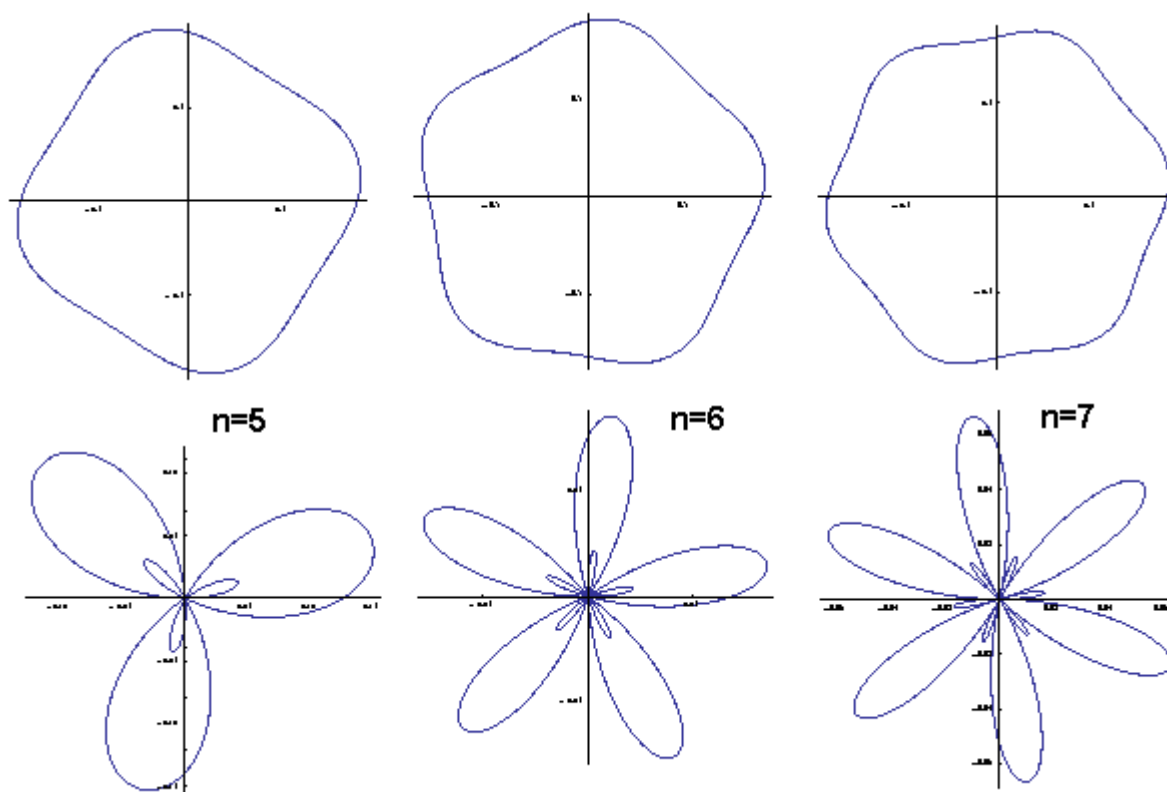


Рисунок 3а-- Центры при $n = 5,6,7$. Количество «лепестков» и «сторон» многоугольника равно $n - 1$.

