



**межвузовский сборник
научных трудов**

**МАТЕМАТИЧЕСКОЕ
И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ**



2019

межвузовский сборник научных трудов

**МАТЕМАТИЧЕСКОЕ
И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ**

РГРТУ, 2019

Математическое и программное обеспечение вычислительных систем: Межвуз. сб. науч. тр. / Под ред. А.Н. Пылькина. – Рязань: Издательство ИП Коняхин А. В. (Book Jet), 2019. – 120 с.

ISBN 978-5-6042175-7-3

Представлены статьи, посвященные различным аспектам разработки программных средств ЭВМ, вычислительных систем и сетей, вопросам математического моделирования, методам обработки информации.

Предназначен для научно-педагогических работников вузов, инженеров, аспирантов и студентов старших курсов.

Авторская позиция и стилистические особенности публикаций полностью сохранены.

Редакционная коллегия

д-р техн. наук, проф., засл. работник высшей школы РФ А.Н. Пылкин (отв. редактор, Рязанский государственный радиотехнический университет); д-р техн. наук, проф., Е.Е. Ковшов (Московский государственный технологический университет «Станкин»); д-р техн. наук, проф. И.А. Цветков (Рязанский государственный радиотехнический университет); д-р техн. наук, проф. К.А. Майков (МГТУ им. Н.Э. Баумана); д-р техн. наук, проф. В.В. Белов (Рязанский государственный радиотехнический университет); д-р техн. наук, проф. В.В. Золотарев (Институт космических исследований РАН, г. Москва); д-р техн. наук, проф. И.Ю. Каширин (Рязанский государственный радиотехнический университет); д-р техн. наук, профессор В.Н. Малыш («Российская академия народного хозяйства и государственной службы при Президенте Российской Федерации» г. Липецк); И.А. Благодарова (Рязанский государственный радиотехнический университет).

Рецензент: к.т.н., доцент, зав. каф. «Информатики, информационных технологий и защиты информации» Липецкого государственного педагогического института Скуднев Д.М.

ISBN 978-5-6042175-7-3

© Коллектив авторов, 2019

© ИП Коняхин А.В. (Book Jet), 2019

ВВЕДЕНИЕ

Межвузовский сборник научных трудов содержит результаты научных исследований и разработок по следующим направлениям:

- программное обеспечение вычислительных систем, новые информационные технологии;
- прикладная математика, теория информации, искусственный интеллект;
- ЭВМ в системах обработки, управления и обучения;
- автоматизированное проектирование аппаратных средств вычислительных систем;
- информационные технологии в экономических и социальных системах.

Сборник сформирован на основе статей, в которых рассматриваются различные аспекты разработки программных средств ЭВМ, вычислительных систем и сетей, включая вопросы автоматизации проектирования, теории обработки информации и математического моделирования, и предназначен для студентов, аспирантов и преподавателей технических вузов и научных работников.

Материалы для сборника предоставлены сотрудниками:

- Рязанского государственного радиотехнического университета.

УДК 519.725**Овечкин Г.В., Овечкин П.В., Чан Т.З.****ВЫСОКОСКОРОСТНОЙ МНОГОПороГОВЫЙ ДЕКОДЕР
ДЛЯ ВОССТАНОВЛЕНИЯ СТИРАНИЙ
НА БАЗЕ СОВРЕМЕННЫХ ГРАФИЧЕСКИХ ПРОЦЕССОРОВ****Рязанский государственный радиотехнический университет,
г. Рязань**

Для исправления ошибок, возникающих при передаче цифровых данных по каналам с шумами, обычно применяются помехоустойчивые коды. При этом для высокоскоростных каналов связи необходимо использовать как можно более эффективные и при этом вычислительно простые алгоритмы кодирования и декодирования. Обзор наиболее перспективных методов кодирования по критерию «эффективность-производительность» показал, что одними из лучших для высокоскоростных каналов спутниковой связи являются многопороговые декодеры (МПД) самоортогональных кодов (СОК) [1, 2]. Данные декодеры, являясь модификацией простейшего порогового декодера Мессе, позволяют декодировать даже очень длинные коды с линейной от длины кода сложностью реализации.

В десятках публикаций по МПД рассматривается эффективность его применения в двоичном симметричном или гауссовском каналах. Интерес для техники связи также представляют каналы со стираниями, использование которых позволяет существенно упростить процесс декодирования по сравнению со случаем исправления ошибок. Для такого канала существуют версии МПД алгоритмов [3], которые обеспечивают восстановление стираний с эффективностью, близкой к эффективности оптимального декодера, и сохраняющих при этом минимально возможную линейную сложность реализации.

В последнее время внимание специалистов по телекоммуникациям все больше уделяется программно-ориентированным системам (SDR) [4], в которых значительная часть цифровой обработки сигналов выполняется на обычном персональном компьютере, имеющем в своем составе многоядерный центральный или графический процессор. В таких системах для обеспечения необходимой скорости декодирования информации используются программные версии декодеров, обеспечивающие высокоскоростное и эффективное исправление ошибок. В статье рассмотрены аспекты программной реализации символьных МПД для восстановления стираний с использованием графических процессоров.

Многопороговые декодеры для исправления стираний

Символьный МПД является простейшим декодером мажоритарного типа, позволяющим декодировать недвоичные СОК [5]. Схема кодера для простого недвоичного СОК с одной информационной и одной прове-

рочной ветвями представлена на рис. 1. Отметим, что кодер состоит только из регистра сдвига и сумматоров по модулю q . Данные после кодера передаются по q -ичному симметричному каналу, в котором символы либо передаются правильно, либо «стираются» с вероятностью P_s . Схема символьного МПД, восстанавливающего стирания, представлена на рис. 2.

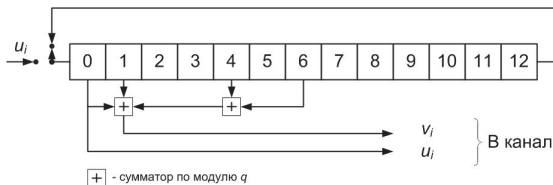


Рис. 1. Схема кодера блокового не двоичного СОК



Рис. 2. Схема блокового символьного МПД

При поступлении данных на вход декодера стертые символы информационного регистра отмечаются признаком стирания, остальные же символы информационного регистра гарантированно не содержат ошибок. Каждая ячейка синдромного регистра состоит из двух полей. В первом поле содержится разность по модулю q между принятым из канала проверочным символом и проверочным символом, полученным на основе принятых из канала информационных символов, а во втором поле содержится количество стертых символов, которые были использованы при вычислениях значения первого поля. Работа порогового элемента ($qПЭ$) декодера заключается в следующем. Если декодируемый символ информационного регистра отмечен признаком стирания, то анализируются поля с количеством стираний соответствующих ему элементов синдромного регистра. Если одна из ячеек синдромного регистра содержит только одно стирание, то, очевидно, что это стирание вызвано декодируемым информационным символом. Соответственно стирание в информационном регистре заменяется значением ошибки E ячейки синдромного регистра с

одним стиранием, также на величину E корректируются все значения символов синдромного регистра, связанных с декодируемым информационным символом, и на 1 уменьшается количество стираний в них.

Программная реализация МПД

При высокоскоростной программной реализации МПД большое внимание должно уделяться возможности распараллеливания вычислений. На сегодняшний день уже почти все компьютеры потребительского класса, а также современные мобильные телефоны оборудованы многоядерными центральными процессорами. Однако при программной реализации символьных МПД для исправления стираний наиболее быстрые программные версии декодеров обеспечивают скорость только порядка 15 МБайт/с при использовании 6-ти ядер процессора Core i7 для одновременной параллельной обработки шести принятых блоков.

Перспективным направлением для ускорения работы программных версий МПД является их реализация на графических процессорах (GPU) с использованием архитектуры CUDA и языка CUDA C [8]. Отличительной особенностью применения GPU по сравнению с CPU является то, что современные GPU имеют тысячи ядер (современные CPU не более 16), и это позволяет параллельно выполнять несколько тысяч однотипных операций. Одним из основных способов создания быстродействующего декодера на CUDA является размещение в разделяемой и константной памяти всех используемых кодером и декодером массивов и обеспечение бесконфликтного доступа к элементам таких массивов, а также одновременное декодирование нескольких символов внутри одного принятого из канала сообщения используемыми потоками GPU.

В результате выполненных работ для двоичного СОК с кодовой скоростью $R = 1/2$, длиной блока $n = 1600$ символов и минимальным кодовым расстоянием $d = 5$ (с проверками, расположенными в позициях 0, 109, 295 и 372) на базе архитектуры CUDA был разработан блочный символьный МПД, исправляющий стирания, который использует только регистровые переменные и всего 4,7 Кб разделяемой памяти, которая отводится на хранение данных информационного и синдромного регистров МПД. Это позволяет на каждом из 10 мультипроцессоров

выполнять одновременное декодирование $N_{bl} = \left\lfloor \frac{sharedPerSMM}{sharedPerMTDBlock} \right\rfloor = \left\lfloor \frac{96KB}{4,7KB} \right\rfloor = 20$ принятых из канала блоков, таким образом, суммарно выполняется одновременное декодирование $SMMsCount \cdot N_{bl} = 10 \cdot 20 = 200$ принятых сообщений. На рис. 3 представлена предлагаемая схема использования SMM.

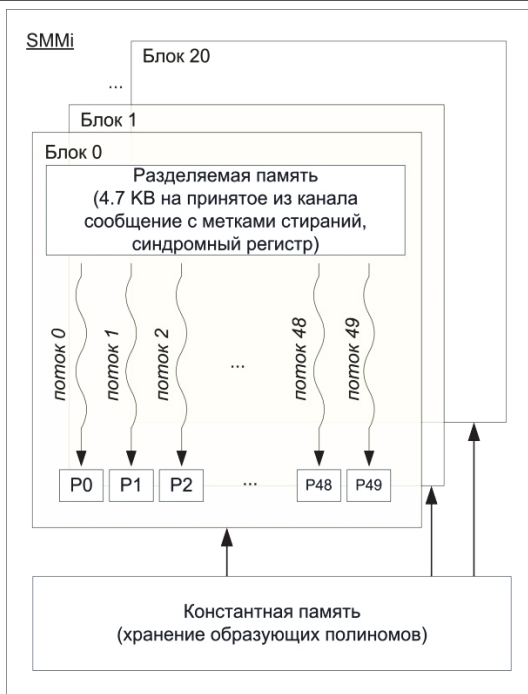


Рис. 3. Схема использования SMM

Для увеличения скорости декодирования используемые СОК были подобраны так, что позиции проверок отстоят друг от друга не менее чем на 50 позиций. Это позволяет при декодировании одного блока использовать 50 потоков для одновременного исправления 50 символов с бесконфликтным обращением и записью в разделяемую память. Схема символьного МПД с кодовой скоростью $R = 1/2$, длиной блока $n = 1600$ символов и минимальным кодовым расстоянием $d = 5$ (с проверками, расположенными в позициях 0, 109, 295 и 372) представлена на рис. 4. На первом такте потоки 0, 1, ..., 49 отвечают за исправление 750, 751, ..., 799 символов информационного регистра. При этом, благодаря синхронизации потоков, всегда происходит обращение только к 50 последовательным символам разделяемой памяти, что не вызывает конфликтов банков памяти. На следующем такте используемые 50 потоков выполняют параллельное исправление 700, 701, ..., 749 символов информационного регистра и т.д. Таким образом, поток 0 выполняет декодирование 799, 749, 699, ..., 99, 49 символов принятого из канала сообщения, поток 1 – 798, 748, 698, ..., 98, 48, а 49-й поток – 750, 700, 650, ..., 50 и 0 символов блока МПД.

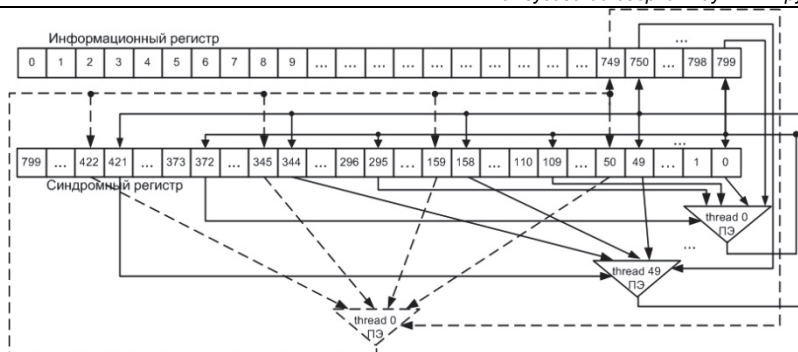


Рис. 4. Схема МПД с использованием 50 параллельных потоков

Как следует из таблицы, в которой представлены результаты оценивания скорости работы декодеров, использование предложенных приемов обеспечило общую скорость декодирования всей системы 270 МБайт/с (более 2 Гбит/с) для кода с $R = 1/2$, длиной блока 1600 символов, $d = 5$ и 20 итераций декодирования. При реализации МПД использована графическая карта NVIDIA GeForce GTX 1060. Для кода длины 4000 символов, $d = 9$ и 20 итераций МПД обеспечивает скорость исправления стираний 150 МБайт/с. Таким образом, программные версии МПД можно использовать в высокоскоростных программно-ориентированных системах на базе графических процессоров.

Таблица. Результаты оценивания быстродействия декодеров

Параметры кода и декодера	Количество одновременно декодируемых блоков	Скорость декодирования
$d = 5, n = 1600,$ $it = 20$	200	270 МБайт/с
$d = 7, n = 2200,$ $it = 20$	140	225 МБайт/с
$d = 9, n = 4000,$ $it = 20$	80	150 МБайт/с

Заключение

Рассмотрены особенности программной реализации МПД с использованием GPU. Показано, что разработанные декодеры способны обеспечить обработку входного потока со скоростью от 150 до 270 МБайт/с при использовании GPU GTX 1060. Такая скорость позволяет использовать МПД в высокоскоростных программно-ориентированных системах.

Большой объем дополнительной информации по многопороговому декодированию можно найти на наших специализированных web-сайтах www.mtdbest.ru и www.mtdbest.iki.rssi.ru. Работа выполнена при финансовой поддержке РФФИ (грант 18-07-00525).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Зубарев Ю.Б., Овечкин Г.В. Помехоустойчивое кодирование в цифровых системах передачи данных // Электросвязь. М., 2008. № 12. С. 2-11.
2. Zolotarev V.V., Zubarev Ju.B., Ovechkin G.V., Optimization Coding Theory and Multithreshold Algorithms, Geneva, ITU, 2015, 159 p.
3. Grinchenco N., Gromov A., Ovechkin G. Improving performance of multithreshold decoder over binary erasure channel // 2017 6th Mediter-ranean Conference on Embedded Computing, MECO 2017 – Including ECYPS 2017.
4. Software defined radio: architectures, systems, and functions. Dillinger, Madani, Alonistioti. Wiley, 2003. 454 p.
5. Zolotarev V.V., Ovechkin G.V. Efficient Multithreshold Decoding of Nonbinary Codes. ISSN 1064-2269, Journal of Communications Technology and Electronics, 2010. Vol. 55, No. 3, P. 302-306.

УДК 004:89

Бубнов А.А., Бубнов С.А.

О ВОЗМОЖНОСТИ ИСПОЛЬЗОВАНИЯ ДСМ-МЕТОДА ДЛЯ ОЦЕНКИ НАДЁЖНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ НА ЭТАПЕ ТЕСТИРОВАНИЯ

Рязанский государственный радиотехнический университет,
г. Рязань

Тестирование программного обеспечения и возможность оценки его надёжности с применением ДСМ-метода

Процесс тестирования программного обеспечения направлен на выявление дефектов в нём, в результате чего накапливается достаточно большой объём информации относительно количества и характеристик обнаруженных дефектов. Поэтому задача анализа результатов тестирования является весьма важной. Решение этой задачи в условиях современных методик разработки и тестирования значительно упрощается, если процесс тестирования поддерживается специальной баг-трекинговой системой (системой хранения информации о дефектах).

Стандартное описание дефекта включает в себя, большей частью, следующую информацию:

- идентификационный номер дефекта;
- название продукта;
- название компонента продукта;
- кому назначено устранение дефекта;
- статус дефекта;
- значимость дефекта;
- решение по устранению дефекта;
- краткое описание дефекта.

Следует отметить, что осуществление прогнозирования относительно количества дефектов и их атрибутов в процессе тестирования позволит значительно повысить качественные показатели как самого программного продукта, так и работ по его валидации.

Существует достаточно много методов и моделей, которые позволяют с определенной достоверностью реализовывать упомянутые выше прогнозы [2]. Однако большая часть моделей не позволяет локализовать дефект, соотнеся его с определённым участком кода.

В этой статье предлагается использовать для решения задачи прогнозирования дефектов и их характеристик, выявляемых в процессе тестирования программного обеспечения, с применением ДСМ-метода, а конкретно – его теоретико-множественной интерпретации [1].

ДСМ-метод позволяет автоматически порождать гипотезы, его автором является В.К. Финн [6]. Выражаясь более точно, ДСМ-метод обнаруживает закономерности в данных при помощи индукции Д.С. Милля [3], порождает правила устранения неопределённости в данных с использованием фальсификации К. Поппера [4], формирует предсказания с помощью рассуждения по аналогии и оценивает правдоподобие порождённых гипотез с помощью абдукции Ч.С. Пирса [5]. Таким образом, ДСМ-система позволяет обнаружить связь между структурой объекта и его поведением, где структура определяет поведение (иначе – речь идёт об обратном ДСМ-методе).

Суть ДСМ-метода в контексте рассматриваемой интерпретации заключается в следующем [1].

Элементы структуры объекта называются атомами. Элементы поведения – целевые свойства. Исследуемый объект представляется в виде конечного множества атомов. Объект может обладать (или не обладать) некоторым множеством целевых свойств. Наличие или отсутствие целевых свойств обуславливается соответствующими фрагментами структуры, т.е. конечными множествами атомов (см. рис. 1).

Интеллектуальная система, работающая на основе ДСМ-метода, называется ДСМ-системой, которая решает две основные задачи:

1. Формирование гипотез о возможных причинах наличия (отсутствия) свойств у объектов (с помощью формализованных индуктивных рассуждений);

2. Формирование гипотез о наличии (отсутствии) свойств у тех объектов, для которых это было неизвестно (посредством аналогий).

Касательно прогнозирования дефектов и процесса тестирования можно сделать следующие замечания относительно варианта применения ДСМ-метода.

Под *объектом* будем понимать *тестируемый модуль*, состоящий из *фрагментов кода*, называемых иначе *атомами*. *Фрагментом структуры* объекта (модуля), который определяет поведение системы, будем считать *дефект*. *Свойством объекта* предлагается считать *сбой* (посколь-

ку через наблюдение сбоев делается заключение о наличии дефектов в системе).

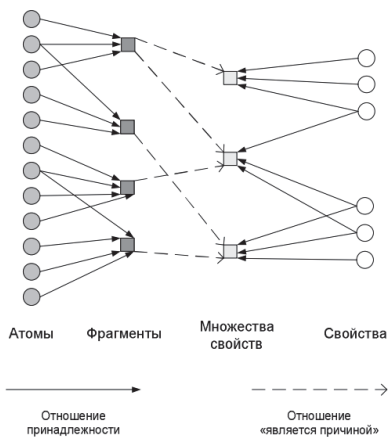


Рис. 1. Составляющие ДСМ-метода

Таким образом, наблюдая сбои, можно формировать гипотезы относительно присутствия или отсутствия дефектов в конкретных фрагментах кода, для которых неизвестно их присутствие или отсутствие. Также можно формировать гипотезы о наличии (отсутствии) сбоев в модулях, которые ещё не тестировались.

Очевидно, что наличие системы интеллектуального вывода в системе учёта дефектов позволит значительно усовершенствовать процесс тестирования и качество разрабатываемого программного продукта за счёт правильного распределения ресурсов тестирования.

Подводя итог, следует отметить, что исходя из приведённого выше материала, ДСМ-метод вполне может быть использован для решения задачи прогнозирования наличия (отсутствия) дефектов в программном обеспечении, обнаруживаемых на этапе тестирования. При этом следует расширить атрибуты дефекта ещё одним атрибутом, связывающим дефект с фрагментом кода, где, как предполагается, он находится. Назовём его *элемент*. Вопросы разбиения на элементы тестируемого программного кода и локализация дефекта на этапе тестирования в данном случае приобретают отдельное значение и подлежат подробному рассмотрению.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Аншаков О.М. ДСМ-метод: теоретико-множественное объяснение // НТИ. Сер. 2. Информ. Процессы и системы. 2012. №9, с. 1-21.
2. Василенко Н.В., Макаров В.А. Модели оценки надёжности программного обеспечения // Вестник Новгородского государственного университета. 2004. №28, с. 126-132

3. Милль Д.С. Система логики силлогистической и индуктивной. М.: ЛЕНАНД, 2011, 832 с.
4. Поппер К.Р. Логика и рост научного знания. М.: Прогресс, 1983, 605 с.
5. Peirce C.S. Abduction and induction. In: Buchler J (ed) Philosophical Writings of Peirce, 1995, Dover, NY, pp 150-156
6. Финн В.К. Искусственный интеллект: Методология, применение, философия, М.: КРАСАНД, 2011. – 448 с.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. В. Дюк, В. Эммануэль. Информационные технологии в медико-биологических исследованиях. Спб., 2003.
2. Гусев А.В., Романов Ф.А., Дуданов И.П. Опыт разработки медицинской информационной системы, 2001.
3. Вялков А.И. Управление и экономика здравоохранения. – М.: ГЭОТАР-Медиа, 2009.
4. Дуданов И.П., Гусев А.В., Романов Ф.А., Воронин А.В. Информационные системы в здравоохранении, 2002.

УДК 519.95

Кабанов А.Н., Матаев К.Р., Фоломкин Д.Н.

ФОРМИРОВАНИЕ ГЛАВНЫХ КОМПОНЕНТ ПРИ МНОГОКРИТЕРИАЛЬНОМ АНАЛИЗЕ СИСТЕМЫ

Рязанский государственный радиотехнический университет,
г. Рязань

Рассмотрена технология контроля многокритериальной системы на основе формирования главных компонент.

Постановка задачи

В статье рассматривается задача повышения оперативности и наглядности контроля многомерной и многокритериальной системы на основе использования метода Тьюки (названного им методом шелушения). На основе вероятностных оценок определяется число слоев для удаления. Полученная очищенная область допустимых изменений параметров используется для формирования главных компонент.

Решение задачи

Для контролируемой системы на основе исследований формируется область параметров работоспособной системы [1]. Для получения устойчивой области изменений параметров применяется метод Тьюки. Рассмотрим кратко этапы решения.

1. В начале определяется точка в многомерном пространстве на основе метода оптимума номинала.

Необходимо найти точку внутри области примерно на равном расстоянии от границы области. В большинстве случаев применяется квадратичный критерий.

Исходная система ограничений:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \geq b_1; \\ \dots\dots\dots; \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_m. \end{cases} \quad (*)$$

Левые части обозначим:

$$\begin{cases} y_1 = a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n; \\ \dots\dots\dots; \\ y_m = a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n; \end{cases}$$

Берём y_1 как целевую функцию и на множестве (*) находим максимальное и минимальное значения y_1 . Затем берем y_2 и опять находим максимальное и минимальное значения y_2 и т.д. ($y_{i\min}, y_{i\max}$).

$$\bar{y}_i = \frac{y_{i\min} + y_{i\max}}{2} - \text{среднее значение.}$$

Тогда система ограничений примет вид:

$$\begin{cases} \bar{y}_1 = a_{11}\bar{x}_1 + \dots + a_{1n}\bar{x}_n; \\ \dots\dots\dots; \\ \bar{y}_m = a_{m1}\bar{x}_1 + \dots + a_{mn}\bar{x}_n. \end{cases}$$

Это система линейных алгебраических уравнений.

Точка внутри области примерно на равном расстоянии от границы области определяется из уравнения $y_{cp} = \bar{A} \cdot \bar{x}_{cp}$:

$$\bar{x}_{cp} = \left(\bar{A}^T \cdot \bar{A} \right)^{-1} \cdot \bar{A}^T \cdot y_{cp}.$$

2. Исключение слоев ведется последовательно до тех пор, пока изменения области допустимых значений станет малой величиной.

Изменения области оцениваются величиной изменения объема.

Для многомерного пространства область допустимых параметров может быть описана системой линейных неравенств. При найденном значении номинальной точки расчет объема целесообразно проводить на основе вычисления объема пирамиды.

Формула объема пирамиды очень проста:

$V = (S \cdot H) / 3$, где S – площадь основания, а H – высота пирамиды. Для примера, пусть задана пирамида с основанием из трех треугольников.

Тогда $S = (S_1 + S_2 + S_3) / 3$. Площадь треугольника удобно рассчитывать по формуле Герона:

$S^2 = p(p-a)(p-b)(p-c)$, p – полупериметр треугольника $p = (a + b + c) / 2$; h – высота пирамиды, которая рассчитывается на основании знания номинальной точки области. После исключения первого слоя области допустимых изменений параметров определяют объем оставшейся области. Шелушение по методу Тьюки ведется до тех пор, пока изменения допустимых значений области станет малой величиной. Полученная после шелушения область может использоваться для формирования главных компонент, используемых для контроля системы.

Структура нейронной сети (НС) для сжатия данных представлена на рис. 1 [2, 3].

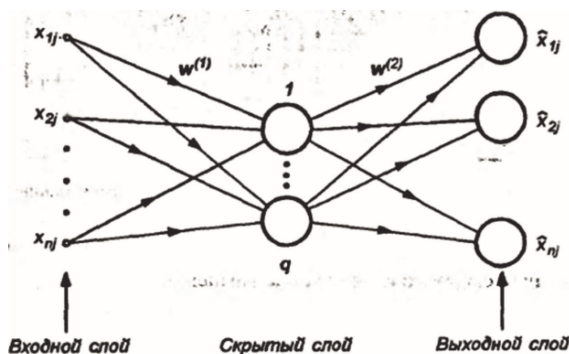


Рис. 1. НС для сжатия данных

Входные и выходные данные нейронной сети имеют одинаковую размерность:

Входные данные	Выходные данные
X1	Хапр1
X2	Хапр2
X3	Хапр3
XN	ХапрN

Скрытый слой определяется числом главных компонент $M < N$. Недостаток структуры: все нейроны подстраиваются одновременно. Реализуем определение главных компонент с помощью нейронной сети в пакете MS Excel. Определение главных компонент в пакете MS Excel представлен в примере:

x1	1	3	7	9	5
x2	2	6	14	18	10
x3	3	9	17	11	4
x4	12	36	68	44	16

W					
-0,21737	-0,43475	0,21196	0,84782		
0,39082	0,78165	0,11788	0,47155		
	Y=W*X				
	9,72292	29,1687	53,6473	29,8540	8,97867
	7,96647	23,8994	47,7489	39,6325	17,7871
WT	-0,21737	0,39082			
	-0,43475	0,78165			
	0,21196	0,11788			
	0,84782	0,47155			
WT*Y	1,00000	3,00001	7,00002	8,99999	4,99998
	2,00000	6,00000	14,0000	18	9,99999
	3,00000	9,00002	17,0000	10,9999	3,99996
	12	35,9999	67,9999	44,0000	16,0000

Выводы

Предложена методика формирования главных компонент объекта контроля на основе вероятностного анализа устойчивой области допустимых изменений параметров.

Предложенный метод формирования главных компонент при многокритериальном анализе системы может быть использован для улучшения качества продукции.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Повышение оперативности статистической идентификации динамического объекта / Д.Н. Фоломкин, А.Н. Кабанов, Г.И. Нечаев // Вестник РГРТУ. – 2012. – № 39. – Рязань: РГРТУ, 2012. – С. 110-113.
2. Паклин Н.Б., Орешков В.И. Бизнес-аналитика: от данных к знаниям – СПб: Питер, 2013 – 704 с.
3. Технологии обработки информации: учеб. Пособие / А.Н. Кабанов, Д.Н. Фоломкин: Рязан. гос. радиотехн. ун-т. Рязань, 2017. – 48 с.

УДК 519.95

Лощинин А.Б.

ИНФОРМАЦИОННЫЕ МЕДИЦИНСКИЕ СИСТЕМЫ

Рязанский государственный радиотехнический университет,
г. Рязань

В данной работе проведено исследование информационных медицинских систем: их цели, классификации, требования к ним.

Системой является совокупность взаимосвязанных элементов, организованных по определённом принципу для выполнения поставленной цели.

Информационная система (ИС) – это комплекс методологических, программных, технических, информационных и организационных средств, поддерживающих процессы функционирования информационной организации. Если информационная система выполняет некоторую часть функций самостоятельно, то она называется автоматизированной. Автоматизированная информационная система обеспечивает возможность выполнения как ручных, так и автоматизированных процессов – пользователь и компьютерные средства работают сообща с целью обработки и дальнейшего использования информации.

Медицинская автоматизированная информационная система (МИС) представляет собой совокупность программно-технических средств, баз данных и знаний, предназначенных для автоматизации различных процессов, протекающих в лечебно-профилактическом учреждении. Слово «автоматизированная» в литературе часто опускается в связи с тем, что часть функций в медицинских информационных системах обязательно должны быть автоматизированными. Автоматизированное рабочее место (АРМ) – рабочее место специалиста, оснащенное средствами вычислительной техники и программными комплексами для сбора, хранения медицинской информации, используемой им в качестве интеллектуального инструмента при принятии решений.

База данных (БД) – это совокупность взаимосвязанных данных при такой минимальной избыточности, которая допускает их использование оптимальным образом для одного или нескольких приложений.

Главная цель информатизации здравоохранения в целом сформулирована следующим образом: создание новых информационных технологий на всех уровнях управления здравоохранением и новых медицинских компьютерных технологий, повышающих качество лечебно-профилактической помощи и способствующих реализации основной функции охраны здоровья населения – увеличению продолжительности активной жизни. Это является основной, глобальной целью МИС. Наряду глобальной целью перед МИС выделяются следующие цели внедрения МИС, способствующие выполнению основной цели МИС: – создание единого информационного пространства, следствиями чего являются ускоренный доступ к информации, повышение качества медицинской документации; – мониторинг и управление качеством медицинской помощи и как результат – снижение вероятности врачебной ошибки, устранение избыточности назначения; – повышение прозрачности деятельности медицинского учреждения и эффективности принимаемых управленческих решений; – сокращение сроков обследования и лечения.

Существуют различные пути, посредством которых в медицинском учреждении компьютеризация используется для достижения целей, указанных выше. Выделяют следующие направления, которые определяют диапазон функциональных возможностей МИС:

1. Сбор, регистрация данных. Сбор информации является одной из основных функций МИС, которому предшествует получение первичных данных о пациенте в результате его осмотра или автоматически с помощью специального оборудования для регистрации состояния больного и, наконец, из других специализированных МИС. На основе автоматизации процессов управления регистрация данных позволяет ускорить процесс обработки информации, планировать использование оборудования, сократить время нахождения пациента в стационаре.

2. Обеспечение обмена информацией и создание информационного пространства. Использование компьютерных сетей и автоматизированных систем регистрации и документирования медицинских данных позволяет осуществлять одновременный доступ различных специалистов к необходимой информации, а также контроль ведения медицинских документов.

3. Хранение и поиск информации. Хранение данных в единой БД позволяет осуществить быстрый поиск необходимых сведений и избежать дублирования информации. Электронные архивы медицинских документов существенно облегчают и ускоряют подготовку рутинных отчетов и могут служить основой для использования алгоритмов интеллектуального анализа данных с целью поиска закономерностей.

4. Статистический анализ данных. Обработка статистической информации позволяет в удобной форме рассчитывать интересующие характеристики и представлять их в наглядной форме для анализа.

5. Контроль эффективности и качества оказания медицинской помощи. Внедрение компьютерных систем для осуществления контроля над этапами лечения и диагностики со стороны руководителей или более опытных специалистов может существенно повысить качество медицинской помощи.

При разработке медицинских информационных систем (МИС) и баз данных предъявляются следующие требования :

- гибкость и способность к эволюции БД и ИС;
- способность систем функционировать в условиях информационной неоднородности;
- иерархичность систем, возможность объединения сетей и их интероперабельность;
- возможность непрерывной модернизации;
- преемственность систем;
- надежность и выживаемость.

С развитием информационной инфраструктуры, ростом скорости передачи информации в сетях становится значительной роль распределенной обработки информации. Это обеспечивает повышение эффективности использования вычислительных ресурсов.

Остро в медицинских информационных системах стоит вопрос безопасности информации. Ее рассматривают с двух точек зрения:

- Защита прав личности от распространения конфиденциальной информации.

- Защита интересов государства и ведомств. Возможность утечки информации, злоупотребление, нарушение этики.

Отдельный вопрос – безопасность информационно-платежных систем обязательного и добровольного медицинского страхования, вопросы права, конфиденциальности информации, законности и правомочности электронной подписи, лицензирование применения средств криптозащиты.

Классификация информационных медицинских систем.

Известны различные виды классификации медицинских информационных систем. Например, приводится классификация в зависимости от уровня управления и организации:

- государственный (федеральный и региональный);
- территориальный (муниципальный, город, район);
- учрежденческий (вузы, структуры ОМС, службы лекарственного обеспечения, медтехники и др.);
- индивидуальный.

Другой пример классификации МИС (Информатика и системы управления, 1998):

- Информационная поддержка работы медицинского персонала.
- Информационное обеспечение экстренной медицинской помощи при чрезвычайных ситуациях.
- Мониторинг уровня здоровья населения.
- Информационное обеспечение научной работы.
- Система информационного обмена при работе в компьютерных сетях (в известной степени).

Также приводится разделение медицинских информационных систем по назначению:

- системы, основной функцией которых является накопление данных и информации;
- диагностические и консультирующие системы;
- системы, обеспечивающие медицинское обслуживание.

УДК 004.4

Горкин А.В.

РАБОТА С МАТЕМАТИЧЕСКИМИ ВЫРАЖЕНИЯМИ
В РАЗРАБОТКЕ ПОД ANDROID (JAVA)Рязанский государственный радиотехнический университет,
г. Рязань.

В данной статье разбираются библиотеки `exp4j` и `jjoe64:graphview`, используемые для работы с математическими выражениями при разработке мобильных приложений под операционной системой Android на языке Java. Рассказывается об основных принципах их работы и интеграции в собственный проект.

Наверное, каждый разработчик при изучении нового языка программирования начинает освоение платформы со своих небольших проектов. Зачастую, в основе таких приложений лежат различные математические задачи. Это касается и мобильной разработки. В статье будут продемонстрированы 2 библиотеки, которые выполняют весьма популярные задачи, реализуемые в небольших проектах:

exp4j – библиотека для парсинга математических выражений из строки для дальнейших расчетов;

jjoe64:graphview – библиотека для отрисовки графиков.

Так же рассмотрим их взаимодействие и объединение в своем проекте. Для написания приложения будет использоваться среда Android Studio. (Для начала работы следует создать пустой проект.) Рассмотрим в отдельности каждую из них.

Exp4j.

Официальный сайт: <https://www.objecthunter.net/exp4j/>

Установка:

Переходим по ссылке <https://www.objecthunter.net/exp4j/download.html> и скачиваем файлы библиотеки в формате jar (exp4j [последняя версия(на момент написания статьи 0.4.8] Binary Jar).

Переходим в Android Studio. Меняем режим отображения проекта на Project (по умолчанию выбрано Android – верхний левый угол окна).

Копируем скачанный файл .jar в папку `app/libs`

Открываем файл `app/build.gradle`

В разделе `dependencies` указываем скопированный файл: `implementation files('libs/exp4j-0.4.8.jar')`

Использование:

Простейший вариант использования:

```
Expression e = new ExpressionBuilder("x - 2")
    // создаем объект e класса Expression,
    // используя конструктор. Вместо "x-2" в
    // реальном проекте скорее всего будет
    // передаваться строковый объект.

variables("x")
    //указываем какие переменные будут
```

использоваться в выражении, возможна работа сразу с несколькими переменными (их требуется указать через запятую).

```
.build()
.setVariable("x", 2.3);           //указываем значение переменной x
double result = e.evaluate();      //для вычисления выражения
                                  используем метод evaluate().
```

Изначально, библиотека распознает ограниченный набор функций, но существует возможность создать собственные (для этого ознакомьтесь с документацией с официального сайта).

jjoe64:graphview.

Ссылка на GitHub: <https://github.com/jjoe64/GraphView>

Установка:

Переходим в Android Studio. Меняем режим отображения проекта на Project (по умолчанию выбрано Android – верхний левый угол окна);

Открываем файл app/build.gradle ;

В разделе dependencies прописываем строчку:

```
implementation 'com.jjoe64:graphview:4.2.2'
```

Использование:

1)Создадим элемент в файле разметки (res/layout)

```
<com.jjoe64.graphview.GraphView
    android:layout_width="match_parent"
    android:layout_height="200dip"
    android:id="@+id/graphTest" />
```

2) Java код.

```
GraphView graphTest = (GraphView)
findViewById(R.id.graphTest);
```

//Создаем объект класса
GraphView, ссылаясь на ранее
созданный элемент разметки.

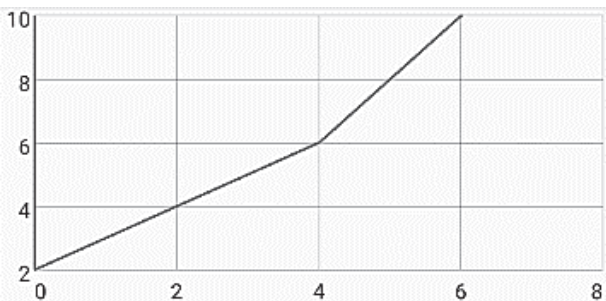
```
LineGraphSeries<DataPoint> series =
new LineGraphSeries<DataPoint>(new
DataPoint[] {
    new DataPoint(0, 1),
    new DataPoint(1, 2),
    new DataPoint(2, 4),
    new DataPoint(3, 5),
    new DataPoint(4, 6)
});
```

//создаем объект
LineGraphSeries, на основе кото-
рого будет построен график, в
конструкторе указывается мас-
сив из объектов DataPoint, кото-
рый в данном случае объявлен
непосредственно в конструкторе.

```
graphTest.addSeries(series);
```

//добавляем созданный
график.

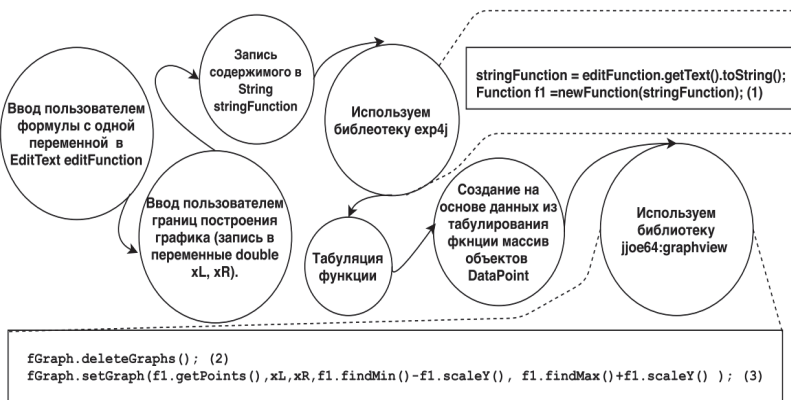
Результат выполнения:



Стоит отметить, что график можно подстраивать под себя, делать его отзывчивым на движения пользователя (например, для изменения масштаба или для перемещения по координатным осям).

Пример. Далее будет продемонстрирован пример объединения этих библиотек в 1 небольшом приложении. Требуется ввести с помощью смартфона функцию и вывести ее график на определенном промежутке.

Схема работы:



(1) Листинг конструктора класса Function

```

Function(String f){
    this.function = new
    ExpressionBuilder(f)
        .variables("x")
        .build();
}

```

(2) Листинг метода deleteGraphs

(3) Листинг метода setGraph

```

void setGraph(DataPoint[] dataPoints, double minX,
double maxX, double minY, double maxY){
    LineGraphSeries<DataPoint> series = new
    LineGraphSeries<>(dataPoints);

    graphXY.getViewport().setYAxisBoundsManual(true);
    graphXY.getViewport().setMinY(minY);
    graphXY.getViewport().setMaxY(maxY);

    graphXY.getViewport().setXAxisBoundsManual(true);
    graphXY.getViewport().setMinX(minX);
    graphXY.getViewport().setMaxX(maxX);
}

```

```

void deleteGraphs(){
    graphXY.setViewport().setScalable(true);
    graphXY.setViewport().setScalableY(true);
    graphXY.addSeries(series);
    graphXY.setCursorMode();
}

```



Рис. 1. Пример работы приложения

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Документация по библиотеке exp4j (Дата последнего обращения 20.12.2018) <https://www.objecthunter.net/exp4j/apidocs/index.html>.
2. Документация по библиотеке jjoe64.GraphView (Дата последнего обращения 20.12.2018) <https://github.com/jjoe64/GraphView>

УДК 519.725

Гринченко Н.Н., Корябкин В.В.

ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ИСПРАВЛЕНИЯ СТИРАНИЙ С ПОМОЩЬЮ КАСКАДНЫХ СХЕМ КОРРЕКЦИИ ОШИБОК

Рязанский государственный радиотехнический университет,
г. Рязань

Современные методы помехоустойчивого кодирования характеризуются весьма высоким уровнем достоверности декодирования, но все еще очень большими вычислительными затратами на обработку каждого декодируемого символа. Последнее обстоятельство и определяет исключительно высокую ценность методов кодирования и последующего успешного декодирования при большом шуме. Эти алгоритмы в несколько раз повышают к.п.д. использования каналов связи, а также на много десятичных порядков увеличивают надежность хранения цифровых данных. Это объясняет огромное количество международных конференций и тысячи публикаций по теории и прикладным вопросам теории инфор-

мации, где проблемы эффективного декодирования оказываются одними из главных тем дискуссий.

Следует отметить, что в ряде случаев для упрощения процедуры декодирования ненадежно принятые символы отмечают как стирания, которые восстановить гораздо легче и намного быстрее, чем исправить ошибки [1]. Поэтому в высокоскоростных системах передачи и, особенно, хранения данных приемник вместо применения сложного алгоритма исправления ошибок передачи/хранения некоторым образом, например, используя контрольные суммы «стирает» отдельные ненадежные символы или даже целые блоки и в дальнейшем их восстанавливает с помощью декодера помехоустойчивого кода.

На сегодняшний день одними из наиболее эффективных методов, восстанавливающих стирания, являются многопороговые декодеры (МПД) самоортогональных помехоустойчивых кодов (СОК), обладающие одним из лучших соотношением эффективности и сложности реализации [2]. Одной из особенностей МПД и используемых с ними СОК является то, что коды с большим кодовым расстоянием способны обеспечить меньшую вероятность невосстановления стирания по сравнению с кодами с меньшим кодовым расстоянием. Но это справедливо для небольшой вероятности стираний в канале. При большой же вероятности стирания оказывается, что коды с большим кодовым расстоянием неспособны их исправить. Это существенно усложняет получение малых вероятностей невосстановления стираний при большом шуме в канале. Для решения данной проблемы можно использовать каскадные схемы коррекции ошибок, в которых во внутреннем каскаде используется СОК с небольшим кодовым расстоянием, а во внешнем каскаде простой для декодирования высокоскоростной код, например, код с контролем четности, код Хэмминга или код БЧХ. Предполагается, что МПД внутреннего кода будет работать почти как оптимальный декодер и значительно снизит вероятность стирания по сравнению с вероятностью стирания в канале. После этого декодер внешнего кода позволит еще существенно уменьшить вероятность стирания, что обеспечит требуемую вероятность невосстановления стирания даже при большой вероятности стирания в канале.

Рассмотрим основные принципы организации каскадной схемы, в которой во внутреннем каскаде используется СОК, декодируемый с помощью МПД, а во внешнем – код с контролем четности, код Хэмминга или код БЧХ (рис. 1).

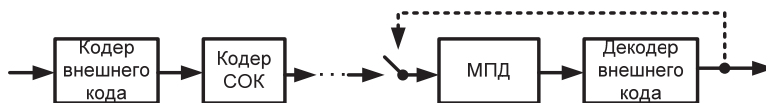


Рис. 1. Схема кодирования и декодирования каскадного кода

При декодировании такого кода сначала выполняется декодирование СОК с помощью МПД, после чего осуществляется декодирование внешнего кода. Заметим, что данный процесс при декодировании может повторяться многократно, позволяя улучшать решение декодера после каждой итерации.