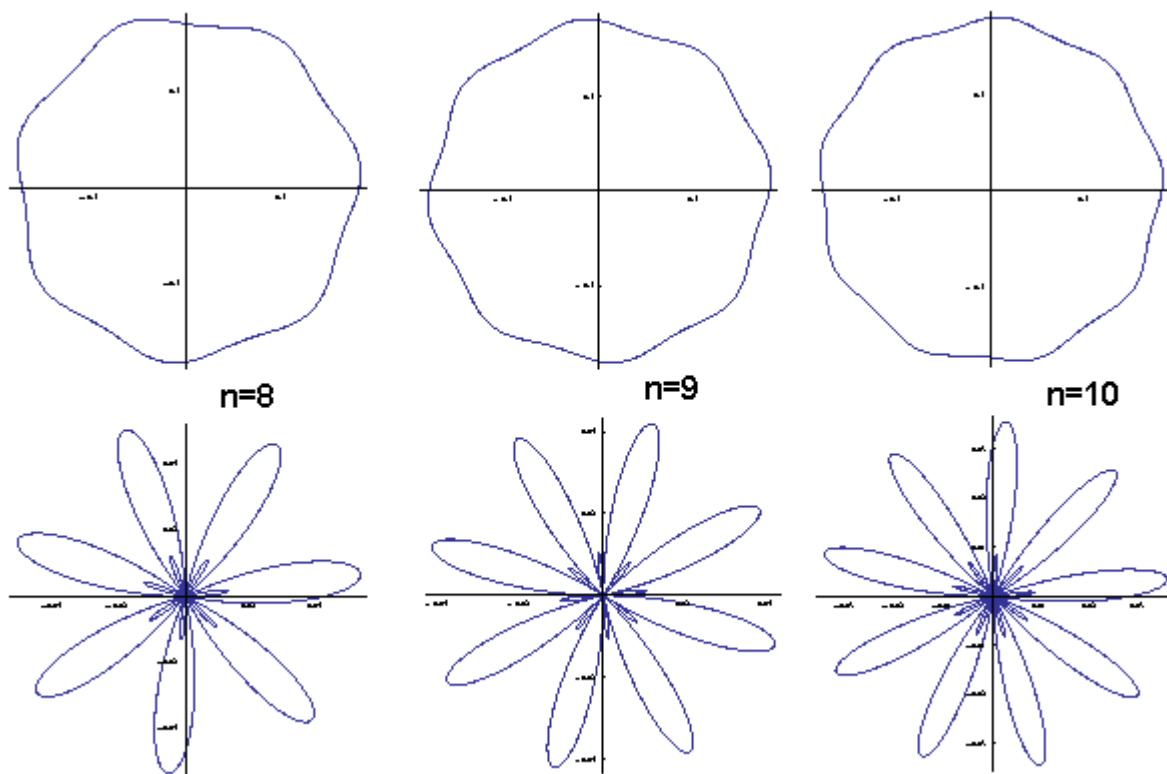


Рисунок 3б-- Центры при $n = 8,9,10$. Число «лепестков» и «сторон» многоугольника также равно $n - 1$.

В уравнении (2) необходимо взять вместо x, y выражения $r \cdot \cos \varphi, r \cdot \sin \varphi$ соответственно, а член A заменить на Ar .

На рис. 4а построены траектории центров для системы уравнений $\dot{x} = e^x \cos(y) - x - 1, \dot{y} = e^x \sin(y) - y$, а на рис. 4б – для системы $\dot{x} = \sin(x)ch(y) - x, \dot{y} = \cos(x)sh(y) - y$.