Выполним аналитическую оценку эффективности предложенной схемы каскадирования в предположении, что данные коды образуют код-произведение. Пусть во внутреннем каскаде используется СОК с кодовым расстоянием d_{in} , а во внешнем – код длиной n_{out} и кодовым расстоянием d_{out} . Если предположить, что при выбранной вероятности стирания P_e в канале МПД выполняет близкое к оптимальному декодирование, то вероятность невосстановленного стирания после МПД можно оценить снизу как [1]

$$P_{MTD} = P_e^{d_{in}}.$$

Внешний код с кодовым расстоянием d_{out} позволяет восстановить $t = d_{out} - 1$ стираний в блоке длиной n_{out} . Если в блоке будет больше стираний, то они не восстанавливаются. Следовательно, вероятность невосстановления стирания после декодера каскадной схемы оценивается снизу как

$$P_{out} = \sum_{k=t+1}^{n_{out}} \frac{k}{n_{out}} C_{n_{out}}^k P_{MTD}^k (1 - P_{MTD})^{n_{out} - k}.$$

Далее представим зависимости вероятности невосстановления стирания после декодирования от вероятности стирания в двоичном стирающем канале связи для различных внешних кодов, полученные с помощью компьютерного моделирования. При этом в качестве внутреннего кода использовался СОК с кодовой скоростью $R_{in} = 1/2$, кодовым расстоянием $d_{in} = 9$ и длиной $n_{in} = 10200$ бит.

График зависимости вероятности стирания после декодирования внутреннего кода от вероятности стирания в канале показан на рис. 2 кривой 1. При использовании в качестве внешнего кода с контролем четности длиной 255 битов, восстанавливающего одно стирание, получаются характеристики, представленные кривой 2, при использовании внешнего кода с контролем четности длиной 32 бита, восстанавливающего одно стирание, – кривой 3, при использовании внешнего кода Хэмминга (255, 247), восстанавливающего два стирания, – кривой 4, при использовании внешнего кода БЧХ (255, 239), восстанавливающего четыре стирания, – кривой 5. Пунктирами на рисунке показаны нижние оценки вероятности стирания после декодера, полученные с помощью ранее полученных выражений. Отметим, что за счет применения внешних кодов удастся получить на 1..5 десятичных порядков меньшую вероятность невосстановления стираний в области эффективной работы МПД. Так же

отметим, что полученные нижние оценки вероятности стирания после декодера для предложенных каскадных схем являются достаточно точными, что позволяет их использовать для оценивания эффективности каскадных схем коррекции ошибок при очень малых целевых вероятностях невозможности стираний (порядка 10^{-12} и менее), для которых затруднительно применение компьютерного моделирования.

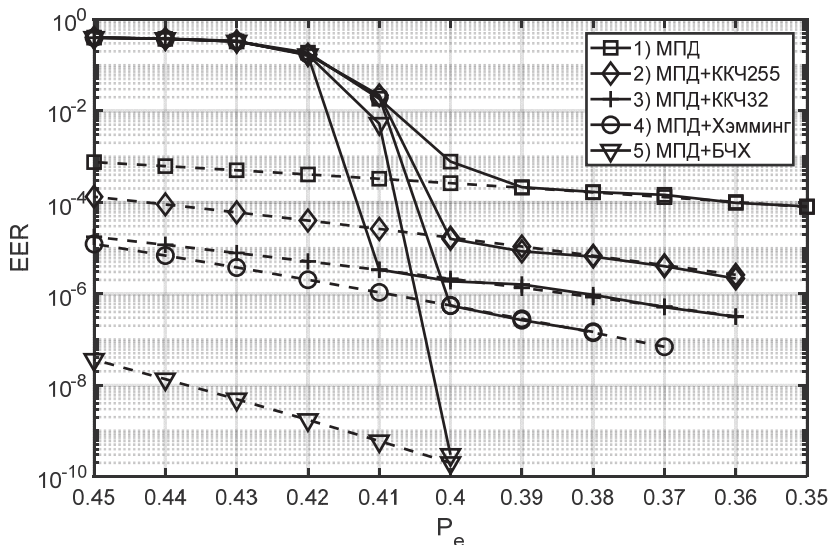


Рис. 2. Характеристики методов восстановления стираний

Таким образом, в работе предложена каскадная схема восстановления стираний, основанная на простых для декодирования внешних кодах и внутренних СОК, использование которой позволяет без существенного увеличения вычислительной сложности на несколько десятичных порядков уменьшить вероятность стирания после декодирования по сравнению с МПД для внутреннего СОК.

Работа выполнена при поддержке РФФИ (грант № 18-07-00525).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Золотарев В.В., Зубарев Ю.Б., Овечкин Г.В. Многопороговые декодеры и оптимизационная теория кодирования. М.: Горячая линия – Телеком, 2012, 239 с.
2. Баринев И.В., Овечкин Г.В. Методы декодирования помехоустойчивых кодов для восстановления стираний // Труды Всероссийской научно-технической конференции «Интеллектуальные и информационные системы» (23 – 25 ноября 2016 г., г. Тула). 2016. С. 64-67.

УДК 519.816

Кузин И.Н.

ИССЛЕДОВАНИЕ ЯВЛЕНИЯ «ПЕРЕСТАНОВКИ РАНГОВ» В МЕТОДЕ АНАЛИЗА ИЕРАРХИЙ

Рязанский государственный радиотехнический университет,

г. Рязань

В данной статье рассматривается перестановка рангов в методе анализа иерархий с приведением примера данного явления

Метод анализа иерархий – это метод, помогающий осуществить принятие решения и позволяющий лицу выбрать такой вариант, который наилучшим образом подходит именно ему. Самая простая иерархия состоит из трех уровней, а именно из цели, критериев и альтернатив. Приоритеты – это числа, которые показывают значимость критериев и альтернатив относительно цели. При использовании относительных приоритетов в методе анализа иерархий появляется явление, которое имеет название «перестановка рангов».

Перестановка рангов – это ситуация, которая происходит из-за внесения новой или избавления от старой альтернативы, находящейся в модели. Результатом данного явления является смена предпочтений, существующих среди альтернатив. Причиной является то, что изменения на множестве сравниваемых альтернатив при относительном оценивании вызывают перераспределение приоритетов, ведь если вносимая или извлекаемая альтернатива имеет больший приоритет по какому-либо критерию, то приоритет другой альтернативы по тому же критерию станет меньше [1].

Рассмотрим пример такого случая: пусть есть главная цель F , критерии (G_1, G_2) и альтернативы (y_1, y_2, y_3). Допустим, приоритеты обоих критериев равняются 0,5. Предпочтения между альтернативами будут следующие: y_3 превосходит y_1 по критерию G_1 , но уступает по G_2 . В свою очередь y_1 так же превосходит y_2 по критерию G_1 , но уступает по G_2 . Представим приоритеты альтернатив относительно критериев в виде табл. 1.

Таблица 1

	G_1	G_2
y_1	0,3	0,3
y_2	0,085	0,675
y_3	0,615	0,025

В таком случае, итоговое распределение приоритетов между альтернативами будет следующее (табл. 2).

Таблица 2

	G_1	G_2	F	Ранг
y_1	0,3	0,3	0,3	3
y_2	0,085	0,675	0,38	1
y_3	0,615	0,025	0,32	2

Теперь, если мы удалим из модели альтернативу y_3 , то приоритеты альтернатив относительно критериев станут следующими (табл. 3).

Таблица 3

	G_1	G_2
y_1	0,8	0,25
y_2	0,2	0,75

Рассмотрим, как изменится распределение приоритетов между альтернативами (табл. 4).

Таблица 4

	G_1	G_2	F	Ранг
y_1	0,8	0,25	0,525	1
y_2	0,2	0,75	0,475	2

Такое перераспределение можно объяснить следующим образом: удаление альтернативы y_3 привело к тому, что более высокий приоритет альтернативы y_1 над альтернативой y_2 по критерию G_1 добавил первой альтернативе преимущества. По второму критерию альтернатива y_3 не имела большого приоритета, поэтому в данном критерии практически не произошло изменений. В результате превосходство альтернативы y_2 над альтернативой y_1 по критерию G_2 является слишком малым для того, чтобы окупить отставание по критерию G_1 .

Чтобы избежать перестановки рангов можно использовать абсолютные приоритеты. Однако для этого необходимо, чтобы для каждого критерия существовала альтернатива, которая бы не изменялась при изменении множества альтернатив.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Саати Т.Л. Принятие решений. Метод анализа иерархий. – М.: Радио и связь, 1993. – 278 с.

УДК 004.89

Власов П.А.

КОМПЬЮТЕРНЫЕ ВИРУСЫ И ИХ КЛАССИФИКАЦИЯ

Рязанский государственный радиотехнический университет,
г. Рязань

Рассматривается понятие компьютерного вируса, как возможной угрозы нормального функционирования вычислительных систем, а также дается их классификация и способы профилактической защиты.

Первая массовая эпидемия компьютерного вируса произошла в 1986 г., когда вирус Brain «заражал» дискеты для первых массовых персональных компьютеров. В настоящее время известно несколько десятков тысяч вирусов, попадающих на компьютеры с различными операционными системами и распространяющихся по компьютерным сетям.

Обязательным свойством компьютерного вируса является способность к размножению (самокопированию) и незаметному для пользователя внедрению в файлы, загрузочные секторы дисков и документы. Название «вирус» по отношению к компьютеру и компьютерным программам пришло из биологии именно по признаку способности к саморазмножению.

После заражения компьютера вирус может активизироваться и заставить компьютер выполнять какие-либо действия. Его активизация может быть связана с различными событиями (наступлением определенной даты, запуском программы, открытием документа и так далее).

Разнообразны последствия действия вирусов; по величине вредных воздействий их можно разделить на:

- неопасные, влияние которых ограничивается уменьшением свободной памяти на диске, графическими, звуковыми и другими внешними эффектами;
- опасные: могут привести к сбоям и зависаниям при работе компьютера;
- очень опасные: активизируясь, приводят к потере данных и программ (изменению или удалению файлов и каталогов), форматированию винчестера и так далее.

По «среде обитания» вирусы можно разделить на файловые, загрузочные, макровирусы и сетевые.

Файловые вирусы. Они различными способами внедряются в исполняемые файлы (программы) и обычно активизируются при их запуске. После запуска зараженной программы вирус находится в оперативной памяти компьютера и является активным (то есть может заражать другие файлы) вплоть до момента выключения компьютера или перезагрузки операционной системы.

Загрузочные вирусы. Записывают себя в загрузочный сектор диска. При загрузке ОС с зараженного диска вирусы внедряются в оперативную память компьютера. В дальнейшем они ведут себя так же, как файловый, то есть могут заражать файлы при обращении к ним компьютера.

Профилактическая защита от таких вирусов состоит в отказе от загрузки операционной системы с гибких дисков и установке в BIOS вашего компьютера защиты загрузочного сектора от изменений.

Макровирусы. Заражают файлы документов Word и электронных таблиц Excel. Макровирусы являются фактически макрокомандами (макросами), которые встраиваются в документ.

После загрузки зараженного документа в приложение макровирусы постоянно присутствуют в памяти компьютера и могут заражать другие документы. Угроза заражения прекращается только после закрытия приложения.

Профилактическая защита от них состоит в предотвращении запуска вируса. При открытии документа в приложениях Word и Excel сообщается о присутствии в них макросов (потенциальных вирусов) и предлагается запретить их загрузку. Выбор запрета на загрузку макросов надежно защитит ваш компьютер от заражения макровирусами, однако отключит и полезные макросы, содержащиеся в документе.

Сетевые вирусы. По компьютерной сети могут распространяться и заражать компьютеры любые обычные вирусы. Это может происходить, например, при получении зараженных файлов с серверов файловых архивов. Однако существуют и специфические сетевые вирусы, которые используют для своего распространения электронную почту и Всемирную паутину.

Интернет-черви (worm) – распространяются в компьютерной сети во вложенных в почтовое сообщение файлах. Автоматическая активизация червя и заражение компьютера могут произойти при обычном просмотре сообщения. Опасность таких вирусов состоит в том, что они по определенным датам активизируются и уничтожают файлы на дисках зараженного компьютера.

Кроме того, интернет – черви часто являются троянами, выполняя роль «троянского коня», внедренного в операционную систему. Такие вирусы «похищают» идентификатор и пароль пользователя для доступа в Интернет и передают их на определенный почтовый адрес. В результате злоумышленники получают возможность доступа в Интернет за деньги ничего не подозревающих пользователей.

Лавинообразная цепная реакция распространения вируса базируется на том, что он после заражения компьютера начинает рассылать себя по всем адресам электронной почты, которые имеются в адресной книге пользователя. Кроме того, может происходить заражение и по локальной

сети, так как червь перебирает все локальные диски и сетевые диски с правом доступа и копируется туда под случайным именем.

Профилактическая защита от интернет-червей состоит в том, что не рекомендуется открывать вложенные в почтовые сообщения файлы, полученные из сомнительных источников.

Особой разновидностью вирусов являются активные элементы (программы) на языках Java Script или VB Script, которые могут выполнять разрушительные действия, то есть являться вирусами (*скрипт-вирусами*). Такие программы передаются по Всемирной паутине в процессе загрузки Web-страниц с серверов Интернета в браузер локального компьютера.

Профилактическая защита от скрипт-вирусов состоит в том, что в браузере можно запретить получение активных элементов на локальный компьютер.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. <http://www.5byte.ru/10/0033.php>

УДК 005.6(075.8)

Матаев К.Р., Кабанов А.Н.

ТЕХНОЛОГИЯ ПРОГНОЗИРОВАНИЯ МНОГОМЕРНЫХ ВРЕМЕННЫХ РЯДОВ

Рязанский государственный радиотехнический университет,
г. Рязань

В статье рассматривается последовательность операций для получения прогноза временного ряда. На основе многомерноматричных представлений [1] решена задача разбиения временных рядов на однородные группы и получение рекуррентных оценок параметров многомерных временных рядов на сколь угодно постоянном интервале.

Постановка задачи

Исходный многомерный процесс описывается уравнением:

$$y(1,0;n) = x(0,1;n) \cdot b(2,0) + e(1,0;n), \quad (1)$$

где $Y(1,0;n) = \{y_1(n), y_2(n), \dots, y_k(n)\}$ – наблюдаемые временные ряды; $x(0,1;n) = \{x_1[n], x_2(n), \dots, x_l(n)\}$ – наблюдаемые параметры временных рядов; $b(2,0) = \{b_{ij}\}$, $i = \{1, \dots, k\}$; $j = \{1, \dots, l\}$ – оцениваемые коэффициенты временных рядов; $e(1,0;n) = \{e_1(n), e_2(n), \dots, e_k(n)\}$ – помеха типа белого шума. Необходимо рекуррентно оценивать коэффициенты $\{b_{ij}\}$ для однородных групп временных рядов.

Решение задачи

1. Множество наблюдаемых процессов необходимо разбить на однородные группы по наблюдаемым данным $\{y_1(n), y_2(n), \dots, y_k(n)\}$, $\{x_1[n], x_2(n), \dots, x_l(n)\}$. Данная операция выполняется на основе пространения алгоритма Прима на многомерный случай.

2. После определения однородных групп необходимо исключить аномальные измерения в отдельных процессах. Это можно выполнить с помощью метода наименьших модулей.

Для отдельного временного ряда уравнение имеет вид

$$Y[m] = X^T[m]b[n] + e[m]; \quad m = \overline{n - N, n}$$

где $X[m] = \{x_1[n], x_2[n], \dots, x_l[n]\}$ – вектор наблюдаемых линейно независимых факторов; b – вектор неизвестных и подлежащих оценке параметров; $e[m]$ – помеха типа белого шума.

В основу алгоритма положен итеративный метод решения на основе МНМ. При этом оценка $\hat{b}$, полученная на основе N результатов измерения, имеет вид

$$\hat{b}_N[n] = A_N^{-1}[n]Z_N[n],$$

где

$$A_N[n] = \sum_{m=n-N}^n X[m]R[m]X^T[m];$$

$$Z_N[n] = \sum_{m=n-N}^n X[m]R[m]Y[m],$$

где $R[m] = (Y[m] - \hat{Y}[m])^{-1}$; $\hat{Y}[m] = X^T[m]\hat{b}_N[n]$ – оценка m -го измерения выходного сигнала.

Начальные значения $R[m]=1$; $m = \overline{n - N, n}$ соответствуют определению параметров $\hat{b}$ по методу наименьших квадратов (МНК). Далее вычисления оценок проводятся итерационно до тех пор, пока изменения оценок за одну итерацию не достигнут заданной малой величины. При этом наименьший весовой коэффициент $R[l^*]$ указывает на наиболее грубое l^* -измерение.

3. Исходный процесс $y[n]$ представлен в матричном виде (1).

Модель процесса по результатам наблюдения на интервале $(n_0 - n_0 + N)$ имеет вид

$$yA(1, 0; n) = x(1, 0; n) \cdot \hat{b}(2, 0; n_0 + N);$$

Для квадратичного критерия оптимальные оценки определяются соотношением:

$$\hat{b}(2, 0; n_0 + N) = A^{-1}(1, 1; n_0 + N) \cdot z(2, 0; n_0 + N);$$

$$A(1, 1; n_0 + N) = \sum_{n=n_0}^{n_0+N} x(1, 0; n) \cdot x(0, 1; n);$$

$$z(2, 0; no + N) = \sum_{n=n_0}^{no+N} x(1, 0; n) \cdot y(1, 0; n);$$

Для повышения оперативности вычислений на скользящем постоянном интервале используем известные матричные соотношения

Если $A(1, 1) = B(1, 1) + x(1, 0) * x(0, 1)$, то

$$A^{-1} = B^{-1} - \frac{B^{-1} * x(1, 0) * x(0, 1) * B^{-1}}{x(0, 1) * B^{-1}(1, 1) * x(1, 0) + 1}$$

Если $A(1, 1) = B(1, 1) - x(1, 0) * x(0, 1)$, то

$$A^{-1} = B^{-1} + \frac{B^{-1} * x(1, 0) * x(0, 1) * B^{-1}}{1 - x(0, 1) * B^{-1}(1, 1) * x(1, 0)}$$

При скольжении с постоянным интервалом составляющие процесса в начале интервала удаляются из матрицы А, последующие составляющие добавляются в матрицу А.

При использовании данного алгоритма наблюдается значительный выигрыш по числу вычислительных операций по сравнению с процедурой метода наименьших квадратов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Математическая статистика и прогнозирование: учеб. пособие / М.П. Булаев, Н.В. Дорошина, А.Н. Кабанов; Рязан. гос. радиотехн. ун-т. Рязань, 2014. 64 с.

УДК 004.89

Аяззова Е.А., Коровин А.А.

КРИПТОСТОЙКОСТЬ ШИФРОВ

Рязанский государственный радиотехнический университет,

г. Рязань

Дается определение криптографической стойкости шифров. Рассматриваются модели стойкости, приводятся примеры криптостойких алгоритмов шифрования, на примере AES-алгоритма. Выявляются преимущества от использования крипто-систем с открытым ключом.

Криптографический алгоритм, или шифр – это математическая формула, описывающая процессы зашифрования и расшифрования. Чтобы зашифровать открытый текст, криптоалгоритм работает в сочетании с ключом-словом, числом или фразой. Одно и то же сообщение одним алгоритмом, но разными ключами будет преобразовываться в разный шифртекст. Защищённость шифртекста целиком зависит от двух вещей: стойкости криптоалгоритма и секретности ключа.

Стойкость шифра – это его способность противостоять атакам на него. Стойким считается алгоритм, который для успешной атаки требует

от противника недостижимых вычислительных ресурсов, недостижимого объёма перехваченных открытых и зашифрованных сообщений или же такого времени раскрытия, что по его истечению защищенная информация будет уже не актуальна.

В криптографии различают три типа стойкости.

1. Вычислительная стойкость – когда имеется потенциальная возможность вскрыть шифр, но при выбранных в шифры параметрах и ключах на современном этапе развития криптоанализа у противника не хватит вычислительных ресурсов и времени для вскрытия.

2. Информационно-теоретическая стойкость (или абсолютная стойкость), когда криптоаналитик не может раскрыть криптосистему ни теоретически, ни практически, даже имея бесконечно большие вычислительные ресурсы. Доказательства стойкости в такой модели выводятся из теории информации.

3. Доказуемая стойкость, при которой доказательство стойкости криптосистем сводят к решению определенной трудно решаемой математической проблемы, положенной в основу алгоритму [1].

Стойкость количественно должна отражать трудоемкость нарушения информационной безопасности. Наиболее употребительной единицей измерения стойкости является временная сложность наилучшего известного алгоритма, нарушающего безопасность. Однако иногда используются и другие единицы измерения стойкости:

- длительность вычислений по вскрытию ключа (с учетом развития вычислительной модели нарушителя);

- требуемый объем памяти для вскрытия ключа;

- стоимость вскрытия ключа;

- количество энергии, необходимое для вскрытия ключа;

- физический объем вычислительной модели для вскрытия ключа.

Ключ – некоторая величина, которая, работая в сочетании с криптоалгоритмом, производит определённый шифртекст.

Преимущество от использования криптосистем с открытым ключом состоит в том, что они предоставляют возможность создания электронных цифровых подписей. Цифровая подпись позволяет получателю сообщения убедиться в аутентичности источника информации, а также проверить, была ли информация изменена или искажена, пока находилась в пути. Таким образом, цифровая подпись является средством аутентификации и контроля целостности данных. Кроме того, она несёт принцип неотказа, который означает, что отправитель не может отказаться от факта своего авторства подписанной им информации. Эти возможности столь же важны для криптографии, как и секретность.

Помимо этого в процессе шифрования генерируется хэш-функция, которая берёт ввод произвольной длины, называемый прообразом, например, сообщение любого размера, тысячи или миллионы бит и генерирует

строго зависящий от прообраза вывод фиксированной длины. Хэш-функция гарантирует, что если информация будет любым образом модифицирована, даже на один бит, в результате получится совершенно иное хэш – значение [2].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Сушко С.А. Лекция 3. Практическая криптология. URL: <http://bit.nmu.org.ua/ua/student/metod/cryptology/лекция3.pdf>
2. Власов П.А., Гагаринский В.Ю. Криптографическая стойкость шифров. Научный альманах. По материалам международной научно-практической конференции «Вопросы образования и науки». Тамбов, 2018 – 205с.

УДК 004.89

Власов П.А., Гагаринский В.Ю.

ОДНОМЕРНАЯ ДИНАМИКА.

ОПРЕДЕЛЕНИЕ ОПТИМАЛЬНОГО РЕШЕНИЯ.

ЗАДАЧА О КУЗНЕЧИКЕ, КОТОРЫЙ СОБИРАЕТ МОНЕТЫ

Рязанский государственный радиотехнический университет,

г. Рязань

Задачи динамического программирования делятся на два типа. Подсчет количества вариантов решения. В этом случае находят суммарное количество решений подзадач. Поиск оптимального решения (оптимизация целевой функции). В этом случае выбирают лучшее среди всех решений подзадач.

Определение оптимального решения. Задача о кузнечике, который собирает монеты

Задачи, в которых требуется определить наилучшее решение среди возможных (производится поиск оптимального решения), удобно решать при помощи динамического программирования. Проиллюстрируем применение метода динамического программирования для поиска оптимального решения на конкретной задаче.

Задача. Кузнечик собирает монеты

Кузнечик прыгает по столбикам, расположенным на одной линии на равных расстояниях друг от друга. Столбики имеют порядковые номера от 1 до N . Вначале кузнечик сидит на столбике с номером 1. Он может прыгнуть вперед на расстояние от 1 до K столбиков, считая от текущего.

На каждом столбике кузнечик может получить или потерять несколько золотых монет (для каждого столбика это число известно). Необходимо определить, как нужно прыгать кузнечику, чтобы собрать наибольшее количество золотых монет. Учитывайте, что кузнечик не может прыгать назад.

Входные данные

В первой строке вводятся два натуральных числа: N и K ($2 \leq N$, $K \leq 10000$), разделённые пробелом. Во второй строке записаны через пробел $N-2$ целых числа – количество монет, которое кузнечик получает на каждом столбике, от 2-го до $N-1$ -го. Если это число отрицательное, кузнечик теряет монеты. Гарантируется, что все числа по модулю не превосходят 10000.

Выходные данные

В первой строке программа должна вывести наибольшее количество монет, которое может собрать кузнечик. Во второй строке выводится число прыжков кузнечика, а в третьей строке – номера всех столбиков, на которых побывал кузнечик (через пробел в порядке возрастания).

Если правильных ответов несколько, выводим любой из них (табл. 1).

Таблица 1

Входные данные	Выходные данные
5 3	7
2 -3 5	3
	1 2 4 5

Для решения задачи будем использовать массив динамики $d[]$, где $d[i]$ – наибольшее количество золотых монет, которое может собрать кузнечик, находясь в данный момент на столбике i .

База динамики. Начальное заполнение массива динамики $d[1]=0$ – если кузнечик находится на первом столбике, то он собрал 0 монет[1].

Правило перехода динамики $d[i] = d[\text{num_max}] + a[i]$, где num_max – номер столбика с максимальным количеством собранных кузнечиком монет, который предшествовал данному. То есть, кузнечик, находясь на столбике с номером i , может собрать такое максимальное количество монет, которое равно сумме предыдущего максимального количества монет и количеству монет на текущем столбике.

Ответом к задаче будет являться число $d[n]$, где n – номер последнего столбика (Пример 1) [2].

```
d[1]=0; a[1]=0; a[n]=0;
for (int i=2;i<=n;++i)
{ int num_max = i - 1;
//поиск предыдущего столбика
//с максимальным количеством монет
for(int j=max(1,i-k);j<=i-1;++j)
if (d[j]>d[num_max])
{ num_max=j;
}
d[i]=d[num_max]+a[i];
//Текущее максимальное значение
p[i] = num_max;
```

```

}
printf("%d\n", d[n]);

```

Пример 1. Фрагмент программы

Дополнительно в программе используется массив $p[]$, где $p[i]$ – номер столбика с максимальным количеством монет, предшествующего i -му столбику. Этот массив будет использоваться для восстановления ответа – перечисления всех номеров столбиков, на которых побывал кузнечик.

Восстановление ответа

В задаче про кузнечика и многих других нужно не только дать ответ в виде количества способов решения или наилучшего решения, но и указать всевозможные способы или способ достижения этого наилучшего решения. Например, в задаче про кузнечика требовалось указать номера столбиков, по которым перемещался кузнечик. В программе был использован массив предков $p[]$, который для каждого i столбика хранил значение $p[i]$ – номер того столбика, с которого кузнечик прыгнул на i столбик. Теперь мы просто можем переходить к предыдущему столбику с конца решения вот таким образом – $i = p[i]$. Это позволит восстановить ответ – указать цепочку столбиков, на которых был кузнечик. Получаемые на каждом шаге цикла номера столбиков будем записывать в массив ans . Поскольку заполнение массива происходит, начиная с последнего столбика и, заканчивая первым, то вывод вектора ans нужно осуществлять в обратном порядке. Приведем фрагмент программы (Пример 2).

```

int i=n;i vector<int> ans; ans.push_back(i); do
{
i=p[i];
ans.push_back(i);
}
while (i>1); printf("%d\n",ans.size()-1);
for(auto i=ans.rbegin();i!=ans.rend();++i)
printf("%d ",*i);

```

Пример 2. Фрагмент программы

Для теста, указанного в задаче проиллюстрируем заполнение массива динамики d и массива предков p . Входные данные: 5 – количество столбиков, 3 – количество прыжков, которые может делать кузнечик (от 1 до 3), 2 3 5 – количество монет, которые лежат на столбиках (табл. 2).

Таблица 2

i – номер столбика	1	2	3	4	5
$a[i]$ – количество монет	0	2	-3	5	0
$d[i]$	0	2	-1	7	7
$p[i]$	-1	1	2	2	4

Из заполненной таблицы следует, что $d[5] = 7$ – максимальное количество собранных монет. Восстановление ответа (столбиков, на которых был кузнечик) проходит следующим образом: 5 столбик – последний, на котором он был; его предком является столбик 4; в свою очередь его предком является столбик 2, а его предком, столбик 1. То есть получается последовательность элементов вектора ans : 5, 4, 2, 1. Выведем ее в обратном порядке и получим: 1, 2, 4, 5 [3].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Порублев И.Н., Ставровский А.Б. Алгоритмы и программы. Решение олимпиадных задач. – М.: ООО «И.Д. Вильямс», 2007. – 475 с.
2. Ворожцов А.В., Винокуров Н.А. Лекции «Алгоритмы: построение, анализ и реализация на языке программирования Си». – М.: Издательство МФТИ, 2007.
3. Национальный исследовательский университет «Высшая школа экономики». Факультет компьютерных наук. Летняя школа по компьютерным наукам. Август. 2016. Параллель С. Шуйкова И.А.

УДК 004.23

Магомадов Р.У., Коровин А.А.

АЛГОРИТМ ОТРИЦАТЕЛЬНОГО ОТБОРА В РАМКАХ ЗАЩИТЫ ИНФОРМАЦИИ

Рязанский государственный радиотехнический университет,
г. Рязань

Интенсивное развитие вычислительной техники и программного обеспечения обостряет проблему компьютерной безопасности, повышает требования к контролю за состоянием системы, требует качественных методов поиска сбоев и аномалий в ее работе.

Наиболее распространенной иммунологической моделью искусственной иммунной системы, применяемой в сфере компьютерной безопасности является алгоритм отрицательного отбора (АОО), который в самом общем виде формулируется следующим образом: определяется понятие «свой», как нормальная динамика поведения системы, которая описывается множеством строк символов фиксированной длины; создается набор детекторов «произвольных чужих», каждый из которых не должен совпадать с любой строкой нормальной совокупности строк символов «своего». При этом используется правило частичного совпадения, согласно которому две строки совпадают тогда и только тогда, когда они идентичны в определенном числе смежных позиций; происходит непрерывное сопоставление новых поступлений строк в детекторную систему. Если строка совпадает с одним из детекторов она классифицируется как представитель «чужого» [1].

Случайно сгенерированный детектор тестируют, делая проверку на «правильном» наборе данных. В случае срабатывания детектора, его удаляют и генерируют новый.

Решение конкретной задачи требует описания специфической реализации некоторых операторов и функций. Иммунная сеть может использовать вещественное кодирование антител, при котором для вычисления расстояния применяется Евклидова метрика. При этом вокруг антител формируется одномерная радиальная область распознавания с радиусом r , называемым кросс-реактивным порогом. Кросс-реактивный порог включен в состав структуры антитела, что дает возможность адаптивной настройки его значения. Таким образом, иммунная сеть заполняет пространство «чужих» распознающими гиперсферами разного радиуса, что дает возможность его более полного покрытия.

Для расчета значений аффинности связи «антиген-антитело» используется следующее соотношение: $Ab-Ag = kr + DE(Ab-Ag)$, где r – кросс-реактивный порог антитела (детектора); kr – коэффициент значимости кросс-реактивного порога (параметр настройки алгоритма). Параметр kr является очень важным параметром обучения. Он управляет робастностью получаемого решения. Увеличение этого параметра заставляет иммунную сеть поддерживать детекторы большего радиуса, что дает более грубое, но более устойчивое решение. Чрезмерное увеличение kr отрицательно сказывается на точности решения.

Сжатие иммунной сети осуществляется на основании самораспознавания клеток, которое численно выражается в виде аффинности связей антител друг с другом. При этом сжатие осуществляется в зависимости от величины параметра порога сжатия, являющегося параметром алгоритма обучения; в некоторых случаях сжатие необходимо, так как наблюдается избыточность распознающих элементов. Таким образом, иммунная сеть в конце каждого поколения гарантировано не содержит детекторов, распознающих «свои» антигены [2].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Брюхомицкий Ю.А. Мониторинг информационных процессов методами искусственных иммунных систем // Известник Южного федерального университета. Технические науки. 2012. С 82-89.
2. Бидюк П.И., Фефелов А.А., Литвиненко В.И. Иммуносетевая модификация алгоритма отрицательного отбора для решения задачи обнаружения аномалий // «АСАУ»-10.2007.С 15-24.

УДК 004.89**Гагаринский В.Ю.****ОДНОМЕРНАЯ ДИНАМИКА.
ПОДСЧЕТ КОЛИЧЕСТВА ВАРИАНТОВ РЕШЕНИЯ**

Рязанский государственный радиотехнический университет,

г. Рязань

Рассматривается способ решения задачи с использованием одномерной динамики методом динамического программирования, в котором используется один параметр.

Задачи, в которых при наличии нескольких возможных вариантов решения, предполагается выбор в качестве ответа одного наилучшего (оптимального) решения, теоретически можно решать перебором всевозможных вариантов и выбором среди них наилучшего. При больших размерностях входных данных перебор всех вариантов (так называемый метод «грубой силы») не будет эффективен по времени. В этом случае можно попробовать решить задачу большей размерности при помощи решения подзадач меньшей размерности. Такой метод решения называется динамическим программированием [1].

Для того, чтобы задача могла быть решена методом динамического программирования должны быть выполнены следующие условия: в задаче можно выделить подзадачи аналогичной структуры меньшего размера; среди выделенных подзадач есть тривиальные, то есть имеющие «малый размер» и очевидное решение; оптимальное решение подзадачи большего размера может быть построено из оптимальных решений подзадач; решения подзадач запоминаются в таблицы, имеющие разумные размеры.

Задачи динамического программирования делятся на два типа. Подсчет количества вариантов решения. В этом случае находят суммарное количество решений подзадач. Поиск оптимального решения (оптимизация целевой функции). В этом случае выбирают лучшее среди всех решений подзадач.

Одномерная динамика. Решение задач

Рассмотрим пример одной из двух указанных выше типов задач на примере задач, решаемых с помощью одномерной динамики, то есть динамики, в которой используется один параметр.

Подсчет количества вариантов решения. Последовательность из 0 и 1.

Требуется подсчитать количество последовательностей длины N , состоящих из 0 и 1, в которых никакие две единицы не стоят рядом.

Если воспользоваться полным перебором, то, например, для $N = 64$ потребуется сгенерировать 2^{64} последовательностей из нулей и единиц и проверить каждую из них на соответствие условию задачи, что неэффективно по времени. Рассмотрим решение задачи с помощью динамического программирования.

Приведем примеры последовательностей, для некоторых небольших чисел N .

Таблица 1

$N = 1$	$N = 2$	$N = 3$	$N = 4$
0	00 01	000, 001 010	0000, 0001, 0010 0100, 0101
1	10	100, 101	1000, 1001, 1010
2 варианта	3 варианта	5 вариантов	8 вариантов

Из табл. 1 видим, что к каждой последовательности длиной $(N-1)$ можно приписать 0 в любом случае и 1 только в том случае, если она заканчивается на 0 (по условию задачи). То есть, каждая последовательность длины $(N-1)$ может быть продолжена либо одним, либо двумя способами.

Обозначим за i длину последовательности. $F(i)$ – количество последовательностей длины i , удовлетворяющих условию задачи.

Присвоим начальные значения динамики: $F[1] = 2$; $F[2] = 3$. Это будет база динамики.

Правило перехода динамики – $F(i+1) = F(i) + F(i-1)$ позволяет получить ответ для большей подзадачи на основе ответов к предыдущим подзадачам.

Ответом будет являться число $F(N)$ [2].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Порублев И.Н., Ставровский А.Б. Алгоритмы и программы. Решение олимпиадных задач. – М.: ООО «И.Д. Вильямс», 2007. – 475с.
2. Национальный исследовательский университет «Высшая школа экономики». Факультет компьютерных наук. Летняя школа по компьютерным наукам. Август. 2016. Параллель С. Шуйкова И.А.

УДК 003.65

Сальников А.П., Магомадов Р.У.

ИМИТОСТОЙКОСТЬ ШИФРА

Рязанский государственный радиотехнический университет,

г. Рязань

Надежность или стойкость шифров определяется объемом работы криптоаналитика, необходимой для их вскрытия. Шифрсистема может служить объектом нападения противника, располагающего разного уровня интеллектуальным и вычислительным потенциалом.

Задача криптоанализа может интересовать некую фирму, обладающую солидным персоналом и оборудованием. Также речь может идти о работе по добыванию информации мощной государственной организа-

цией типа АНБ США. Возможности потенциального противника определяют требования, предъявляемые к надежности шифрования [1].

Целостность данных обеспечивается присоединением к передаваемым данным специального кода (имитовставки), вырабатываемой по секретному ключу. Имитовставка является разновидностью контрольной суммы, то есть некоторой эталонной характеристикой сообщения, по которой осуществляется проверка целостности последнего. Алгоритм формирования имитовставки должен обеспечивать ее зависимость по некоторому сложному криптографическому закону от каждого бита сообщения. Проверка целостности сообщения выполняется получателем сообщения путем выработки по секретному ключу имитовставки, соответствующей полученному сообщению, и ее сравнения с полученным значением имитовставки. При совпадении делается вывод о том, что информация не была модифицирована на пути от отправителя к получателю [1].

Помимо пассивных действий со стороны потенциального противника, состоящих в подслушивании или перехвате передаваемой по каналу связи зашифрованной информации, возможны также его активные действия, состоящие в попытках подмены или имитации сообщения.

Воздействие помех в канале связи на передаваемую информацию также рассматривается как попытка навязывания ложной информации.

Для выявления этого факта предусмотрена имитозащита. Разработано достаточно много ее способов, позволяющих установить приемному устройству вмешательство противника (или помех) в процесс передачи по каналу связи истинного сообщения $x \in X$. Обычно такие способы имитозащиты основаны на использовании и шифров и не обеспечивают стопроцентную гарантию защиты от навязывания ложной информации.

Рассмотрим вариант навязывания информации способом имитозащиты от них. Пусть $X = Y$ и передаются зашифрованные сообщения $y \in Y$. Противник перехватывает передаваемое сообщение « y » и через определенный промежуток времени посылает его в канал связи. Если за этот промежуток времени не менялся ключ на приемном и передающем устройстве, то приемное устройство повторно воспримет ранее передаваемую информацию. Для защиты от таких действий противника используют метки времени (проставляется время отправления в открытом сообщении $x \in X$). В случае, когда отмеченное время отличается от времени принятия сообщения y на величину, превышающую некоторый допустимый предел, приемное устройство делает вывод о принятии ложного сообщения. В некоторых случаях используют секретный алгоритм изменения ключа шифра через короткие промежутки времени. Если задержка в приеме превышает величину этого промежутка, то приемник не расшифрует навязываемое сообщение [2].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Шептура С.В. Интернет-курс по дисциплине «Математические методы защиты информации» // Московская финансово-промышленная академия. М., 2010.

2. Гатченко Н.А., Исаев А.С., Яковлев А.Д. «Криптографическая защита информации» – СПб: НИУИТМО, 2012.-142 с.

УДК. 004.415.2.031.43

Чернышов А.С.

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СИСТЕМЫ КРЕДИТНОГО БРОКЕРА

Рязанский государственный радиотехнический университет,

г. Рязань

В статье рассматриваются аспекты разработки
кредитного брокера.

Ввиду большого роста многообразия финансово-кредитных организаций, оказывающая кредитные услуги, потребителю становится всё сложнее в них ориентироваться. Вследствие чего возросла роль посредников, которые способны помочь клиенту с поиском лучших предложений.

Одним из таких посредников является кредитный брокер, который взаимодействует с клиентами, банками, а также кредитными бюро, которые предоставляют дополнительную информацию о потенциальном заемщике.

В данной статье рассматривается ряд аспектов проектирования автоматизированной системы, целью которой является помочь взаимодействующим клиентам, желающим получить кредит, и банкам, готовых этот кредит предоставить.

Бизнес процесс можно описать следующим образом:

– клиент обращается в брокерскую контору, передает все необходимые документы, сообщает желаемую сумму, срок погашения и согласовывает все детали.

– после получения всей информации кредитный брокер обращается в кредитные бюро, которые предоставляют дополнительную информацию о клиенте, а также дают сформированную рейтинг-оценку заемщика.

– брокер делает запрос банкам, которым может быть интересен данный клиент, исходя из его кредитной истории, суммы кредита, срока погашения, рейтинга и других интересующих банков параметров.

– по истечению некоторого времени, брокер анализирует полученные предложения от банков, выявляя самые выгодные условия. После чего брокер связывается с клиентом, который принимает окончательное решение, на основе предложенных альтернатив.

Схему взаимодействия можно представить следующим образом:

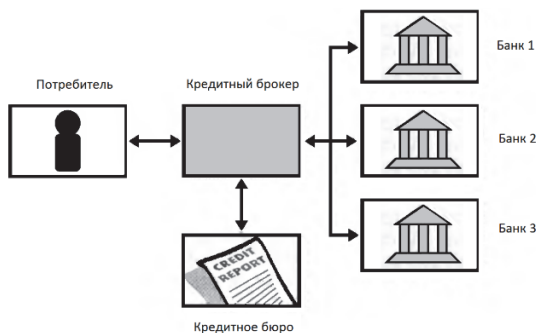


Рис. 1. Общая схема кредитного брокера

Рассмотрим два основные реализации режима работа брокера с банками:

Синхронный (последовательный). Брокер передает данные клиента последовательно каждому банку, при этом запрос к следующему банку выполняется при получении ответа от текущего.

Асинхронный (параллельный). Брокер одновременно делает запросы во все банки и ждет, пока они не пришлют ответы.

Синхронный режим подразумевает последовательную обработку всех запросов на получение кредита. Данный режим проще реализовать, и он более гибкий в плане управления, так как не связан с параллельными потоками и другими нюансами одновременной обработки данных. С другой стороны, такое решение весьма неэффективно, потому что банки обладают **независимыми вычислительными ресурсами**, в связи с этим ответ от банка может сильно затянуться, что затормозит весь процесс в общем.

Асинхронный режим предполагает одновременную отправку всех запросов сразу. В связи с этим, во-первых, все банки могут приступить к обработке сразу, а во-вторых нет необходимости ждать ответа от конкретного банка. Указанное решение обеспечивает намного более высокую скорость обработки данных.

Таким образом, весь процесс можно реализовать следующим образом.

Кредитный брокер получает сумму и срок кредита с помощью метода `GetLoanQuote` (сумма, срок), после чего получает от кредитного бюро кредитный бал (метод `GetCreditScore`). Затем **асинхронно** вызывается метод **`GetLoanQuote` (сумма, срок, балл)**, который делает запрос в соответствующие банки, после получения ответа, кредитный брокер выбирает лучшую ставку и возвращает ее. Таким образом, весь процесс можно реализовать следующим образом.

Если запросы отправляются в N банков, и каждый банк тратит одинаковое время на выполнение запроса, то данное решение будет выполняться в N раз быстрее чем аналогичное решение с использованием синхронной обработки. Отметим, что поскольку ответы от банков могут приходить в людей последовательно, необходимо, что брокер имел возможность обрабатывать в любом порядке.

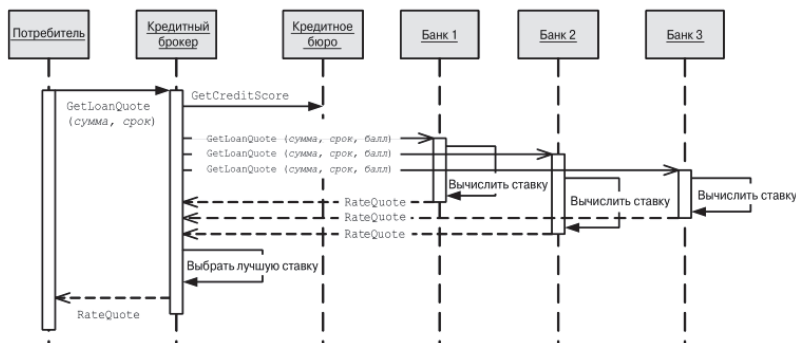


Рис. 2. Асинхронная, параллельная обработка запросов на получение кредита

Таким образом, при реализации программ, которые взаимодействуют с независимыми вычислительными ресурсами, рационально использовать именно асинхронные запросы, в противном случае, система будет сильно зависеть от сторонних ресурсов, и, если они окажутся узким местом, система будет крайне неэффективной.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Грегор Хоп, Бобби Вульф. Шаблоны интеграции корпоративных приложений. – М: Вильямс, 2016. – 672 с.

УДК: 004.438

Васильев И. В.

РАЗРАБОТКА КОНФИГУРАЦИИ 1С ДЛЯ САНАТОРИЯ

Рязанский государственный радиотехнический университет,

г. Рязань

В данной статье рассмотрена конфигурация 1С для управления санаторием.

В настоящее время большую популярность в российском сегменте автоматизации деятельности на предприятии завоевал программный продукт 1С:Предприятие. Мною была разработана конфигурация «1С:Управление санаторием», с помощью которой можно автоматизировать рабочие места:

– службы бронирования и размещения, администратора отеля;

- службы бронирования услуг;
- управляющего отеля, санатория, гостиницы, дома отдыха.

ИС: Управление санаторием позволяет одновременно автоматизировать различные блоки оперативного учета на предприятии (отельный, медицинский).

Для управления отелем санатория создана подсистема [1]. Отель. Здесь пользователь может управлять номерами отеля, устанавливать цены на номера определённого типа, а также, записывать посетителей в номера. Для управления услугами санатория создана подсистема Услуги, в которой пользователь может создавать услуги с определённой ценой и записывать посетителей на необходимую услугу. В подсистеме Управление есть справочник болезней, корпусов отеля санатория, материально-технической базы, а также посетителей.