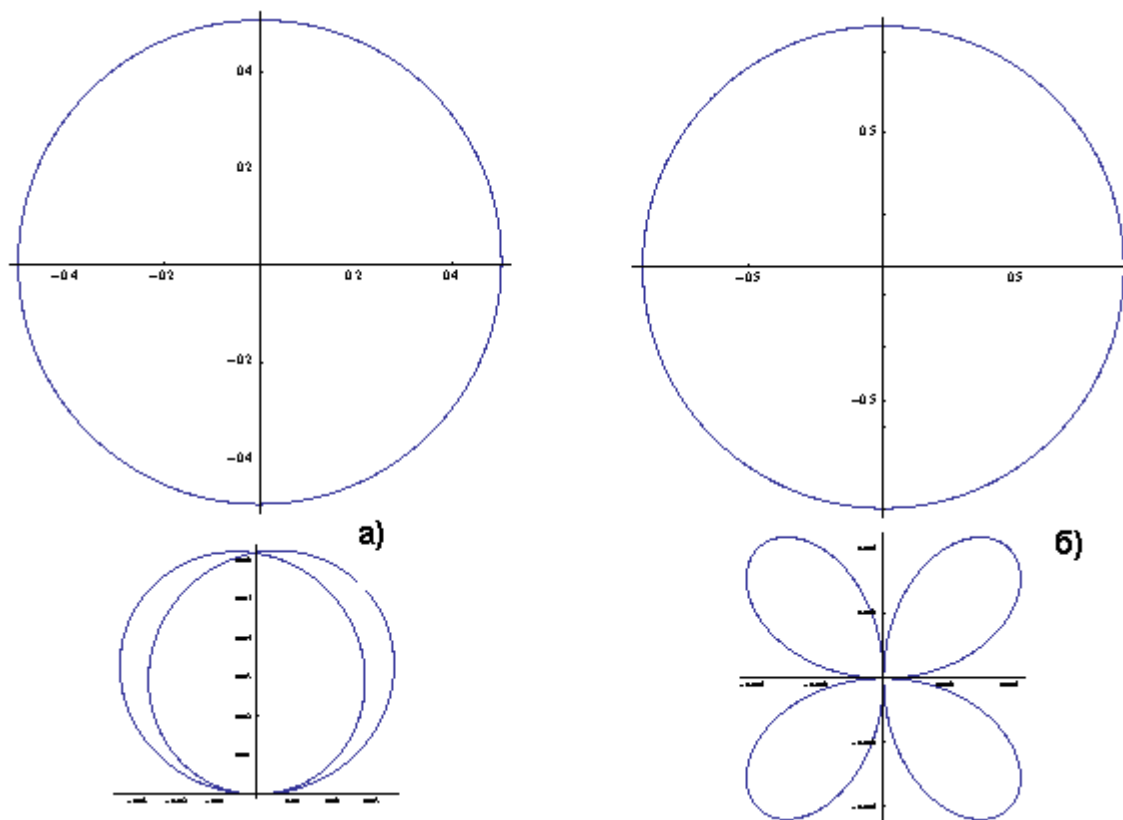


Рис.4.а) $f(z) = e^z - z - 1$; 4б) $f(z) = \sin(z) - z$.

Выводы.

1. В нашей работе берутся члены ряда из однородных многочленов, как и в других работах [3 - 5], но эти члены *гармоничны*. Может этим и объясняются сложные необходимые условия в форме связей между коэффициентами в статьях Альмухамедова и И.С. Куклеса [3, С. 87].

Между тем, в работах А.Пуанкаре есть условие центра $\frac{\partial X}{\partial x} - \frac{\partial Y}{\partial y} = 0$, вытекающее из

условий полного дифференциала, совпадающее с одним из условий Коши – Римана для гармонических функций.

2. В работе используются современные программные комплексы, позволяющие

В этих системах правые части являются действительными и мнимыми частями комплексных функций $e^z - z - 1$ и $\sin(z) - z$ соответственно. Вычитаемые x и y необходимы, чтобы члены соответствующих рядов начинались с $n = 2$. Без них особая точка O будет фокусом.

проверять и иллюстрировать полученные аналитически результаты.

Литература

1. Пуанкаре А. Избранные труды в 3 томах. Том III. Математика. Теоретическая физика. Анализ математических и естественно - научных работ Анри Пуанкаре. М.: Наука, 1974. – 772 с.
2. Пуанкаре А. О кривых, определяемых дифференциальными уравнениями. – М.-Л.: ГИТ-ТЛ, 1947. – 392 с.
3. Немыцкий В.В и Степанов В.В. Качественная теория дифференциальных уравнений. М.-Л.: ГИТ-ТЛ, 1947. - 448 с.

4. Курс обыкновенных дифференциальных уравнений/ Н.П.Еругин, Й.З.Штокало, П.С.Бондаренко, И.А.Павлюк др. – К.: Вища школа, 1974. – 472 с.

5. Понтрягин Л.С. Обыкновенные дифференциальные уравнения. – М.: ГИФ-МЛ, 1961. – 312 с.

Филер З.Е., Андрюхин А.И. Компьютерные расчеты и визуализации характеристик устойчивости систем: проблема центра-фокуса и аналитические функции Эта работа относится к известной проблеме определения параметров устойчивости систем. Теоретические обоснования решения этой задачи рассматриваются для случая центра. Решения для линейных и гармоничных функций, а также однородных многочленов высших степеней обсуждаются. Результаты компьютерных расчетов приведены. В расчетах использовался пакет Mathematica.

Ключевые слова: проблема центра-фокуса, устойчивость системы, визуализация, компьютер.

Filer Z.E., Andryukhin A.I. Computer calculations and visualization systems stability characteristics: the center-focus problem and analytical functions. This work refers to the well-known problem of determining the parameters of the systems stability. Theoretical basis of solving this problem are being considered for the center of the case. Solutions for linear and harmonic functions, as well as the homogeneous polynomials of higher degrees are discussed. computer calculation results are given. The calculations used Mathematica package.

Keywords: center-focus problem, system stability, visualization, computer.

Статья поступила в редакцию 20.11.2016
Рекомендована к публикации д-ром физ.-мат. наук С.Н. Судаковым

УДК 004.657

Методы разработки систем учета научно-технических мероприятий

Ходич Д.В., Завадская Т. В., Грищенко В.И.
Донецкий национальный технический университет,
кафедра программной инженерии
dima_khodich@mail.ru

Ходич Д.В., Грищенко В.И. Методы разработки систем учета научно-технических мероприятий. В данной статье были проанализированы существующие методы разработки программных систем учета научно-технических мероприятий. В результате анализа был выявлен самый оптимальный метод разработки программной системы. На основе этого метода была разработана система учета научно-технических мероприятий для ДонНТУ с открытым кодом. В системе реализовано генерирование отчетности по данным из БД, которая была спроектирована с учётом специфики ВУЗа.

Ключевые слова: научно-технические мероприятия, система учета мероприятий, методы разработки систем учёта, генерация отчётов, база данных, клиент-серверная архитектура.