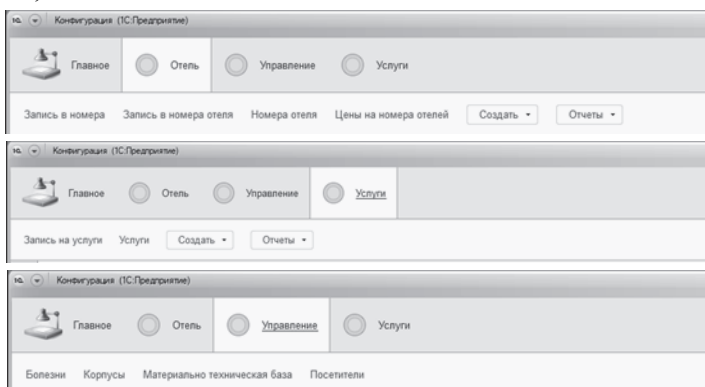


Рис. 1. Внешний вид подсистем

В системе также есть отчеты в виде диаграмм [2] по записям в отель и на услуги, с помощью которых можно проанализировать, в какие месяцы доход поднимался, а в какие – падал.

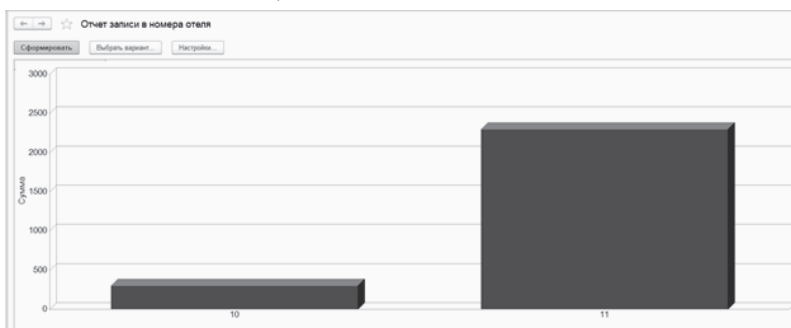


Рис. 2. Отчет по записи в отель

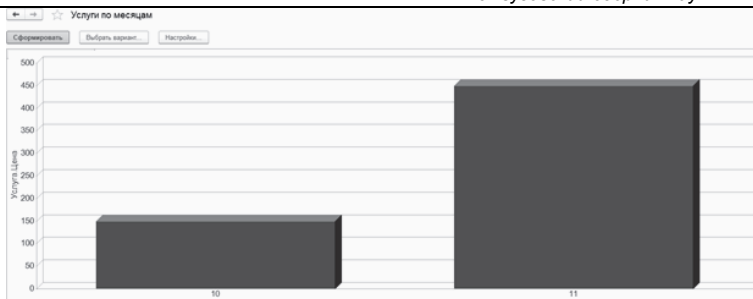


Рис. 3. Отчет по записи на услуги

Таким образом, разработанная конфигурация позволяет вести учет и управлять санаторием. Она является мощным инструментом для пользователей, избавляя их от лишних действий при работе с системой.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Хрусталёва Е.Ю. 101 совет начинающим разработчикам в системе «1С:Предприятие 8» – М.: 1С-Паблишинг, 2015. – С 156-193. – 214 с. – ISBN 978-5-9677-2322-3
2. Хрусталёва Е.Ю. Разработка сложных отчетов в «1С:Предприятии 8». Система компоновки данных – 7-е изд. – М.: 1С-Паблишинг, 2012 – С. 309-351. – 458 с. – ISBN 978-5-9677-1698-0.

УДК 004.91

Беляев Д.Ю., Дмитриева Т.А.

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ РЕШЕНИЯ ПРОБЛЕМЫ РУЧНОГО СОСТАВЛЕНИЯ ОТЧЕТНЫХ ДОКУМЕНТОВ

Рязанский государственный радиотехнический университет»,

г. Рязань

Обоснована необходимость разработки программного средства автоматизирующего составление отчетных документов.

Проблема ручного оформления отчетных документов является актуальной. Для того чтобы составить отчетный документ, человеку необходимо проанализировать необходимые ему данные, найти эти данные в различных источниках, собрать найденные данные в один документ, отредактировать, проверить. Этот процесс занимает очень много времени. Также необходимо учитывать человеческий фактор, так как существует вероятность того, что человек может ошибиться. Необходимо разработать программное обеспечение, которое максимально автоматизирует данный процесс и избавит от возникающих проблем. Процесс работы должен быть организован таким образом, чтобы человеку достаточно было заполнить необходимый ему шаблон документа данными, хранящимися в программе,

проверить созданный документ и распечатать его. Данное программное обеспечение существенно уменьшит время, затрачиваемое на составление отчетных документов.

Есть несколько способов решения данной проблемы. Одним из способов является разработка клиент-серверного приложения [1]. В этом случае необходимо спроектировать базу данных, в которой хранятся данные, которые требуются для формирования отчетных документов. Пользователь имеет возможность изменения и добавления новых данных, он выбирает необходимые данные, на основе которых формируются отчетные документы.

Второй способ решения этой проблемы – создание конфигурации 1С [2]. Создание конфигурации 1С является наиболее рациональным способом решения данной проблемы. Это связано с тем, что затрачивается намного меньше времени на разработку данного решения по сравнению с разработкой клиент-серверного приложения. Не нужно создавать базу данных, так как 1С Предприятие имеет встроенную базу данных (файловый вариант), также можно использовать клиент-серверные варианты (подключение базы данных Microsoft SQL Server, Oracle Database). Есть возможность создавать прикладные объекты конфигурации (справочники, документы, константы, перечисления, регистры и др.), которые будут содержать все необходимые в создании отчетных документов данные. На основе документов создаются их печатные формы при помощи встроенного конструктора печатных форм, в котором выбираются необходимые реквизиты и автоматически формируется макет печатной формы, который можно отредактировать и получить готовый отчетный документ [3]. В 1С намного проще расширить функционал созданной конфигурации или внедрить в другую конфигурацию, что делает разработку многофункциональной.

Подводя итоги можно сказать, что для решения проблемы ручного оформления отчетных документов лучше использовать 1С:Предприятие, основными достоинствами которого являются:

- 1) низкая сложность разработки;
- 2) время, затрачиваемое на разработку;
- 3) расширение функционала;
- 4) внедрение в другие конфигурации.

Таким образом, в статье рассмотрена необходимость разработки программного обеспечения, автоматизирующего процесс составления отчетных документов, проведен анализ возможных средств разработки и обоснован выбор в качестве средства разработки платформы 1С:Предприятие.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Дэйт К.Дж. Введение в системы баз данных – М.: Вильямс, 2018.

2. Основы конфигурирования в системе «1С. Предприятие 8» – М.: Интернет-Университет Информационных Технологий, 2016.

3. Хрусталева Е.Ю. Разработка сложных отчетов в «1С:Предприятия 8». Система компоновки данных – М.: ООО «1С-Паблишинг», 2016.

УДК 519.72

Ермаков Д.С.

АЛГОРИТМЫ КОДИРОВАНИЯ И ДЕКОДИРОВАНИЯ ПО ХЭММИНГУ

Рязанский государственный радиотехнический университет,

г. Рязань

В данной статье рассматриваются алгоритмы кодирования и декодирования кодов Хэмминга.

Кодом Хэмминга называется (n, k) – код, который задается матрицей проверок $H(n, k)$, имеющей строк и столбцов, причем столбцами $H(n, k)$ являются все различные ненулевые двоичные последовательности длины m .

Для описания алгоритма кодирования нужно сгенерировать код Хемминга для некоторого информационного кодового слова. Как пример возьмём 15-битовое кодовое слово $x_1 \dots x_{15}$.

Вставим в информационное слово проверочные биты $r_0 \dots r_4$ таким образом, чтобы номера их позиций представляли собой целые степени двойки: 1, 2, 4, 8, 16... Получим 20-разрядное слово с 15 информационными и 5 проверочными битами. Первоначально проверочные биты устанавливаем равными нулю.

В общем случае количество проверочных бит в кодовом слове равно двоичному логарифму числа бит кодового слова (включая проверочные биты), округлённому в большую сторону до ближайшего целого. Например, информационное слово длиной 1 или 2 бита требует двух проверочных разрядов, 3- или 4-битовое информационное слово – трёх, 5...11-битовое – четырёх, 12...26-битовое – пяти и т.д.

Добавим к таблице пять строк (по количеству проверочных битов), в которые поместим матрицу преобразования. Каждая строка будет соответствовать одному проверочному биту (нулевой проверочный бит – верхняя строка, четвёртый – нижняя), каждый столбец – одному биту кодируемого слова. В каждом столбце матрицы преобразования поместим двоичный номер этого столбца, причём порядок следования битов будет обратный – младший бит расположим в верхней строке, старший – в нижней. Например, в третьем столбце матрицы будут стоять числа 11000, что соответствует двоичной записи числа три: 00011.

В правой части таблицы мы оставили пустым один столбец, в который поместим результаты вычислений проверочных битов. Вычисление проверочных битов производим следующим образом. Берём одну из строк матрицы преобразования (например, r_0) и находим её скалярное произведение с кодовым словом, то есть перемножаем соответствующие биты обеих строк и находим сумму произведений. Если произведение

получилось больше единицы, находим остаток от его деления на 2. Иными словами, мы подсчитываем сколько раз в кодовом слове и соответствующей строке матрицы в одинаковых позициях стоят единицы и берём это число по модулю 2.

Если описывать данный процесс в терминах матричной алгебры, то операция представляет собой перемножение матрицы преобразования на матрицу-столбец кодового слова, в результате чего получается матрица-столбец проверочных разрядов, которые нужно взять по модулю 2. Полученные проверочные биты вставляем в кодовое слово вместо стоявших там ранее нулей.

Алгоритм декодирования по Хэммингу абсолютно идентичен алгоритму кодирования. Матрица преобразования соответствующей размерности умножается на матрицу-столбец кодового слова и каждый элемент полученной матрицы-столбца берётся по модулю 2. Полученная матрица-столбец называется «матрица синдромов». Легко проверить, что кодовое слово, сформированное в соответствии с алгоритмом, описанным в предыдущем разделе, всегда даёт нулевую матрицу синдромов. Матрица синдромов становится ненулевой, если в результате ошибки (например, при наличии шумов) один из битов исходного слова изменил своё значение.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Питерсон У., Уэлдон Э. Коды, исправляющие ошибки: Пер. с англ. М.: Мир, 1976, – 600 с.
2. Блейхут Р. Теория и практика кодов, контролирующих ошибки. Пер. с англ. М.: Мир, 1986, – 576 с.

УДК 004.421+004.424

Дмитриева Т.А., Палагин Д.А.

РАЗРАБОТКА ИНТЕРАКТИВНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ОБУЧЕНИЯ АЛГОРИТМИЧЕСКОМУ ЯЗЫКУ

Рязанский государственный радиотехнический университет,
г. Рязань

Обоснована необходимость разработки интерактивного программного средства для обучения алгоритмическому языку посредством демонстрации работы программы.

При обучении школьников и студентов любому алгоритмическому языку программирования [1] возникает необходимость объяснения как работает программа, как изменяются значения переменных. Кроме того, необходимо продемонстрировать соответствие схемы алгоритма, которая визуализирует наиболее значимые части программы, и операторов программы. Чем сложнее алгоритм, тем больше времени занимает процесс изображения его схемы. На настоящий момент существует мно-

жество программных средств, позволяющих автоматизировать процесс получения схем алгоритмов для программ, написанных на различных языках программирования (FCEditor, FCMaker, Draw.io), однако эти программные средства не имеют другого необходимого функционала, который описан выше.

В связи с этим возникает потребность в разработке интерактивного программного обеспечения обучения алгоритмическому языку, которое должно соответствовать следующим требованиям.

1. Иметь возможность построения схем алгоритмов.
2. Поддерживать большинство распространённых алгоритмических языков программирования.
3. Иметь наглядную визуализацию пошагового исполнения программного кода.
4. Возможность конвертировать программный код с одного языка программирования на другой.
5. Иметь возможность расширяемого функционала.
6. Автономность – не должна зависеть от доступа к интернету.

Подобная программа, в первую очередь, должна автоматизировать работу преподавателей, обучающих алгоритмическим языкам программирования, в изображении схем алгоритмов по коду программы и демонстрации работы программы.

Программное обеспечение должно предусматривать следующие управляющие элементы.

1. *Основное меню.* В этом элементе должен быть представлен перечень возможных функций, реализуемых в программе: перевод кода в другой язык программирования, интерпретация схем алгоритмов, учебный материал, представление алгоритма кода программы по шагам. При выборе пункта меню, должно открываться соответствующее подменю.

2. *Меню перевода кода.* В этом элементе должно быть пользовательское окно ввода кода программы с возможностью загрузить из файла и изменить содержимое, выпадающее меню выбора языка программирования, в котором представлен исходный код, выпадающее меню выбора языка программирования конечного кода, окно вывода результата с возможностью сохранить в файл.

3. *Меню построения схем алгоритмов.* В этом элементе должно быть пользовательское окно ввода кода программы с возможностью загрузить из файла и изменить содержимое, выпадающее меню выбора языка программирования, в котором представлен исходный код, форма, на которой будет отображаться готовая схема алгоритма с возможностью сохранения в файл.

4. *Меню учебного материала.* В этом элементе должно быть выпадающий список с выбором языка программирования, необходимого для обучения. После выбора пункта списка, должен появляться на форме список тем для изучения.

5. Меню пошагового выполнения кода. В этом элементе должно быть пользовательское окно ввода кода программы с возможностью загрузить из файла, кнопка начала работы, которая выделит первую строку кода программы, кнопка перехода к следующему шагу, кнопка перехода к предыдущему шагу исполняемого кода. Последовательность шагов задается пользователем.

Разработанное программное обеспечение будет иметь следующие преимущества:

- 1) автоматизация типичных действий;
- 2) снижение человеческого фактора;
- 3) сокращение затраченного времени на рисование схем алгоритмов;
- 4) демонстрация ошибок логики в кодах программ учащихся.

Данный продукт может найти свое применение в различных образовательных учреждениях – школах, ВУЗах, специализированных курсах по обучению алгоритмическим языкам, например в городской школе программистов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Докукина Т.К. Программирование и алгоритмические языки. – М.: Машиностроение, 2002.

УДК 004.9

Щелконогов А.Н.

РАЗРАБОТКА ПРОСТЕЙШИХ НЕЙРОСЕТЕЙ В KERAS

Рязанский государственный радиотехнический университет,

г. Рязань

В данной статье рассматриваются принципы разработки простейших нейросетей в Keras.

Искусственные нейронные сети (для краткости нейросети или просто сети) – это класс моделей машинного обучения, в основе которых лежат исследования центральной нервной системы млекопитающих. Нейросеть состоит из нескольких взаимосвязанных нейронов, организованных в слои, которые обмениваются между собой сообщениями (как говорят, «возбуждаются») при выполнении определенных условий.

В настоящее время разработчики программного обеспечения при создании собственных нейросетей, в том числе, глубоких, могут использовать богатые возможности активно развивающейся библиотеки Keras [1].

Для разработки простейшей нейросети, которая распознает рукописные цифры, можно использовать набор данных MNIST, включающих 60000 обучающих и 10000 тестовых примеров, и библиотекой Keras, которая предоставляет средства для загрузки набора данных и разбиения его на обучающий набор X_{train} и тестовый набор X_{test} .

Для поддержки вычислений на GPU следует преобразовать данные к типу *float32* и пронормировать их в интервал $[0,1]$.

В наборы *Y_train* и *Y_test* следует загрузить правильные метки и применяем к ним унитарное кодирование.

Нормировка значений, ассоциированных с пикселями, выполняется с целью приведения этих значений к диапазону $[0, 1]$ (это значит, что яркость каждого пикселя делится на максимально возможную яркость 255).

Последний слой нейросети состоит из единственного нейрона с функцией активации *softmax*, являющейся обобщением сигмоиды. *Softmax* сплющивает *k*-мерный вектор, содержащий произвольные вещественные числа, в *k*-мерный вектор вещественных чисел из интервала $[0, 1]$.

При разработке нейросети необходимо выбрать оптимизатор, то есть конкретный алгоритм, который будет обновлять веса в процессе обучения модели.

Кроме того, необходимо выбрать целевую функцию, которую оптимизатор использует для навигации по пространству весов (часто целевая функция называется также функцией потерь, а процесс оптимизации – минимизацией потери).

Для этого следует определить, какие метки надо предсказать. Например, такая целевая функция, как «бинарная перекрестная энтропия», подходит для предсказания бинарных меток, а функция «категориальная перекрестная энтропия» используется для предсказания многоклассовых меток.

Для оценки качества обученной модели могут использоваться различные показатели, например, такие, как верность (отношение числа правильных предсказаний к общему числу меток), точность (доля правильных ответов модели), полнота (доля обнаруженных истинных событий). Показатели качества похожи на целевые функции. При этом показатели качества используются не для обучения модели, а для оценки ее качества. Определенную таким образом модель необходимо откомпилировать, т.е. привести к виду, допускающему исполнение базовой библиотекой (Theano или TensorFlow).

Для обучения откомпилированной модели используется функция *fit()*. При этом следует задать значения таких параметров, как *epochs* (число периодов, определяющее, сколько раз обучающий набор предъявляется модели) и *batch_size* (число, определяющее, сколько обучающих примеров должен увидеть оптимизатор, прежде чем он обновит веса). При этом оптимальные значения этих параметров определяются экспериментально (с ориентацией на достижение экстремума целевой функции).

После того как модель обучена, ее следует проверить на тестовом наборе, который содержит ранее не предъявлявшиеся примеры. Обучающий и тестовый наборы данных не должны пересекаться, так как нет никакого смысла оценивать модель на примере, который она видела во

время обучения. Смысл обучения в том, чтобы модель могла обобщаться на ранее не встречавшиеся данные, а не в том, чтобы она запоминала то, что уже известно.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Библиотека Keras – инструмент глубокого обучения. Реализация нейронных сетей с помощью библиотек Theano и TensorFlow / пер. с англ. Слинкин А. А. – М.: ДМК Пресс, 2018. – 294 с.: ил.

УДК 004.624

Дмитриева Т.А., Колунтаева Е.С.

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ИМПОРТА ДАННЫХ О СОТРУДНИКАХ ИЗ EXCEL В 1С

Рязанский государственный радиотехнический университет,
г. Рязань

Обоснована необходимость разработки программного средства для импорта данных о сотрудниках предприятия из Excel в 1С.

Для рассматриваемого нами предприятия имеются следующие исходные данные: на предприятии формируется документ «Зарплата ведомость.xls» в Microsoft Office Excel данные из которого необходимо использовать в типовой конфигурации 1С:Предприятие в документе планирования [1]. Необходимо загрузить в 1С из Excel такие данные о сотрудниках, как: ФИО, оклад и премиальную часть за месяц с разбивкой по статьям.

Процесс внесения в документ планирования большого объема информации вручную пользователи сотрудниками планово-экономического отдела является очень трудозатратным. Кроме того, это довольно монотонная работа, которая требует большой концентрации внимания и очень часто приводит к возникновению технических ошибок в документах. Число элементов, которые необходимо вводить составляет сотни или даже тысячи. Данные требуется загружать постоянно в силу особенностей технологии работы планово-экономического отдела рассматриваемого предприятия. Именно поэтому задача автоматизации данного процесса является актуальной и практически значимой.

На предприятии информация о сотрудниках вводится в таблицы Microsoft Excel. С помощью Excel удобнее работать с информацией (копировать, переносить и т.д.), чем в форме создания документа планирования в 1С. При переносе данных в Excel необходимо соблюдать следующее условие – заголовки таблиц должны совпадать с заголовками в документах планирования программы 1С: Предприятие. Также после загрузки данных в 1С необходима возможность конфигурации загруженной таблицы для отладки нумерации столбцов, строк и т.п.

Для решения задачи необходимо написать средствами 1С обработку [2], при помощи которой таблицы с отобранной информацией будут подгружаться в программу 1С:Предприятие. При использовании разработанного программного обеспечения будет сокращено время импорта данных (поскольку с помощью Excel обработать данные можно гораздо быстрее, чем изначально в 1С), а также исключены ошибки, связанные с ручным переносом данных из одной программы в другую.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Гартвич А.В. 1С:Предприятие 8.3. Практическое планирование закупок, производства и продаж в 1С:Предприятие 8. – М.: ООО «1С-Паблишинг», 2007.
2. Радченко М.Г., Хрусталева Е.Ю. 1С:Предприятие 8.3. Практическое пособие разработчика Примеры и типовые приемы. – М.: ООО «1С-Паблишинг», 2013.

УДК 004.8

Перевезенцев Е.Е., Ромашкова В.В.

ГИБРИДНЫЕ ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

Рязанский государственный радиотехнический университет,

г. Рязань

В данной статье рассматриваются гибридные интеллектуальные системы и понятие проблемно-структурной методологии функциональных гибридных интеллектуальных систем.

«Гибрид» – система, состоящая из интегрированных подсистем, каждая из которых может иметь различные языки представления и методы вывода. Подсистемы объединяются семантически и по действию каждая с каждой. Под гибридной интеллектуальной системой принято понимать систему, использующую более одного метода имитации интеллектуальной деятельности человека.

Методология гибридных интеллектуальных систем – учение о структуре, логической организации, методах и средствах деятельности разработчика гибридных интеллектуальных систем. Методологическое знание выступает в форме предписаний и норм, фиксирующих содержание и последовательность (организацию) определенных видов деятельности, и описаний фактически выполняемой деятельности. Основная функция знания – внутренняя организация и регулирование процесса познания и анализа сложных задач, синтеза и эксплуатации гибридов.

Разработку гибридных интеллектуальных систем принято называть гибридизацией. Опыт показывает, что на разработку «вручную» функциональной гибридной интеллектуальной системы, решающей сложную задачу, затрачивается 3-5 человеко-лет. Выполняя гибридизацию, разработчик имеет дело с моделью сложной задачи (объектом-оригиналом), моделями методов решения задач (объектами-прототипами), моделью

гибридной интеллектуальной системы (объектом-результатом). Первые два объекта – источники информации для гибридизации, а последний – основа формообразования, программирующая то, что должно появиться после трансформации информации [1].

Проблемно-структурная методология – учение о структуре, логической организации, принципах, методах и средствах деятельности разработчика, выполняющего гибридизацию, объект-оригинал которой – сложная задача, объекты-прототипы – методы формализованного представления систем, объект-результат – функциональная гибридная интеллектуальная система. Так как объект-оригинал – сложная задача, а объект-результат – функциональная гибридная интеллектуальная система, структура которых зависит от состава и взаимосвязей подзадач из декомпозиции сложной, неоднородной задачи, методология названа проблемно-структурной (ПС).

Рассмотрим этапы разработки функциональной гибридной интеллектуальной системы и «жизненный цикл» [1].