Введение

За многие годы было проведено множество различных научно-технических мероприятий (НТМ). Учитывая то, что по каждому из мероприятий нужно оформить большое количество подготовительных документов (на начальном этапе) и множество отчетов после проведения мероприятия, в архивах университетов и различных организаций накопилось большое количество информации в бумажном виде.

Актуальность данной работы состоит в том, что система проектируется с учетом современных технологий контроля данных, обладает клиент-серверной архитектурой, что в свою очередь позволяет с лёгкостью работать с БД обычным пользователям ПК, а не программистам. Также благодаря такой архитектуре система доступна на сервере в любой момент времени через глобальную сеть Интернет с удобным и интуитивно понятным пользовательским интерфейсом.

Существует два основных метода разработки и реализации систем учёта НТМ – офлайн приложение и приложение с клиент-серверной архитектурой, структура которой представлена на рис. 1.

Первый метод подходит для организаций, в которых контроль над данными по мероприятиям и сотрудникам ведёт один человек и структура БД статична.

Второй метод подходит, как и для организаций, в которых контроль над данными ведёт один человек, так и для организаций, в которых контроль над данными и сотрудниками

ведут несколько человек. Однозначным плюсом такой архитектуры является то, что каждый пользователь, взаимодействующий с системой, обращается к актуальной версии БД по мероприятиям посредством запросов к серверу. Также в системах с такой архитектурой возможно разделение полномочий и возможностей пользователей в системе.

Целью данной работы является:

- 1) Определение оптимального метода разработки системы учёта НТМ;
- 2) Реализация программной системы и оценка ее эффективности.

. Специфика задачи

За многие годы в ДонНТУ накопилось большое количество бумажной отчетности по мероприятиям. Все эти документы время от времени приходится просматривать для подготовки отчетов за определенный период времени или годовых отчетов. Чтобы облегчить работу ответственным за проведение мероприятий было решено создать автоматизированную систему составления приказов и оформления отчетов.

В ДонНТУ ежегодно проводятся следующие виды научно-технических мероприятий НТМ:

- конференция;
- семинар;
- форум;
- симпозиум;
- олимпиада;
- презентация.

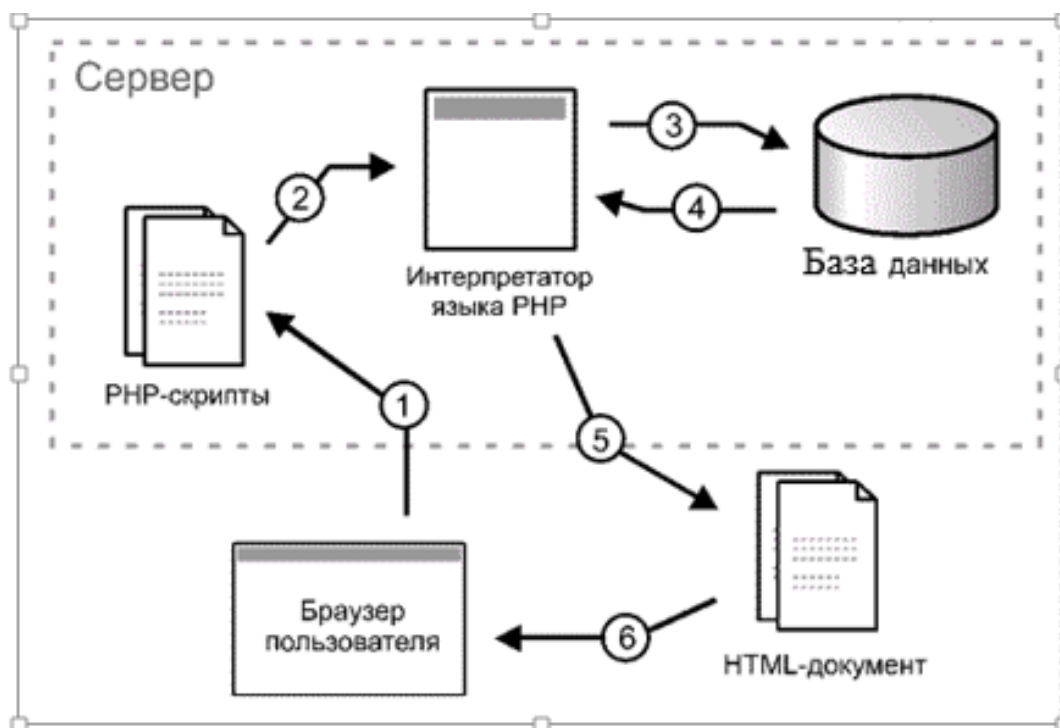


Рисунок 1 – Архитектура программной системы

Система учета мероприятий ДонНТУ является информационной системой, которая в свою очередь направлена на:

- уменьшение рутинной работы;
- сокращение времени на оформление отчетов и приказов;
- уменьшение числа возможных ошибок;
- резкое сокращение времени для обработки данных и получении нужной отчетности.