1) «Идентификация сложной задачи». В соответствии со схемой ролевых концептуальных моделей определяются цель, исходные данные для решения задачи, ее идентификатор и спецификатор.

2) «Редукция сложной задачи и построение множества ее декомпозиций». Продолжаются работы по системному анализу сложной задачи. Акценты смещаются на ее состав и структуру. Исходная задача-система редуцируется на множество однородных подзадач, и строится множество возможных декомпозиций.

3) «Спецификация подзадач». Каждая подзадача идентифицируется в соответствии со схемой ролевых концептуальных моделей, определяются цель решения и исходные данные, классификатор и частичноспецификатор. Процедура неформальная и выполняется разработчиком.

4) «Выбор методов». Работы выполняются для построенных декомпозиций и специфицированных подзадач. Класс метода формализованного представления системы выбирается модельером с помощью автоматизированной системы с базой знаний.

5) «Проверка неоднородности» выполняется, чтобы повысить у модельера уверенность в том, что сложная задача удовлетворяет модели «неоднородная задача». Заключение о неоднородности исходной задачи делает модельер на основании оценок по выбору методов для каждой подзадачи. Если для всех подзадач оценки в пользу одного итого же класса метода формализованного представления системы будут близки друг к другу, то можно считать, что сложная задача удовлетворяет модели «однородная задача». В этом случае работы следует продолжить в рамках одной из методологий автономного моделирования. Если в декомпозиции обнаружены области неоднородности, то это указывает на необходимость гибридизации.

6) «Разработка автономных моделей». В соответствии с методологией элементов функциональной гибридной интеллектуальной системы разрабатывается гетерогенное модельное поле для решения подзадач из декомпозиции сложной задачи. Если такие модели имеются, они включаются в это множество. Работы идут согласно методологии автономных моделей и носят итерационный характер.

7) «Спецификация областей релевантности моделей». Актуальность такой спецификации объясняется альтернативностью (функциональной избыточностью) инструментариев решения подзадачи. На выбор альтернативы влияют параметры внешней среды. Такие параметры и характеристики названы существенными, а множества их значений – областями релевантности методов (моделей).

8) «Спецификация связей сложных задач». Заканчивается спецификация сложной задачи с учетом обмена информацией с другими сложными задачами определением отношений на множестве сложных задач.

9) «Разработка интерфейсов автономных моделей». Интерфейсы позволяют устанавливать отношения интеграции на гетерогенном модельном поле. Их необходимость объясняется различиями языков описания в схемах ролевых концептуальных моделей метода формализованного представления системы. Для разработки интерфейсов используются адаптируемые программные утилиты, сокращающие трудозатраты на организацию межпрограммного обмена.

10) «Разработка таблицы гибридных стратегий». Выполняется замена отношений декомпозиции на отношения интеграции, и устанавливаются отношения соответствия между исходными данными и целями сложной задачи и подзадач. Работы заканчиваются построением таблицы гибридных стратегий, отображающей связи между подзадачами.

11) «Выбор стратегии из таблицы гибридных стратегий» Эксперт выбирает стратегию из таблицы гибридных стратегий для синтеза функциональной гибридной интеллектуальной системы. Принятие решений отражает ситуацию, когда сложная задача имеет изменчивые состав и структуру и должна решаться всякий раз заново.

12) «Инициализация гибридной интеллектуальной системы по выбранной стратегии». В режиме взаимодействия с пользователем инициализируется функциональная гибридная интеллектуальная система, интерпретация которой – метод решения сложной задачи. Для синтеза архитектуры гибридной интеллектуальной системы разработаны эвристические алгоритмы, суть которых – преобразование гибридной стратегии, заданной матрицей «подзадача – отношения интеграции – подзадача», в список «модель – отношение интеграции – модель».

Этот список, дополненный перечнем необходимых интерфейсных утилит, – исходная информация для инициализации архитектуры гибридной интеллектуальной системы.

13) «Машинные эксперименты и интерпретация результатов». Выполняются модельные эксперименты, в результате которых, в определенном декомпозицией порядке, имитируется решение подзадач и получается результирующая информация. Если результаты решения устраивают пользователя, то он может не выполнять последующие этапы. Однако предусмотрены и другие возможности.

14) «Формирование множества альтернатив» Полученное решение заносится в список альтернатив, и синтез повторяется с другими оценками пользователем альтернативных моделей из гетерогенного модельного поля. Таким итерационным процессом можно сформировать множество альтернативных решений. Если оно содержит более 7-10 элементов, которые не могут быть обработаны пользователем вручную, то предусмотрен этап совершенствования.

15) «Совершенствование множества альтернатив и анализ результатов». Альтернативы рассматриваются как популяция индивидумов, эволюция которой имитируется генетическим алгоритмом. Результаты анализируются пользователем, спецификатор неоднородной задачи пополняется информацией об операции, реализующей принятую альтернативу. В результате заполняется эвалюэитор – схема, определяющая оценки результатов решения однородных задач из состава неоднородной.

16) «Эксплуатация и модернизация». Базы данных проекта и базы знаний элементов функциональной гибридной интеллектуальной системы поддерживаются в актуальном состоянии. В случае изменений в знаниях одного из элементов совершенствуется гетерогенное модельное поле.

Рассмотренную методологию отличает:

1) применение к сложным задачам, которые подвергаются всестороннему изучению с использованием методов системного анализа на моделях «неоднородная задача» и «однородная задача»;

2) применение оригинального информационного языка функциональных гибридных интеллектуальных систем для представления объектов-оригиналов, объектов-прототипов и объектов-результатов гибридизации;

3) применение оригинального, основанного на знаниях метода подбора метода формализованного представления системы для решения подзадач из декомпозиции неоднородной задачи;

4) применение схем концептуальных моделей, направляющих деятельность разработчика и позволяющих избежать множества ошибок;

5) разработка гетерогенного модельного поля для сложной задачи, в рамках которого строится метод ее решения; применение оригинального, эвристического алгоритма синтеза метода решения сложной задачи;

6) применение оригинального: эвристического алгоритма принятия решения о выборе моделей из гетерогенного модельного поля в соответствии с областями релевантности метода формализованного представления системы.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Колесников А.В., Кириков И.А., Листопад С.В., Румовская С.Б., Доманицкий А.А. Решение сложных задач коммивояжера методами функциональных гибридных интеллектуальных систем / Под ред. А.В. Колесникова. – М.: ИПИ РАН, 2011. – 295с.

УДК 519.81

Титов Д.Д.

АЛГОРИТМЫ ВЫЧИСЛЕНИЯ ЦЕНТРОИДОВ ИНТЕРВАЛЬНЫХ НЕЧЕТКИХ МНОЖЕСТВ ВТОРОГО ТИПА

Рязанский государственный радиотехнический университет,
г. Рязань

Рассмотрены алгоритмы вычисления центроидов интервальных нечетких множеств второго типа и даны рекомендации по их применению.

В данной работе рассматриваются три алгоритма вычисления центроидов интервальных нечетких множеств второго типа: алгоритм Карника-Менделя (КМ), улучшенный алгоритм Карника-Менделя и метод сокращения типа замкнутой формы.

В общем виде проблему вычисления таких центроидов можно представить в виде:

Пусть дано

$$x_i \in X_i \equiv [\underline{x}_i, \bar{x}_i], \quad i = 1, 2, \dots, N,$$

$$w_i \in W_i \equiv [\underline{w}_i, \bar{w}_i], \quad i = 1, 2, \dots, N,$$

где $\underline{x}_i \leq \bar{x}_i$, $\underline{w}_i \leq \bar{w}_i$, $i = 1, 2, \dots, N$.

Необходимо вычислить

$$C = \frac{\sum_{i=1}^N X_i W_i}{\sum_{i=1}^N W_i} \equiv [c_l, c_r],$$

где

$$c_l = \min_{\substack{\forall x_i \in [\underline{x}_i, \bar{x}_i] \\ \forall w_i \in [\underline{w}_i, \bar{w}_i]}} \sum_{i=1}^N x_i w_i / \sum_{i=1}^N w_i,$$

$$c_r = \max_{\substack{\forall x_i \in [\underline{x}_i, \bar{x}_i] \\ \forall w_i \in [\underline{w}_i, \bar{w}_i]}} \sum_{i=1}^N x_i w_i / \sum_{i=1}^N w_i.$$

Последние два уравнения могут быть переписаны в виде [1]

$$c_l = \min_{\forall w_i \in [\underline{w}_i, \bar{w}_i]} \frac{\sum_{i=1}^N \underline{x}_i w_i}{\sum_{i=1}^N w_i} = \frac{\sum_{i=1}^L \underline{x}_i \bar{w}_i + \sum_{i=L+1}^N \underline{x}_i \underline{w}_i}{\sum_{i=1}^L \bar{w}_i + \sum_{i=L+1}^N \underline{w}_i}, \quad (1)$$

$$c_r = \min_{\forall w_i \in [\underline{w}_i, \bar{w}_i]} \frac{\sum_{i=1}^N \bar{x}_i w_i}{\sum_{i=1}^N w_i} = \frac{\sum_{i=1}^R \bar{x}_i \underline{w}_i + \sum_{i=R+1}^N \bar{x}_i \bar{w}_i}{\sum_{i=1}^R \underline{w}_i + \sum_{i=R+1}^N \bar{w}_i}. \quad (2)$$

где L и R являются точками переключения.

Алгоритмы Карника-Менделя

I. Алгоритм КМ для вычисления c_l [1], [2]

1) Инициализируем w_i , установив

$$w_i = \frac{\underline{w}_i + \bar{w}_i}{2}, \quad i = 1, 2, \dots, N,$$

а затем вычислим

$$c = \frac{\sum_{i=1}^N \underline{x}_i w_i}{\sum_{i=1}^N w_i}.$$

2) Найдем точку переключения k ($1 \leq k \leq N-1$) такую, что

$$\underline{x}_k \leq c \leq \underline{x}_{k+1}.$$

3) Положим

$$w_i = \begin{cases} \bar{w}_i, & i \leq k \\ \underline{w}_i, & i > k \end{cases}$$

и вычислим

$$c' = \frac{\sum_{i=1}^N \bar{x}_i w_i}{\sum_{i=1}^N w_i}. \quad (3)$$

4) Проверим $c = c'$, если верно, то останавливаемся, присваиваем $c_l = c$ и $L = k$. Если нет, то переходим к шагу 5.

5) Присвоим y значение c' и возвратимся к шагу 3.

II. Алгоритм КМ для вычисления c_r [1], [2]

1) Отсортируем $\bar{x}_i$ ($i = 1, 2, \dots, N$) в порядке возрастания. Сопоставим веса w_i в соответствии с $\bar{x}_i$ и пронумеруем их так, чтобы их индексы совпадали с индексами $\bar{x}_i$.

2) Инициализируем w_i , установив

$$w_i = \frac{w_i + \bar{w}_i}{2}, \quad i = 1, 2, \dots, N,$$

а затем вычислим

$$c = \frac{\sum_{i=1}^N x_i \bar{w}_i}{\sum_{i=1}^N w_i}.$$

3) Найдем точку переключения k ($1 \leq c \leq N-1$) такую, что

$$\bar{x}_k \leq c \leq \bar{x}_{k+1}.$$

4) Положим

$$w_i = \begin{cases} \bar{w}_i, & i \leq k \\ \underline{w}_i, & i > k \end{cases}$$

и вычислим

$$c' = \frac{\sum_{i=1}^N x_i w_i}{\sum_{i=1}^N w_i}.$$

5) Проверим $c = c'$, если верно, то останавливаемся, присваиваем $c_r = c$ и $R = k$. Если нет, то переходим к шагу 5.

6) Присвоим y значение c' и возвратимся к шагу 3.

Было доказано, что алгоритмы КМ сходятся монотонно и суперэкспоненциально быстро [3]; однако до сходимости все еще требуется несколько (обычно от двух до шести) итераций. Как указано в [3], «открытой задачей является поиск оптимального способа инициализации алгоритма КМ», оптимального в том смысле, что суперэкспоненциальный коэффициент сходимости минимизируется.

Улучшенные алгоритмы Карника-Менделя

Улучшенный алгоритм КМ очень похож на предыдущий. Он также состоит из двух отдельных для вычисления c_l и c_r . Его улучшением является вычисление оптимальной точки переключения, что позволяет уменьшить количество итераций. Обширное моделирование показывает, что в среднем улучшенные алгоритмы КМ могут сэкономить около двух итераций, что соответствует сокращению времени вычислений более чем на 39 % [4].

1. Оптимальная точка переключения

Когда мы использовали (3), чтобы инициализировать алгоритм КМ, c_l в первой итерации может быть представлено в виде

$$c = \frac{\sum_{i=1}^N \underline{x}_i \frac{\overline{w}_i + \underline{w}_i}{2}}{\sum_{i=1}^N \frac{\overline{w}_i + \underline{w}_i}{2}},$$

которая выглядит немного отлично от выражения (1) и предполагает лучший выбор для инициализации алгоритма КМ из возможных.

Заметим, что выражение (1) показывает, что для вычисления c_l , $\overline{w}_i$ используется когда $i \leq L$, а $\underline{w}_i$ когда $i \geq L$. Это подразумевает, что лучшей инициализацией c_l является нахождение лучшего предположения для $L - L_0$. Так как y_l наименьшее значение Y , можно предположить, что вероятнее всего оно меньше, чем $\underline{x}_{[N/2]}$, центрального элемента $\underline{x}_l$. Как следствие, L_0 должно быть меньше, чем $[N/2]$. После проведения

вычислений с инициализацией $L_0 = \left\{ \left\lceil \frac{N}{2} \right\rceil, \left\lceil \frac{N}{2,1} \right\rceil, \dots, \left\lceil \frac{N}{2,6} \right\rceil \right\}$ оказалось что наименьшее количество итераций для алгоритма будет при значении $L_0 = [N/2, 4]$. [4] Таким же образом оптимальной начальной точкой переключения для c_r является $R_0 = [N/1, 7]$ [4].

II. Алгоритм УКМ для вычисления y_l . [4]

1) Отсортируем $\underline{x}_i$ ($i = 1, 2, \dots, N$) в порядке возрастания. Сопоставим веса $\overline{w}_i$ в соответствии с $\underline{x}_i$ и пронумеруем их так, чтобы их индексы совпадали с индексами $\underline{x}_i$.

2) Положим $k = [N/2, 4]$ (ближайшее целое к $N/2, 4$), вычислим

$$a = \sum_{i=1}^k \underline{x}_i \overline{w}_i + \sum_{i=k+1}^N \underline{x}_i \underline{w}_i,$$

$$b = \sum_{i=1}^k \overline{w}_i + \sum_{i=k+1}^N \underline{w}_i,$$

$$c = a / b.$$

3) Найдем точку $k' \in [1, N-1]$ такую, что $\underline{x}_{k'} \leq c \leq \underline{x}_{k'+1}$.

4) Проверяем $k' = k$. Если верно, то останавливаемся, присваиваем $c_l = c$ и $k = L$. Если нет, то продолжаем.

5) Вычисляем $s = \text{sign}(k' - k)$ и

$$a' = a + s \sum_{i=\min(k,k')+1}^{\max(k,k')} \underline{x}_i (\overline{w}_i - \underline{w}_i),$$

$$b' = b + s \sum_{i=\min(k,k')+1}^{\max(k,k')} (\overline{w}_i - \underline{w}_i),$$

$$c' = a' / b'.$$

6) Установим $c = c'$, $a = a'$ и $k = k'$. Возвращаемся к шагу 3.

III. Алгоритм УКМ для вычисления c_r . [4]

1) Отсортируем $\overline{x}_i$ ($i = 1, 2, \dots, N$) в порядке возрастания. Сопоставим веса w_i в соответствии с $\overline{x}_i$ и пронумеруем их так, чтобы их индексы совпадали с индексами $\overline{x}_i$.

2) Положим $k = \lceil N / 1,7 \rceil$ (ближайшее целое к $N / 1,7$), вычислим

$$a = \sum_{i=1}^k \overline{x}_i \underline{w}_i + \sum_{i=k+1}^N \overline{x}_i \overline{w}_i,$$

$$b = \sum_{i=1}^k \underline{w}_i + \sum_{i=k+1}^N \overline{w}_i,$$

$$c = a / b.$$

3) Найдем точку $k' \in [1, N-1]$ такую, что $\overline{x}_{k'} \leq c \leq \overline{x}_{k'+1}$.

4) Проверяем $k' = k$. Если верно, то останавливаемся, присваиваем $c_r = c$ и $k = R$. Если нет, то продолжаем.

5) Вычисляем $s = \text{sign}(k' - k)$ и

$$a' = a + s \sum_{i=\min(k,k')+1}^{\max(k,k')} \overline{x}_i (\overline{w}_i - \underline{w}_i),$$

$$b' = b + s \sum_{i=\min(k,k')+1}^{\max(k,k')} (\overline{w}_i - \underline{w}_i),$$

$$c' = a' / b'.$$

6) Установим $c = c'$, $a = a'$ и $k = k'$. Возвращаемся к шагу 3.

Алгоритм сокращения типа замкнутой формы

Этот алгоритм применяется для кусочно-линейных интервальных нечетких множеств второго типа. В этом случае мы можем представить верхнюю и нижнюю функции принадлежности, как множество вершин, которые определяют геометрически фигуру, образуемыми этими функциями. Основная идея, лежащая в основе этого алгоритма, основана на предположении, что отпечаток принадлежности имеет постоянную ширину, в которой ищется центроид. Таким образом, центроид может быть вычислен с использованием формул замкнутой формы без необходимости итераций на основе геометрического подхода.

I. Алгоритм для вычисления c_i . [5]

1) Отсортируем вершины LMF и UMF (x_i, y_i) ($i = 1, \dots, n_1$) и $(\bar{x}_i, \bar{y}_i)$ ($i = 1, \dots, n_2$) в направлении против часовой стрелки, где (x_1, y_1) – первая вершина и $x_1 = \min(x_i)$. n_1 и n_2 – количество вершин LMF и UMF соответственно.

2) Найдем $k_1 \in [1, \dots, n_1]$, такое, что

$$\underline{x}_{k_1} = \min_{\underline{x}_i} \left\| \underline{x}_i - \left(\bar{x}_1 + \frac{\bar{x}_2 - \bar{x}_1}{2.4} \right) \right\|$$

и найдем $k_2 \in [1, \dots, n_2]$, такое, что $\bar{x}_{k_2} \leq \underline{x}_{k_1} \leq \bar{x}_{k_2+1}$.

3) Положим

$$\begin{aligned} x_i &= [\bar{x}_1, \underline{x}_2, \dots, \underline{x}_{k_1}, \bar{x}_{k_2}, \dots, \bar{x}_{n_2}, \bar{x}_1], \\ y_i &= [\bar{y}_1, \underline{y}_2, \dots, \underline{y}_{k_1}, \bar{\mu}_{\bar{y}}(\underline{x}_{k_1}), \bar{y}_{k_2}, \dots, \bar{y}_{n_2}, \bar{y}_1], \\ (i &= 0, \dots, n) \end{aligned}$$

и вычислим

$$A_{pL} = \frac{1}{2} \sum_{i=0}^{n-1} (x_i y_{i+1} - x_{i+1} y_i),$$

$$A_{pL} c_{pL} = \frac{1}{6} \sum_{i=0}^{n-1} (x_i + x_{i+1}) \cdot (x_i y_{i+1} - x_{i+1} y_i).$$

4) Вычислим кандидата на минимальный центроид

$$L = x_L + \frac{-A_{pL} + \sqrt{A_{pL}^2 + 2\Delta\mu_{\bar{y}}(x_L)A_{pL}(c_{pL} - x_L)}}{\Delta\mu_{\bar{y}}(x_L)},$$

где $\Delta\mu_{\bar{A}}(x) = \bar{\mu}_{\bar{A}}(x) - \underline{\mu}_{\bar{A}}(x)$.

Если $\Delta\mu_{\bar{A}}(x_L) = \Delta\mu_{\bar{A}}(L)$, тогда останавливаемся, $c_l = L$; в противном случае переходим к шагу 5.

5) Вычисляем улучшенный результат

$$c_l = L_e = \frac{(A_{pL} + A_{FOU_L})L - A_L c_L}{A_{pL} + A_{FOU_L} - A_L},$$

где

$$\begin{aligned} A_{FOU_L} &= (L - x_L) \Delta\mu_{\bar{A}}(x_L), \\ A_L &= \frac{L - x_L}{2} (\Delta\mu_{\bar{A}}(x_L) - \Delta\mu_{\bar{A}}(L)), \\ c_L &= \frac{2L + x_L}{3}. \end{aligned}$$

II. Алгоритм для вычисления c_r . [5]

1) Отсортируем вершины LMF и UMF $(\underline{x}_i, \underline{y}_i)$ ($i = 1, \dots, n_1$) и $(\bar{x}_i, \bar{y}_i)$ ($i = 1, \dots, n_2$) в направлении против часовой стрелки, где (x_1, y_1) – первая вершина и $x_1 = \min(x_i)$. n_1 и n_2 – количество вершин LMF и UMF соответственно.

2) Найдем $k_1 \in [1, \dots, n_1]$ такое, что

$$\underline{x}_{k_1} = \min_{\underline{x}_i} \left\| \underline{x}_i - \left(\bar{x}_1 + \frac{\bar{x}_2 - \bar{x}_1}{2.4} \right) \right\|$$

и найдем $k_2 \in [1, \dots, n_2]$, такое, что $\bar{x}_{k_2} \geq \underline{x}_{k_1} \geq \bar{x}_{k_2+1}$.

3) Положим

$$\begin{aligned} x_i &= [\underline{x}_1, \bar{x}_2, \dots, \bar{x}_{k_2}, \underline{x}_{k_1}, \bar{x}_{k_1}, \underline{x}_{k_1+1}, \dots, \bar{x}_{n_1}, \underline{x}_1], \\ y_i &= [\underline{y}_1, \bar{y}_2, \dots, \bar{y}_{k_2}, \underline{\mu}_{\bar{A}}(\underline{x}_{k_1}), \bar{y}_{k_1}, \dots, \bar{y}_{n_1}, \underline{y}_1], \\ (i &= 0, \dots, n) \end{aligned}$$

и ВЫЧИСЛИМ

$$\begin{aligned} A_{pR} &= \frac{1}{2} \sum_{i=0}^{n-1} (x_i y_{i+1} - x_{i+1} y_i), \\ A_{pR} c_{pR} &= \frac{1}{6} \sum_{i=0}^{n-1} (x_i + x_{i+1}) \cdot (x_i y_{i+1} - x_{i+1} y_i). \end{aligned}$$

4) Вычислим кандидата на максимальный центроид

$$R = x_R + \frac{-A_{pR} + \sqrt{A_{pR}^2 + 2\Delta\mu_{\tilde{A}}(x_R)A_{pR}(c_{pR} - x_R)}}{\Delta\mu_{\tilde{A}}(x_R)},$$

где $\Delta\mu_{\tilde{A}}(x) = \overline{\mu_{\tilde{A}}}(x) - \underline{\mu_{\tilde{A}}}(x)$.

Если $\Delta\mu_{\tilde{A}}(x_R) = \Delta\mu_{\tilde{A}}(R)$, тогда останавливаемся, $c_r = R$; в противном случае переходим к шагу 5.

6) Вычисляем улучшенный результат

$$c_r = R_e = \frac{(A_{pR} + A_{FOUR})R - A_R c_R}{A_{pR} + A_{FOUR} - A_R},$$

где

$$A_{FOUR} = (R - x_R)\Delta\mu_{\tilde{A}}(x_R),$$

$$A_R = \frac{R - x_R}{2}(\Delta\mu_{\tilde{A}}(x_R) - \Delta\mu_{\tilde{A}}(R)),$$

$$c_R = \frac{2R + x_R}{3}.$$

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Mendel J. M. Rule-Based Fuzzy Logic Systems: Introduction and New Directions. – Upper Saddle River, NJ: Prentice-Hall, 2001. – 547 с.
2. Karnik N.N. Mendel J. M. Centroid of a type-2 fuzzy set. – Information Sciences, т. 132, 2001. – 195-220 с.
3. Mendel J. M., Liu F. Super-exponential convergence of the Karnik-Mendel algorithms for computing the centroid of an interval type-2 fuzzy set. – IEEE Trans. on Fuzzy Systems, т. 15, н. 2, 2007. 309–320 с.
4. Wu D., Mendel J. M. Enhanced Karnik-Mendel Algorithms for Interval Type-2 Fuzzy Sets and Systems. – 2007 Annual Meeting of the North American Fuzzy Information Processing Society, 2007. – 184-189 с.
5. Ulu C., Guzelkaya M., Eksin I. A closed form type reduction method for piecewise linear interval type-2 fuzzy sets. – International Journal of Approximate Reasoning т. 54, н. 9, 2013. – 1421-1433 с.

УДК 004.9

Скачков А.М.

АНАЛИЗ ПРОБЛЕМЫ ПОДБОРА ЗНАЧЕНИЙ ГИПЕРПАРАМЕТРОВ В НЕЙРОННЫХ СЕТЯХ НА ПРИМЕРЕ БИБЛИОТЕКИ KERAS

Рязанский государственный радиотехнический университет,

г. Рязань

В статье выполнен анализ проблемы настройки гиперпараметров нейросети.

Нейронные сети – одно из направлений в методах машинного обучения. Идея заключается в том, чтобы максимально близко смоделировать работу человеческой нервной системы, а именно, её способности к обучению и исправлению ошибок. В этом состоит главная особенность любой нейронной сети. Она способна самостоятельно обучаться и действовать на основании предыдущего опыта, с каждым разом делая всё меньше ошибок.

Но не всегда созданная нейронная сеть может с большой точностью решать ту или иную задачу даже после долгого времени обучения. Так происходит из-за некачественных данных или неправильного подбора гиперпараметров сети. Таким образом, проблема подбора значений гиперпараметров при обучении нейронных сетей актуальна. Рассмотрим данную проблему на прикладной задаче.

Пусть необходимо обучить нейронную сеть для классификации чисел от 0 до 9. Для удобства в библиотеке Keras есть возможность быстро скачать данные для обучения [2]. Они представлены в виде различных наборов, для поставленной задачи будет использоваться набор данных MNIST [1]. Он представляет собой базу данных рукописных цифр, каждое изображение имеет размер 28 на 28 пикселей (рис. 1).



Рис. 1. Набор данных MNIST

В используемой библиотеке Keras подбор этих параметров осуществляется в методах *fit* и *add*.

Рассмотрим гиперпараметры задаваемые в методе *fit*:

- `batch_size` – размер минивыборки – это количество изображений, подаваемых нейронной сети за один раз;
- `epochs` – количество эпох обучения (сколько раз нейронная сеть обучается).

Рассмотрим метод `add`, в нём только один гиперпараметр – это количество нейронов.

Перейдем к практике. При обучении нейронной сети будем использовать следующие параметры: количество нейронов – 800, размер минивыборки – 200, количество эпох – 10. В результате обучения получили нейронную сеть, решающую задачу распознавания рукописных цифр, которая имеет точность равную 97,89 % (рис. 2).

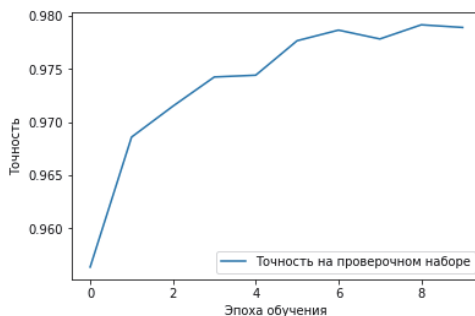


Рис. 2. Визуализация качества обучения нейронной сети

Теперь обучим нейронную сеть с другими гиперпараметрами: уменьшим размер минивыборки до 50, и увеличим количество нейронов до 1000, а также количество эпох до 25. Получили точность равную 97,69 %. Снижение качества распознавания произошло из-за того, что 50 изображений, которые входят в минивыборку, слишком мало и мы не можем выделить однозначные признаки и найти наиболее похожие для их дальнейшего преобразования (рис. 3).

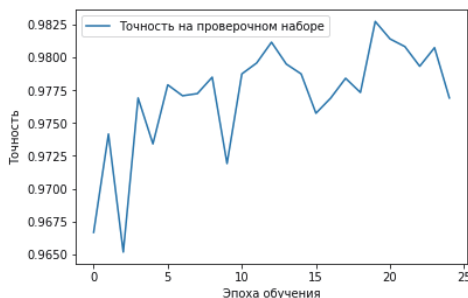


Рис. 3. Визуализация качества обучения нейронной сети с меньшим размером минивыборки

В процессе многочисленных изменений гиперпараметров были установлены наиболее оптимальные количество нейронов – 1000, размер минивыборки – 200, количество эпох – 25. которые дали точность распознавания 98,31 % (рис. 4).



Рис. 4. Визуализация качества обучения нейронной сети с оптимальными гиперпараметрами

В заключении стоит отметить, что подбор гиперпараметров при обучении нейронной сети является очень важным и сложным процессом, так как не существует какого-либо универсального метода подбора параметров, который бы подошел под любую задачу.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. https://en.wikipedia.org/wiki/MNIST_database
2. <https://keras.io/>

УДК 004.852

Балабанов Н.Р.

ПРОБЛЕМЫ ЗАЩИТЫ ИНФОРМАЦИОННЫХ СИСТЕМ ОТ ЧЕЛОВЕЧЕСКОГО ФАКТОРА

Рязанский государственный радиотехнический университет,

г. Рязань

Одной из самых трудных задач, с которыми сталкиваются специалисты по защите информационных систем, является задача защиты от ошибок людей, которые пользуются этими системами.

Как показывает практика, люди – самая слабая часть защиты в информационных системах. Защита, например, базы данных может быть сколь угодно хороша, в ней могут быть применены сколь угодно высокие технологии, но все они будут бесполезны, если злоумышленник будет иметь пароль от неё или критическую ошибку допустит оператор с высокими правами доступа.

Сегодня эта проблема носит название: «Проблема привилегированных пользователей» [1, 2].

К примеру, из всех проблем человеческого фактора, одной из самых серьёзных и трудно решаемых является проблема контроля системных администраторов.

Так как системные администраторы имеют полный доступ к системе (или возможность обойти имеющиеся ограничения), то они обладают полной свободой действий в системе. И ничто не останавливает их от использования доступа в личных целях.

Согласно опросу западных аналитических агентств, в 2012 году среди почти тысячи системных администраторов, 42 % из них когда-либо использовали свое служебное положение для несанкционированного доступа к конфиденциальным и важным данным своей организации. При этом 74 % системных администраторов признались, что они с легкостью обходят любые меры безопасности, ограничивающие их доступ.

Более того, значительная часть системных администраторов призналась, что, в случае неожиданного увольнения или сокращения, охотно прихватят с собой эти самые конфиденциальные данные или подложат «бомбу» в виде программы с таймером, которая нанесёт вред информационной системе через некоторое время после увольнения.

Однако не только сотрудники компании могут представлять угрозу. Малые компании, которым невыгодно содержать «железо» информационной системы самим, часто отдают все свои системы на поддержку третьим лицам – это называется аутсорсинг. При этом пропадает возможность хоть какого-то контроля за тем, кто и зачем получает доступ к конфиденциальной информации.

Как бороться с подобными угрозами?

Как показывает практика, единственный способ – это ужесточение контроля и наказания.

Централизованное управление и слежение за административными аккаунтами позволяет отследить каждое действие, что не позволит никому безнаказанно получать доступ к важной информации.

Однако подобная борьба имеет и обратную сторону – бюрократию. К примеру, компания «Сбербанк» имеет очень мощную систему регламентирования доступа, а также вообще всего, что касается их отдела ИТ. Это вылилось в огромный объём бюрократии, когда простой запрос в базу требует разрешения, получение которого занимает недели, а то и месяцы.

Ещё один способ защиты – сделать информацию в системе бесполезной для кого или чего-угодно, кроме данной системы. То есть зашифровать все данные, исключив человека из работы с ними вообще.

В целом, все методы решения подобных проблем – это пример ответить на поговорку: «Кто сторожит сторожей?»

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Проблема «Привилегированных пользователей» (<https://www.anti-malware.ru/practice/methods/PUM-Wallix-Admin-Bastion>)

2. Проблемы контроля привилегированных пользователей и их решение на примере Wallix AdminBastion Источник: <https://www.anti-malware.ru/practice/methods/PUM-Wallix-Admin-Bastion>

УДК 519.7

Чернобродова Л.А., Кабанов А.Н., Моргунов Д.Р.

РАЗРАБОТКА ИНФОРМАЦИОННОГО ОБЕСПЕЧЕНИЯ МНОГОПАРАМЕТРИЧЕСКОЙ ОПТИМИЗАЦИИ УПРАВЛЕНИЯ ИНВЕСТИЦИОННОЙ ДЕЯТЕЛЬНОСТИ ОРГАНИЗАЦИИ НА ОСНОВЕ ЭКСПЕРТНОГО АНАЛИЗА

Рязанский государственный радиотехнический университет,

г. Рязань

Решается задача определения важности критериев оптимизации.

Введение. В условиях глобальной турбулентности мировой экономики, введения режима санкций в отношении России со стороны западных стран, внешней зависимости от займов, вывоза капитала возникает необходимость поиска новых источников внутреннего экономического роста промышленных предприятий.

Эффективная деятельность предприятий в долгосрочной перспективе, обеспечение высоких темпов их развития и повышение конкурентоспособности в значительной мере определяются уровнем их инвестиционной активности.

Ключевыми характеристиками экономического содержания инвестиций являются следующие:

1. Инвестиции выступают объектом экономического управления.
2. Сущность инвестиций связана с такой категорией, как капитал, который определяет любые ресурсы, созданные и используемые для производства большего количества благ.
3. Инвестиции всегда осуществляются с определенной целью. Целевой установкой инвестиций является достижение конкретного заранее предопределенного эффекта, который может носить как экономический (прирост суммы инвестированного капитала, положительная величина инвестиционной прибыли, положительная величина чистого денежного притока, обеспечение сохранения ранее вложенного капитала и т.п.), так и внеэкономический характер (например, социальный, экологический и другой).
4. Инвестиции непосредственно связаны с фактором времени. При осуществлении инвестиций имеет место отказ от потребления сегодня в пользу будущих благ. При этом экономическая ценность сегодняшних и будущих благ для собственника капитала неравнозначна: при прочих равных условиях возможности будущего потребления менее ценны в сравнении с текущим потреблением.

5. Инвестиции выступают носителем фактора риска, так как инвестор, осуществляя инвестиции, всегда осознанно идет на риск, связанный с возможным снижением или неполучением суммы ожидаемого инвестиционного дохода, а также возможной частичной или полной потерей инвестированного капитала.

Инвестиционная деятельность предприятий во многом зависит от стоимости инвестиционных ресурсов на рынке, заинтересованности сторонних инвесторов во вложенных средствах, ограниченности административных барьеров при получении государственных субсидий и льготных кредитов. Перечисленные факторы вынуждают предпринимательские структуры промышленного сектора активизировать поиск новых источников финансирования, более эффективного механизма их формирования и использования.

Для ранжирования критериев (показателей инвестиционной деятельности) должна быть создана экспертная комиссия и проведен экспертный анализ. Анализ проводится методом парных сравнений (Саати).

Многообразие критериев и порядок их важности.

1. Определение согласованного вектора приоритетов различных критериев методом попарного сравнения [1].

С помощью эксперта формируется матрица

$$\bar{A} = \{a_{ij}\},$$

где a_{ij} означает, во сколько раз i -й критерий важнее j -го;

$$a_{ij} = 1 / a_{ji}.$$

Пример 1.

$$\text{Заполняем матрицу } \bar{A} = \begin{bmatrix} 1 & 4 & 6 & 7 \\ 0,25 & 1 & 3 & 4 \\ 0,167 & 0,333 & 1 & 2 \\ 0,143 & 0,25 & 0,5 & 1 \end{bmatrix}.$$

Находим самое большое собственное число и соответствующий ему собственный вектор. Для этого используем сингулярное разложение. Изложим последовательность операций.

1) $\bar{A} \cdot \bar{A}^T \rightarrow \bar{U} \bar{D}_1 \bar{U}^T$ – находим

$$\bar{D}_1 = \begin{bmatrix} 132,93398 & & & \\ & 1,47127 & & \\ & & 0,12827 & \\ & & & 0,007 \end{bmatrix},$$

$$\bar{U} = \begin{bmatrix} 0,87452 & -0,47533 & \dots & \dots \\ 0,43645 & 0,68939 & \dots & \dots \\ 0,18795 & 0,51186 & \dots & \dots \\ 0,09696 & 0,19186 & \dots & \dots \end{bmatrix};$$

$$2) \bar{A}^T \cdot \bar{A} = \bar{U} \bar{D}_2 \bar{V}^T;$$

$$\bar{D}_2 = \begin{bmatrix} 132,9 & & & \\ & 1,4712 & & \\ & & 0,12 & \\ & & & 0,007 \end{bmatrix}, \bar{V} = \begin{bmatrix} 0,08 & \dots & \dots & \dots \\ 0,34 & \dots & \dots & \dots \\ 0,58 & \dots & \dots & \dots \\ 0,72 & \dots & \dots & \dots \end{bmatrix};$$

$$3) \bar{D} = \bar{U}^T \cdot \bar{A} \cdot \bar{V};$$

$$\bar{D} = \begin{bmatrix} 11,5207 & & & \\ & -1,21296 & & \\ & & -0,35815 & \\ & & & 0,02665 \end{bmatrix}.$$

Самое большое собственное число 11,52. Все остальные диагональные элементы обнуляем:

$$\hat{\bar{D}} = \begin{bmatrix} 11,5207 & & & \\ & 0 & & \\ & & 0 & \\ & & & 0 \end{bmatrix}, \hat{\bar{A}} = \bar{U} \hat{\bar{D}} \bar{V}^T = \begin{bmatrix} 0,89978 & 0 & 0 & 0 \\ 0,44905 & 0 & 0 & 0 \\ 0,19338 & 0 & 0 & 0 \\ 0,09076 & 0 & 0 & 0 \end{bmatrix}.$$

Нормализуем первый столбец матрицы $\hat{\bar{A}} \cdot \|1 \text{ cm}\| = 1,029$.

$$\text{Вес } \bar{w} = \begin{bmatrix} 0,87442 \\ 0,43639 \\ 0,18793 \\ 0,09695 \end{bmatrix} - \text{«согласованный» вектор весов.}$$

Вывод: согласованный вектор весов получается устойчивым. Он сформирован на основе 16 «измерений» см. матрицу $\hat{\bar{A}}$). На основе оценок многих экспертов определяют кратчайшее остовное дерево и производят разбиение на однородные группы экспертов.

Построение кратчайшего остовного дерева с помощью алгоритма Прима в табличной форме

Исходные данные представляем в виде таблицы. Число экспертов для примера $N = 10$.

	X1	X2	X3	X4	X5	X6
№1	1,12	0,92	0,68	0,8	0,88	4
№2	0,52	0,74	0,64	0,59	0,68	2,08
№3	0,41	0,64	0,64	0,66	0,82	3,27
№4	0,36	0,68	0,44	2	0,76	3,66
№5	0,51	0,54	0,54	6,15	0,90	3,93
№6	0,74	1,53	0,82	1,51	0,83	4,42
№7	1,26	0,43	1,16	1,76	1,44	5,42
№8	0,42	1,01	0,95	0,76	0,82	2,98
№9	0,37	0,75	0,58	0,56	0,78	3,5
№10	1,17	0,74	0,85	0,93	0,61	4,23

Здесь $x_i (i = 1 \div 6)$ – это веса, задаваемые экспертом. В таблице даны результаты анализа десяти экспертов. На основе этих данных формируется матрица расстояний между экспертами по указанным ими шести параметрам x_i .

В двумерном пространстве расстояние между точками с координатами (x_1, y_1) и (x_2, y_2) рассчитывается по формуле $p = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$. Для n -мерного случая, когда в координатной системе $(x_1, x_2, \dots, x_n)$ имеются две точки с координатами $(x_{11}, x_{12}, \dots, x_{1n})$ и $(x_{21}, x_{22}, \dots, x_{2n})$, справедлива формула $p = \sqrt{(x_{11} - x_{21})^2 + (x_{12} - x_{22})^2 + \dots + (x_{1n} - x_{2n})^2}$.

Этапы алгоритма.

1. По диагонали в матрице расстояний на основании алгоритма ставят большие числа.
2. Формируется массив меток столбцов матрицы весов. Вначале все метки обнуляются.
3. Произвольно выбранному столбцу присваивается значение метки, равной его номеру.
4. Этап, повторяющийся $N-1$ раз (общий этап). В строках, номера которых равны номерам помеченных столбцов, находится минимальный элемент среди элементов непомеченных столбцов. Столбец, в котором находится минимальный элемент, помечается меткой, номер которой равен номеру его строки. В случае если минимальных элементов несколько, то выбирается любой. После того как будет помечен очередной столбец, элементу, симметричному относительно главной диагонали (для многомерного графа – с «транспонированными индексами»), присваивается сколь угодно большое значение.
5. Заключительный этап. Ребра, включенные в минимальное остовное дерево, определяются по меткам столбцов. Вес остовного дерева задается суммой весов входящих в него ребер.

Для разбиения дерева на k однородных групп необходимо удалить $(k-1)$ ребро так, чтобы суммарная дисперсия всех групп была минимальной.

Вывод: Представлен алгоритм формирования однородных групп при многокритериальной оптимизации управления инвестиционной деятельностью организации.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Саати Т. Принятие решений. Метод анализа иерархий: М: «Радио и связь», 1993. – 320 с.

УДК 519.7

Чернобродова Л.А., Кабанов А. Н., Иешкин А. Ю.

ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ МНОГОКРИТЕРИАЛЬНОЙ АДАПТИВНОЙ ОПТИМИЗАЦИИ В ИНВЕСТИЦИОННОЙ ПОЛИТИКЕ ПРЕДПРИЯТИЙ

Рязанский государственный радиотехнический университет,

г. Рязань

Предложен метод оперативного решения задачи оптимизации для совершенствования инвестиционной политики предприятия

Введение. Инвестиционная политика является важнейшим условием реализации стратегических и тактических задач развития и эффективной деятельности предприятия.

Роль инвестиций в обеспечении эффективности предприятия:

- главный источник формирования производственного потенциала предприятия;
- основной механизм реализации стратегических целей экономического развития;
- главный механизм оптимизации структуры активов;
- основной фактор формирования долгосрочной структуры капитала;
- важнейшее условие обеспечения роста рыночной стоимости предприятия;
- основной механизм обеспечения расширенного воспроизводства основных средств и нематериальных активов;
- главный инструмент реализации инновационной политики;
- один из механизмов решения задач социального развития персонала.

Инвестиционная политика реализуется в ходе инвестиционной деятельности. Инвестиционная деятельность предприятия – целенаправленно осуществляемый процесс изыскания необходимых инвестиционных ресурсов, выбора эффективных объектов инвестирования, формирование сбалансированной инвестиционной программы (инвестиционного портфеля) и обеспечения ее реализации. Инвестиционную деятельность можно

определить как единство процессов вложения ресурсов и получения доходов в будущем.

Инвестиционная деятельность предприятия предполагает наличие риска. Задача управления инвестициями состоит в том, чтобы минимизировать риск и максимизировать эффект.

Так как одна из основных трудностей инвестиционного проектирования – многомерность и многокритериальность, появляется задача – адаптивной оптимизации.

При анализе инвестиционной деятельности предприятия или отдельного взятого инвестиционного проекта, который затрагивает разнообразные интересы участников, возникает необходимость оценки вариантов решений по нескольким критериям.

Оценка эффективности инвестиционной деятельности осуществляется по таким критериям как: чистая приведенная стоимость (NPV), индекс рентабельности инвестиций (PI), внутренняя норма доходности (IRR), срок окупаемости инвестиции (PP) и др.

Так как инвестиционная деятельность предполагает наличие риска, связанного с изменением условий и параметров, то в ходе оценки эффективности проектов может меняться значимость (ранг) критерия, и возникает необходимость адаптации параметров. Адаптация предполагает решение задачи выявления наиболее неопределенных параметров и применения технологии оптимизации.

В качестве неопределенных параметров проекта часто выступают: прибыль, объем производимых работ (услуг); цена и себестоимость единицы продукции, производительность труда и др., которые оказывают влияние на показатели эффективности проекта.

Множественность показателей эффективности (ряда количественных показателей $F_1, F_2, \dots$, из которых одни желательно обратить в максимум, а другие в минимум), называемая *многокритериальностью*, характерна для многих крупномасштабных задач исследования операций.

1. Описание метода многокритериальной оптимизации

Для каждого критерия должна быть задана область допустимых параметров. Эта область может быть представлена системой линейных неравенств.

Для заданного набора критериев для повышения оперативности и точности решения многокритериальных задач нужно исключать лишние неравенства из области ограничений.

Иными словами, план должен быть хотя и не оптимальным по каждому отдельному критерию, но наилучшим по выполнению всех критериев одновременно. Такие планы называют *наиболее эффективными, компромиссными* или *субоптимальными*.

2. Алгоритм предварительного решения системы линейных неравенств [1].