Для решения поставленной цели необходимо осуществить:

- автоматизацию ввода, контроля и загрузки данных в базу НИЧ ДонНТУ;
- автоматизацию регистрации на сайте конференций;
- формирование отчетности по заданным конференциям;

2. Анализ функциональных возможностей и методов построения существующих программных систем учета НТМ

2.1 Программная система "Учет мероприятий и семинаров"

Рассмотрим структуру системы: "Учет мероприятий и семинаров" - предназначена для комплексной автоматизации компаний, занимающихся организацией деловых мероприятий. Программа позволяет вести базы

данных клиентов. Данная программа легко и быстро настраивается под конкретные требования заказчика. Основные функции: - учет клиентов-организаций, которые участвуют в мероприятиях; - ведение базы контактов клиентов, сотрудников организаций; - учет прошедших и будущих мероприятий; - ведение календарного плана (расписание) мероприятий; - ведение базы участников мероприятий (авторов и лекторов); - ведение базы тем выступлений лекторов; - учет договоров, заключенных с клиентами; - учет сотрудников, их логинов и паролей для доступа в программу; - минимизация ошибок ввода, уменьшение времени на обработку заявки; - возможность импорта и экспорта данных [1].

2.2 Программная система "Программа контроля и управления мероприятиями"

Рассмотрим структуру системы: "Программа контроля и управления мероприятиями" включает журнал мероприятий по контролю каждого мероприятия, праздника и события. Контроль реализации мероприятий включает учет дохода от праздника или события. Поднять престиж и осуществить все самые недостижимые цели вы сможете, используя учет управленческий. Автоматизированное управление технологическими процессами позволяет контролировать все рабочие моменты [2].

Существующие аналоги программных систем учета мероприятий не подходят для ДонНТУ, исходя из того, что они не учитывают специфику ВУЗа. Цели, которые ставят перед собой аналогичные системы, не соответствуют целям нашего университета. Также база данных для учета мероприятий ДонНТУ должна иметь значительно больший объем, что поможет в дальнейшем составлять нужную отчетность. Поэтому было принято решение о разработке своей, более узкоспециализированной программной системы. Для разработки информационной системы учета НТМ потребуется использование персональных машин, которые будут ориентированы на конечного пользователя, а также администратора. Создание системы автоматизации

позволит руководителю мероприятия производить оформление документации быстрее и эффективнее

3. Проектирование и реализация собственной программной системы учета НТМ с использованием клиент-серверной архитектуры

В ходе реализации программной системы возникла необходимость создания нескольких модулей:

- модуль просмотра списка таблиц;
- модуль обновления данных в таблицах;
- модуль генерирования отчетов;
- модуль регистрации пользователей.

Модульная структура системы учета НТМ представлена на рис. 2.

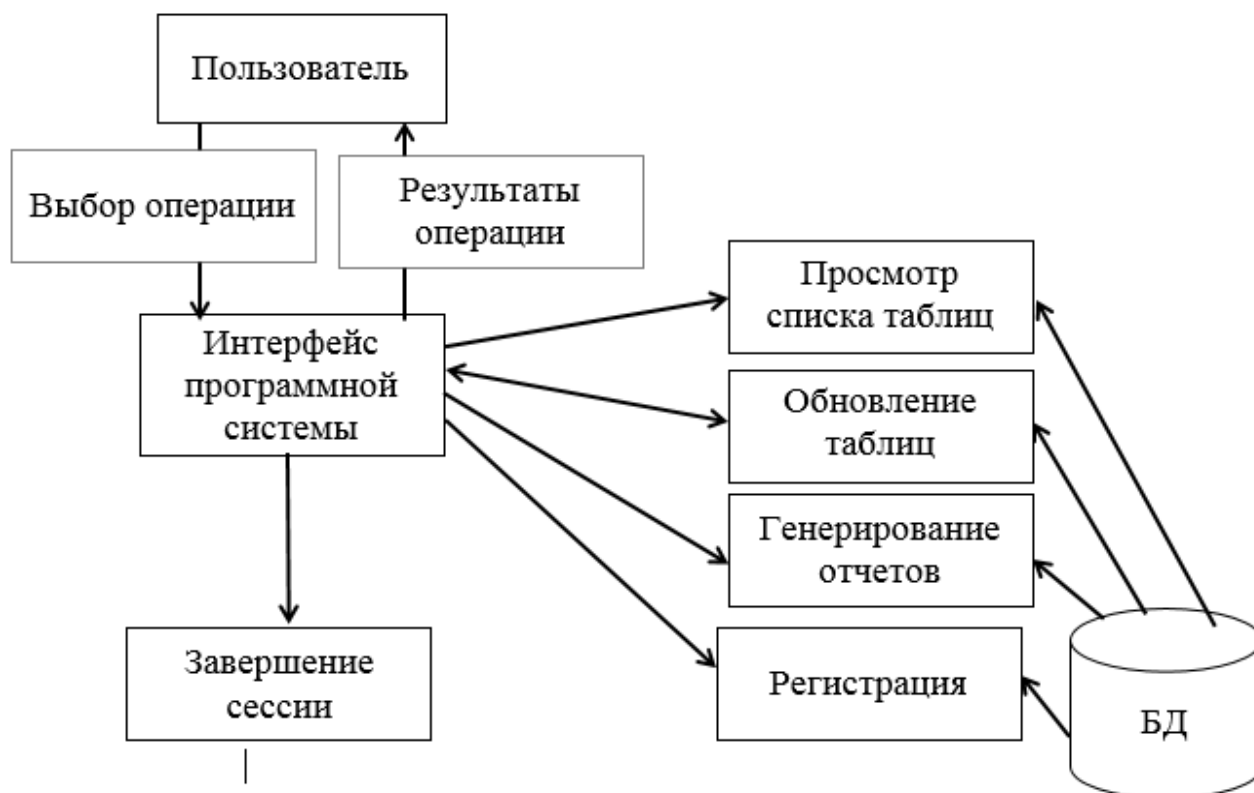


Рисунок 2 – Модульная структура системы

Система включает в себя около 25 таблиц и справочников. Для реализации системы необходима установка и настройка сервера, php, СУБД mysql, phpmyadmin. На web-сервере располагается база данных для мероприятий.

Для создания базы данных использовалась утилита phpmyadmin, которая располагается на сервере. Система для учета участия в мероприятиях состоит из двух частей: серверной и клиентской.

Скрипты, расположенные на сервере, предназначены для получения данных и для обработки запросов (добавления данных, обновления данных, для удаления данных и для редактирования) [3,4].

На данном этапе спроектирована и полностью реализована программная система учёта НТМ. Главная форма представлена на рис. 3.

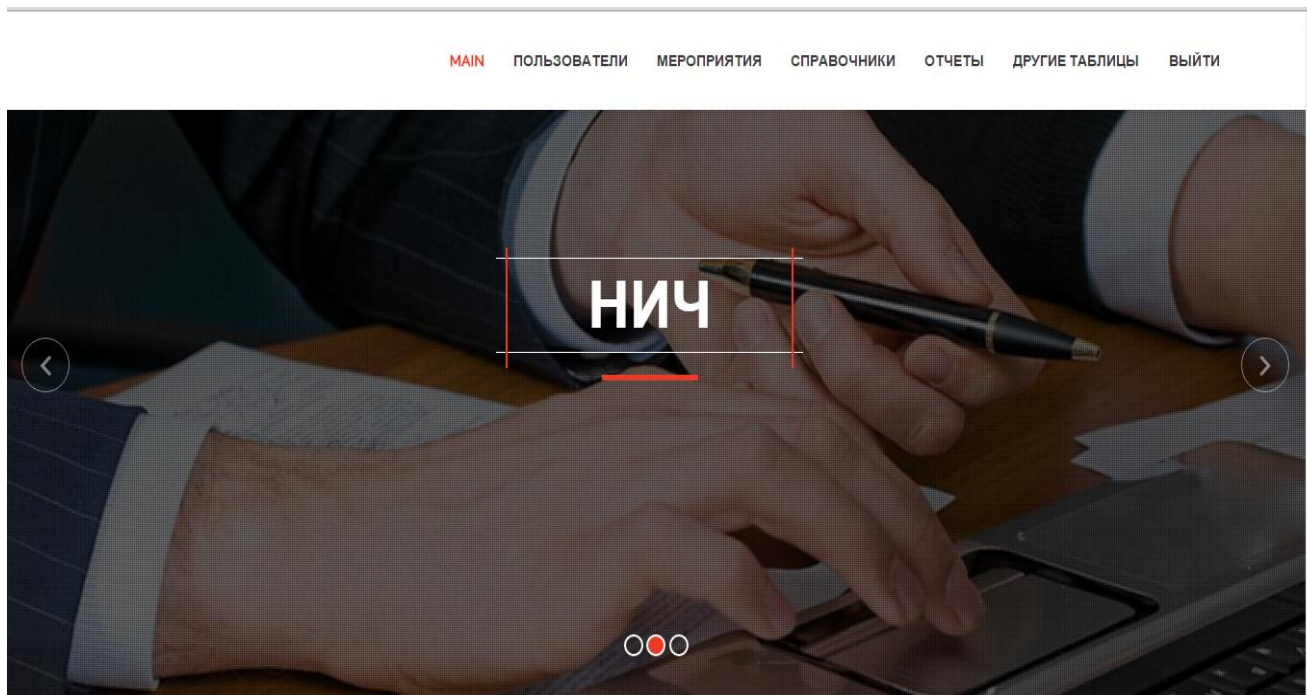


Рисунок 3 - Главная форма

4. Порядок работы с программной системой.

Начальник НИЧ заполняет сведения о заместителях декана по науке факультетов ДонНТУ: ФИО, должность, контактные телефон и эл. почта до 1 декабря текущего года.

1 декабря осуществляется напоминающая рассылка на адрес эл. почты заместителям декана по науке о необходимости подтвердить авторизацию для новых пользователей и внести сведения в факультетский план НТМ на следующий календарный год до 20 декабря.

С 20 по 31 декабря начальник НИЧ проверяет корректность ввода сведений о НТМ в общем плане мероприятий. 31 декабря, проверенный план НТМ получает статус утвержденного, после подписания печатного варианта проректором по научной работе ДонНТУ.

10 января осуществляется рассылка на адрес эл. почты ответственных секретарей оргкомитетов о необходимости подтвердить авторизацию для новых пользователей и необходимости ознакомления с порядком работы с системой (внесения сведений, подготовки приказа и отчета и т.д.).

За 20 дней до даты проведения мероприятия осуществляется рассылка на адрес эл. почты ответственного секретаря соответствующего оргкомитета о необходимости подготовки приказа, о возможности внести окончательные изменения в сведения о мероприятии, включая его отмену.

За 10 дней до даты проведения мероприятия осуществляется рассылка на адрес эл.

почты ответственного секретаря соответствующего оргкомитета о необходимости публикации пресс-релиза и размещения программы (положения) мероприятия.

Через 5 дней после даты проведения мероприятия осуществляется рассылка на адрес эл. почты ответственного секретаря соответствующего оргкомитета о необходимости публикации пост-релиза о проведенном мероприятии.

Через 10 дней после даты проведения мероприятия осуществляется рассылка на адрес эл. почты ответственного секретаря соответствующего оргкомитета о необходимости сдачи отчета о проведенном мероприятии.

В случае отсутствия сведений о подготовке приказа (за 17 дней), публикации пресс-релиза (за 8 дней), программы или положения (за 8 дней), пост-релиза (через 7 дней), отчета (через 13 дней) начальнику НИЧ приходит уведомление на эл. почту.

По требованию начальника НИЧ формируется отчет о проведенных мероприятиях за месяц, квартал, полугодие, год или за выбранный период времени. На основе введенных данных выполняется расчет показателей НТМ.

Выводы

В результате анализа методов разработки программных систем учёта НТМ было выявлено, что лучшим методом является создание клиент-серверного приложения. С использованием данного метода была разработана программная система. Система учёта НТМ имеет следующие достоинства:

- система позволяет упростить работу с мероприятиями, проводимыми в ДонНТУ;

- применение системы значительно ускоряет процесс составления документации по данным мероприятиям, а также повышает качество составляемой отчетности;

- система позволяет ответственным за проведение конкретного мероприятия, т. е. заместителям деканов по науке на каждом факультете, формировать документацию в автоматизированном виде, что также позволит избежать множества ошибок.

Как перспективу развития данной системы можно предоставить возможность генерирования отчетности в системе посредством составления расписания получения отчетов конкретным пользователем. Вторым перспективным направлением развития системы можно назвать расширение путём добавления возможности редактирования отчетов в браузере.

Литература

1. Конфигурация "Учет мероприятий и семинаров"//Простой софт программы для дома и офиса. [Электронный ресурс]. – Режим доступа: <http://prostoysoft.ru/Events.htm>.
2. Учет мероприятий // Автоматизация работы компании по организации и проведению мероприятий. [Электронный ресурс]. – Режим доступа: http://usu.kz/uchet_meroprivatiy.php.
3. Осипов, Д. Л. InterBase и Delphi. Клиент-серверные базы данных / Д.Л. Осипов. - М.: ДМК Пресс, 2015. - 536 с.
4. Растринин, Л.А. Адаптация сложных систем. Методы и приложения / Л.А. Растринин. - Москва: РГГУ, 1981. - 501 с.
5. ИУСМКМ - 2016: VII Международная научно-техническая конференция, 26 мая 2016: - Донецк: ДонНТУ, 2016. – С. 167-172.

Ходич Д.В., Завадская Т. В., Грищенко В.И. Методы разработки систем учета научно-технических мероприятий. В данной статье были проанализированы существующие методы разработки программных систем учета НТМ. В результате анализа был выявлен самый оптимальный метод разработки программной системы. На основе этого метода была разработана система учета НТМ для ДонНТУ с открытым кодом. В системе реализовано генерирование отчетности по данным из БД, которая была спроектирована с учётом специфики ВУЗа.

Ключевые слова: научно-технические мероприятия, система учета мероприятий, методы разработки систем учёта, генерация отчётов, база данных, клиент-серверная архитектура.

Khodich Dima, Grishenko Viktor. Methods of development of accounting systems of scientific and technical events. The analysis has been identified the most optimal method for the development of a software system. On the basis of this method, the system was developed STE accounting for DonNTU open source. The system is implemented to generate reports on the data from the database, which was designed taking into account the specifics of the university.

Key words: STE, accounting system of events, methods for developing accounting systems STE, reports generation, database, client-server architecture.

Статья поступила в редакцию 20.11.2016
Рекомендована к публикации д-ром техн. наук В.Н. Павлышом

CONTENT

Andruckin A.I Interval estimation of parameters of chaotic chips thresholds.....	4
Anokhina I.Y. Development of a scoring model using logistic regression and ROC methods – analysis.....	13
Belkov D.V., Edemskaya E.N. The traffic formed system based self-organized criticality....	22
Belkov D.V., Edemskaya E.N. The statistical analysis of the traffic traces for wireless network.....	29
Vinogradov G.P., Kuznecov V.N. Consistent optimization in control problems of the evolution of knowledge-intensive enterprises.....	35
Dorodnykh N.O., Yurin A.Yu. Building the rule knowledge bases on the basis of UML models.....	44
P. Kuznetsov, A. Judin, S. Maslo, V. Kliukovkin. Mathematical analysis of appearance probability of blackouts.....	51
Levchenko D., Svitlychna V., Babich K. Development of the risk-based simulation model to analyze dynamics of the income and consumption of raw materials.....	57
Malcheva R.V., Zavadskaya T.V. Simulation of the internal operations of the processing elements.....	65
Skorokhod S. V., Kasyanov O. A. The experimental studies of the jpw1 tools ability to correct burst errors.....	72
Tolstykh V.K., Pshenichny K.A Numerical simulation and process control of heat transfer in counterflow heat exchangers.....	80
Tkachev N., Andryukhin A. Designing a C-like general purpose programming language with CTFE capabilities and building a compiler.....	86
Filer Z.E., Andryukhin AI Computer calculations and visualization systems stability characteristics: the center-focus problem and analytical functions.	92
Khodich Dima, Zavadskaya T.V., Grishenko Viktor. Methods of development of accounting systems of scientific and technical events.....	98

Научное издание

ИНФОРМАТИКА И КИБЕРНЕТИКА

Научный журнал

(на русском, украинском, английском языках)

№ 3(5)-2016

Ответственный за выпуск А.И. Андрюхин

Технические редакторы Т.В. Завадская, А.В. Звягинцева

Компьютерная верстка Д.А. Филипишин

Подписано к печати 26.12.2016. Формат 60×84 1/8. Бумага офсетная. Печать ризографическая.
Усл. печ. лист. 10. Уч.- изд. лист. 7,1. Тираж 100 экз.

Адрес редакции: ДНР, 83001, г. Донецк, ул. Артема, 58, ГВУЗ «ДонНТУ», 5-й учебный корпус,
к. 425. Тел.: +38 (062) 301-08-51 E-mail: infcyb.donntu@yandex.ru, URL: <http://infcyb.donntu.org>

Издатель Государственное высшее учебное заведение «Донецкий национальный технический
университет» ДНР, 83001, г. Донецк, ул. Артема, 58. Тел.: +38 (062) 301-08-67, +38 (062) 301-
09-67 Свидетельство о государственной регистрации субъекта издательского дела: серия ДК
№2982 от 21.09.2007

Отпечатано ООО фирма «ДРУК-ИНФО» ЛУН, 83001, г. Донецк, ул. Артема, 58, к. 113, тел.: +38
(062) 335-64-55