Вопрос о строении множества решений произвольной совместной системы неоднородных линейных неравенств

$$a_{j1}X_1 + a_{j2}X_2 + \dots + a_{jn}X_n - a_j \leq 0, \quad j = \overline{1, m}, \quad (1)$$

может быть сведен к поиску общей формулы решений системы однородных линейных неравенств

$$\begin{aligned} a_{j1}X_1 + a_{j2}X_2 + \dots + a_{jn}X_n - a_jX_{n+1} &\leq 0, \quad j = \overline{1, m}, \\ -X_{n+1} &\leq 0. \end{aligned} \quad (2)$$

Действительно, если известна общая формула решений системы, то при $X_{n+1} = 1$ из нее получаем решение для исходной системы (1). Поэтому основное внимание при анализе решений в дальнейшем уделено системе (2).

Определение фундаментальной системы решений для системы однородных линейных неравенств.

Перепишем однородную систему неравенств в виде

$$a_{j1}X_1 + \dots + a_{jn}X_n \leq 0, \quad j = \overline{1, m}. \quad (3)$$

Алгоритм и вычислительная схема для нахождения общей формулы неотрицательных решений системы (3) сводятся к последовательным однотипным преобразованиям таблицы

$$T^1 = [T_1^1 | T_2^1] = \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & a_{11} & \dots & a_{m1} \\ 0 & 1 & 0 & & & \\ \hline 0 & 0 & 1 & a_{1n} & \dots & a_{mn} \end{array} \right], \quad (4)$$

в которой левая часть T_1^1 (до вертикальной черты) – единичная матрица $E(1,1)$ n -й степени, а правая часть T_2^1 – транспонированная матрица исходной матрицы системы (3).

Для удобства изложения будем считать, что все таблицы T^i ($i = 2, 3, \dots$), получаемые в результате последовательных преобразований таблицы T^1 , помещаются под таблицей T^1 (как её последовательные продолжения вниз) и имеют общую с ней разделяющую вертикальную черту; при этом правая и левая части таблицы T^i ($i = 2, 3, \dots$) будут обозначаться соответственно через T_1^i и T_2^i и столбцы всех таблиц T_2^i нумероваться слева направо числами $1, 2, \dots, m$. Переходя к изложению вычислительной схемы, рассмотрим подробно преобразование таблицы T^1 в таблицу T^2 . Если все элементы какого-нибудь столбца таблицы T_2^1 неположительные (неположительный столбец), то заменяем их нулями; отвечающее этому столбцу неравенство системы (3) является зависимым неравенством системы

$$\left. \begin{aligned} a_{j1}X_1 + \dots + a_{jn}X_n &\leq 0, & j = \overline{1, m}; \\ -X_i &\leq 0, & i = \overline{1, n}. \end{aligned} \right\} \quad (5)$$

Затем выбирается основной столбец таблицы T^1 , за который можно принять любой из столбцов таблицы T_2^1 , имеющий хотя бы один положительный элемент. Если таких столбцов не окажется, то таблица T^2 не существует. Пусть t^1 – основной столбец таблицы T^1 . Тогда в таблицу T^2 прежде всего переносятся строки таблицы T^1 , пересекающиеся с основным столбцом по неположительным (нулевым и отрицательным) элементам.

Другие строки таблицы T^2 получаются комбинированием некоторых пар строк таблицы T^1 . Для того чтобы выделить такие пары, условимся называть допустимой пару строк таблицы T^1 , если содержащиеся в них элементы основного столбца t^1 являются ненулевыми элементами противоположных знаков. Если таблица T^1 содержит более двух строк и существуют столбцы таблицы T_1^1 , пересекающие по нулевым элементам обе строки допустимой пары, но не существует никакой другой строки таблицы T^1 , пересекающей по нулевым элементам со всеми столбцами такого рода, то при этом условии допустимую пару назовем уравновешенной. Если таблица T^1 содержит две строки и они составляют допустимую пару, то эта пара считается уравновешенной. Строкой равновесия уравновешенной допустимой пары строк таблицы T^1 назовем такую их линейную комбинацию с положительными коэффициентами, которая пересекается по нулевому элементу со столбцом t^1 (имеет нуль на месте, определяемом столбцом t^1). Строки равновесия всех уравновешенных допустимых пар таблицы T^1 вносятся в таблицу T^2 . После этого все ненулевые элементы столбца таблицы T^2 , возникшего из основного столбца t^1 , заменяются элементом -1 , а все элементы других неположительных её столбцов – нулями. На этом составление таблицы T^2 заканчивается.

Выводы. Таким образом, удаляя лишние неравенства, мы существенно сокращаем время обработки информации и повышаем точность.

Метод имеет практическую значимость и может быть применен для совершенствования методов поддержки принятия инвестиционных решений.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Черников С.Н. Линейные неравенства. – Москва, 1968. – 488 стр.

УДК 004.9

Романов А.И. Медведев Р.Е.

ИСПОЛЬЗОВАНИЕ ДИСТАНЦИОННЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ ПРИ ОРГАНИЗАЦИИ ПОЫШЕНИЯ КВАЛИФИКАЦИИ СОТРУДНИКОВ ПРОИЗВОДСТВЕННЫХ ПРЕДПРИЯТИЙ

Рязанский государственный радиотехнический университет,

г. Рязань

В статье рассмотрены преимущества и основные способы организации повышения квалификации сотрудников предприятия с использованием дистанционных образовательных технологий

Специфика современной экономики, заключающаяся во всеобъемлющем внедрении современных информационно-коммуникационных технологий во все сферы производственной деятельности предприятия, привела к повышению интеллектуализации производственных процессов, порождающему спрос на высококвалифицированные кадры. Наибольшую ценность при этом стал приобретать сотрудник, способный при необходимости получить дополнительные знания без отрыва от производства, с минимальными временными и материальными затратами, тем самым повысить свои профессиональные навыки. С точки зрения эффективности деятельности предприятия и увеличения производительности труда, повышение квалификации сотрудников должно проходить с минимальным отрывом от производства, что заставляет работодателя искать новые пути получения образования и обратить внимание на активно развивающуюся форму организации образовательного процесса, а именно дистанционное обучение (далее – ДО), основанное на широком применении компьютерных и сетевых технологий.

Под дистанционным обучением [1] понимается комплекс образовательных услуг, предоставляемых широким слоям населения в стране и за рубежом с помощью информационной образовательной среды, базирующейся на средствах обмена учебной информацией на расстоянии. Основные преимущества данной формы заключаются в доступности широким слоям населения, системности, параллельности с выполнением производственных задач, использованием современных информационных технологий, учете индивидуальных особенностей обучаемого, гибкости, независимости от временных и географических факторов. Целесообразно выделить две группы обучающих технологий, используемых в рамках организации ДО: электронные учебные материалы (электронные учебники, базы данных или знаний, электронные библиотеки) и системы ДО (далее – СДО).

Электронные учебные материалы (далее – ЭУМ) представлены статичными гипертекстовыми страницами, содержащими теоретический материал в форме мультимедиа информации (аудио-, видео-фрагментов и

анимации), а также заданий или тестовых вопросов. При этом упор делается на усредненного по своей подготовленности обучающегося и не учитываются его индивидуальные особенности, навыки и запросы. Несмотря на наличие большого и разнообразного количества инструментов создания и поддержки учебно-методических материалов и проверочных заданий учебный материал при такой организации рассматривается как мультимедийно-оформленный текст, а сам подход рассчитан на автоматизацию труда дизайнера, а не преподавателя. Недостатки этой группы обучающих средств очевидны: низкое качество учебно-методической информации, изложение материала без учета индивидуальных особенностей изучаемого, отсутствие формально описных методов проверки знаний изучаемого, примитивная автоматизация труда преподавателя (подбор учебного материала, разработка курсов и заданий, контроль процесса обучения).

Недостатки ЭУМ заставляют обратить внимание на активно развивающиеся СДО – сложные многокомпонентные системы, представляющие собой совокупность организационных, телекоммуникационных, педагогических и научных ресурсов, а также технических средств, вовлеченных в создание и практическое осуществление программ с использованием дистанционных технологий обучения. Большинство систем бесплатны, что минимизирует затраты на этапе внедрения системы. Использование СДО в рамках повышения квалификации сотрудников производственных участков, предоставляет обширный набор возможностей для представления учебного материала, проверки знаний и контроля успеваемости, общения и организации образовательного процесса. При данном подходе к повышению квалификации, сотрудники предприятия будут отвлечены от производственной цепочки на минимальный промежуток времени, что позволит сократить сопутствующие финансовые затраты предприятия.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Медведев Р.Е. Классификация программных обучающих сред // Наука и образование в 21 веке: материалы международной науч.-практ. конф. Тамбов: ТРОО «Бизнес – наука – общество». 2013. С. 85-86.
2. Каширин И.Ю., Медведев Р.Е. Онтологическое накопление учебных знаний на основе сервиса информационных сетей // Научно-технические ведомости СПбГПУ. 2013. №174. С. 75-85.

УДК 330.1

Сидорова М.А.

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ BLOCKCHAIN ПРИ СОВЕРШЕНСТВОВАНИИ ВЗАИМОДЕЙСТВИЯ ПРЕДПРИЯТИЙ АВИАЦИОННОЙ ПРОМЫШЛЕННОСТИ

Рязанский государственный радиотехнический университет,

г. Рязань

В работе рассматриваются перспективы применения технологий распределенного реестра (Blockchain) для совершенствования работы предприятия авиационной промышленности.

Blockchain – распределенная база данных, которая содержит информацию обо всех транзакциях (более обобщенно – коммуникациях), проведенных участниками системы, при этом информация хранится в виде «цепочки блоков», в каждом из которых записано определенное число коммуникаций [2].

Таким образом Blockchain можно определить как способ накопления и последующего хранения в виде цифрового реестра информационных данных о проведенных транзакциях, совершенных сделках, заключенных контрактах. Основным преимуществом данного реестра над другими является децентрализованное хранение содержащейся в нем информации и свободный доступ каждого пользователя сети к его актуальной версии.

Анализ основных аспектов применения Blockchain-технологий целесообразно осуществлять с акцентом на выявление положительных эффектов от их использования в конкретной отрасли экономики. В данной статье рассмотрим эти аспекты с позиции внедрения технологии распределенного леджера в авиационную промышленность России.

Технология дает перспективные шансы для повышения потенциала авиационной промышленности, так как способна обеспечить простое выполнение задач, направленных на поддержание системы функционирования отрасли в заданных условиях. Blockchain может быть ориентирована помочь авиационной промышленности в следующих направлениях:

1. Ведение точного, одиночного, чётко прослеживаемого учёта записей серийных номеров для различных приборов, оборудования, комплектующих частей, что позволит создать универсальный прослеживаемый список.

2. Формирование более совершенной линии поставок путем ведения оперативной и точной истории обслуживания, что в свою очередь ведет к повышению прозрачности и эффективности цепей поставок.

3. Обеспечение доверительных отношений между поставщиками услуг, поставщиками и операторами.

Данные направления могут быть осуществлены в рамках организации SMART-контрактов в системе взаимосвязи предприятий отечественной

авиаремонтной промышленности (рис. 1, где 1 – «вход и регистрация на платформе», 2 – «размещение заказа», 3 – «описание всех технических характеристик самолета с указанием истории эксплуатации и сервиса», 4 – «анализ и распределение заказа», 5 – «включение необходимого звена в виде исполнителя заказа», 6 – «поступление заказа и всех характеристик единицы техники», 7 – «обработка заказа», 8 – «исполнение заказа», 9 – «внесение всех изменений в технические характеристики единицы техника», 10 – «оцифровка данных и передача на предыдущий этап цепочки», 11 – «передача самолета»).

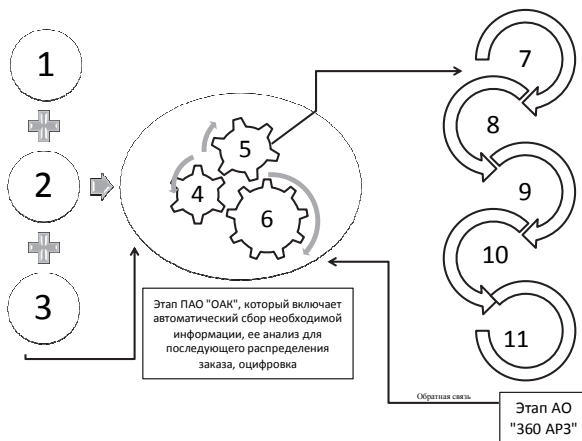


Рис. 1. Схема организации умных контрактов в авиаремонтной отрасли Российской Федерации

Развивая тему использования Blockchain при организации процессов технического обслуживания воздушных судов необходимо сказать о том, что в настоящее время подавляющая часть данных, которые непрерывно собираются в ходе эксплуатации и сервисного обслуживания летных аппаратов, существуют лишь в служебных документах, компонентах и системах воздушных судов, а не формируются в цифровом виде, что значительно замедляет рабочие процессы, а также ограничивает своевременность и полноту необходимой информации. Способность Blockchain оперативно сопоставлять данные, а также определять соответствие информации различных контрагентов является перспективной и многообещающей новинкой для авиационной отрасли. Неизменяемый, защищенные и прозрачный характер технологии распределенного реестра поможет обеспечить удобство организации работ, сокращение ошибок на производстве, а как следствие большую безопасность эксплуатации, позволив произвести оцифровку данных, до сих пор изолированных друг от друга.

Однако, какими бы ни были потенциальные возможности применения данной технологии на практике, необходимо оценить потенциальные технологические проблемы, связанные с ее внедрением. По мнению специалистов, анализирующих специфику технических решений, лежащих в основе Blockchain технологий, на данный момент имеются два нерешенных вопроса: обеспечение пропускной способности сети для нормальной работы Blockchain и предоставление узлу необходимого объема дискового пространства [1].

Таким образом, можно говорить о том, что в такой критичной области как авиация значение источника всех запчастей, наличие данных о качестве продукции и возможность определить нахождение поставщика неисправного компонента имеет решающее значение для агентов промышленности. Технология Blockchain может повысить прозрачность, темп и реагирование этих сложных систем. Технология позволяет создать общую и надежную площадку для ведения и обновления информации о поставщиках, спецификации услуг и запчастей, а также процедуры разрешения конфликтных ситуаций. Также создание площадки с использованием технологии распределенного реестра позволит обеспечить более быструю и эффективную интеграцию новых поставщиков. Все вышеперечисленное призвано способствовать значительному снижению рисков авиакрушений по техническим причинам.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Кличиков А.В., Пряников М.М., Чугунов А.В. Блокчейн-технологии и их использование в государственной сфере. – International Journal of Open Information Technologies №12/2017.
2. Сатоши Накамото. Биткоин: Одноранговая электронная кассовая система [Электронный ресурс]. 2008. 9 с. URL: <https://bitcoin.org/bitcoin.pdf> (дата обращения: 30.01.2019).

УДК 004.9

Воронина О.А. Медведев Р.Е.

СТРУКТУРА АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ ФИНАНСОВО-ЭКОНОМИЧЕСКОГО АНАЛИЗА

Рязанский государственный радиотехнический университет,

г. Рязань

В статье приведена структура автоматизированных информационных систем финансово-экономического анализа предприятий, а также выделены требования к обрабатываемой в сис-темах информации

Эффективное развитие любой организации в условиях конкурентной рыночной борьбы предопределяет качественный аналитический мониторинг ее финансового состояния, целями которого являются построение объективной информационной картины текущей деятельности и определение перспектив дальнейшего развития. Под финансовым состоянием

понимается способность организации финансировать свою деятельность, оно характеризуется целесообразностью размещения финансовых ресурсов, их обеспеченностью и эффективностью использования. Своевременный анализ показателей финансовой деятельности способствует выявлению кризисных ситуаций на этапе их зарождения, принятию мер оперативного вмешательства, уменьшению сопутствующих рисков и обеспечивает достижение поставленных стратегических целей и решение текущих задач.

В процессе анализа происходит обработка исходных данных, вычисление промежуточных значений и построение модели предметной области, представляющей собой сложную систему взаимозависимых показателей [1]. Несмотря на то, что структура и содержание модели определяется целями анализа и особенностями деятельности самой организации, этот процесс очень трудоемок и требует сбора, обобщения и обработки огромных массивов информации, заключенной в различных источниках и документах. При этом оперативный и качественный анализ невозможен без применения специализированных программных продуктов, которые с помощью заложенных в них алгоритмов используют различные методики расчета ключевых финансовых показателей и предоставляют результаты в требуемом для конечного пользователя формате.

Поэтому к числу наиболее важных задач развития предприятия целесообразно отнести совершенствование процессов автоматизации обработки информации о результатах финансово-хозяйственной деятельности организации, так как в условиях современной рыночной экономики качество и своевременность мониторинга финансового состояния напрямую влияет на повышение экономической эффективности и укреплению организации на рынке товаров и услуг.

В общем случае автоматизированные информационные системы финансово-экономического анализа (далее – АИСФЭА) представляют собой сложный программный комплекс [2], состоящий из нескольких подсистем, структура и функции которых зависят от сферы применения программного продукта, но несмотря на это необходимо выделить общие составляющие, а именно:

- подсистема расчета финансовых показателей, служит для анализа финансовых показателей экономического субъекта, в число которых могут входить: имущественное положение, деловая активность, финансовая устойчивость, структура капитала, соотношение и структура собственного и заемного капитала, ликвидность, платежеспособность, прибыль, рентабельность, сопутствующие риски;

- подсистема анализа бухгалтерской отчетности, содержит набор инструментов для статистического и аналитического анализа информации о бухгалтерском балансе, финансовых результатах, движении денежных средств и изменении капитала;

– подсистема расчета аналитических показателей, предназначена для формирования аналитической отчетности разной степени полноты и точности. В процессе расчетов аналитические показатели детализируются, а также выстраиваются в определенные последовательности, логически сгруппированные по направлениям финансового анализа;

– подсистема отчетности, предоставляет пользователю набор гибких инструментов формирования статистической и аналитической отчетности и последующего построения информационно-аналитической модели результатов финансового анализа, с сопутствующей подготовкой рекомендаций для выбора дальнейших путей стратегического развития предприятия.

Для достижение требуемого качества результатов работы АИСФЭА, к информации, обрабатываемой в подобного рода системах предъявляется ряд требований, а именно: достоверность, возможность документального подтверждения обрабатываемой информации; полнота, заключается в степени сформированности базы требуемых аналитических показателей; понятность, соответствие информации требуемым стандартам; актуальность, степень применения показателей в процессе анализа; своевременность, заключается в соответствии показателей анализируемому периоду; значимость, определяет меру необходимости расчета показателей и обработки текущей информации; последовательность, определяет порядок расчета и использования информации во времени.

На сегодняшний день на рынке современных финансово-аналитических продуктов представлено достаточно большое количество инструментов для анализа финансово-экономической деятельности предприятия. Каждый программный продукт имеет свои преимущества и недостатки, связанные в первую очередь с высокой ценой и набором инструментов проведения анализа. Вместе с тем из-за специфичности представленных программных продуктов в большинстве случаев достаточно сложно выбрать систему, учитывающую все стороны деятельности конкретного предприятия. К числу наиболее популярных систем, представленных на российском рынке целесообразно отнести следующие:

«Audit Expert» (<https://www.expert-systems.com/financial/ae/>) – представляет собой аналитическую систему диагностики, мониторинга и оценки финансового состояния предприятия. Результатом обработки информации, соответствующей требованиям международных стандартов финансовой отчетности, являются экспертные заключения о финансовом состоянии предприятия. Помимо этого система отчетности предполагает построение гибких графиков и диаграмм, отражающих динамику ключевых показателей.

«Бизнес-Аналитик» (<http://inec.ru/it/automated-analysis/ba/>) – предназначен для решения задач финансового планирования и бюджетирования, а также комплексного анализа производственно-финансовой деятельности предприятия на основе единой информационной базы. Одним из резуль-

татов работы программного комплекса является сформированный бизнес-план и текстовое заключение о финансовом состоянии предприятия.

«Про-Инвест-ИТ» (<http://www.k-press.ru/comp/2002/4/proinvest/proinvest.asp>) – информационная система финансовой диагностики, предоставляющая набор инструментов как для финансовой диагностики, так и анализа текущего финансового состояния предприятия, а также перспектив его развития. Алгоритмы системы позволяют спрогнозировать значения требуемых финансовых показателей в будущем.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Титоренко Г.А. Автоматизированные технологии в экономике: Учебник – М.: ЮНИТИ, 2009. 399 с.

2. Баскакова О.В. Экономика предприятия (организации): Учебник – М.: Дашков и К, 2013. – 372 с.

УДК 339.1

Рябых А.Н.

ЦИФРОВЫЕ ПЛАТФОРМЫ КОММУНИКАЦИЙ ДЛЯ ФОРМИРОВАНИЯ ПРОДАЖ

Рязанский государственный радиотехнический университет,
г. Рязань

В работе рассматриваются перспективы применения цифровых платформ для совершенствования работы торгового предприятия.

Торговля в современной экономике является одним из главных факторов экономического развития, где внедряются и адаптируются новые высокие технологии, что сильно отражается на самих принципах торговой и коммерческой деятельности.

Отметим, что именно розничная торговля – наиболее динамичный сегмент, демонстрирующий наступление перемен в экономической жизни общества. В последнее время популярна концепция «pop up store», которая активно заключается в торговле. Многие эксперты отказываются признавать новый формат, который свидетельствует о начале циклического изменения в отрасли.

Технология продаж в формате «pop up store» явилась переносом рекламных решений, широко используемой в современном цифровом пространстве, в числе компаний, активно использующих новую технологию оказались сеть дискаунтеров Target, Gucci, Hermes, Louis Vuitton, Chanel, IKEA, TopShop и другие.

Термин произошел от английского «pop-up» («внезапно появляться») и «store» («магазин»). В 2004 году появилась идея о том, что, если новые продукты могут приходить и уходить, то почему не могут делать магазины: их распространяющие?

Рор-уп магазин – это временный магазин, который открывается на короткий срок (день, неделя, месяц) с целью повышения продаж и привлечения новых покупателей. Даты открытия и закрытия магазинов оглашаются заранее и обычно приурочиваются к значимому событию [2].

Специалисты в области маркетинга и торговли выделяют следующие конкурентные преимущества нового формата:

- сокращение издержек на аренду;
- визуализация предмета продаж;
- эффективная форма оценки спроса;
- поддержка лояльности уже привлеченных покупателей;
- существенный рост продаж;
- возможность эффективного тестирования новых продуктов;
- сокращение издержек на рекламу.

Кроме того, появление рор-уп store конкретной компании существенно подчеркивает силу конкурентной позиции этой компании и является своеобразным индикатором конкурентного преимущества, его наглядной демонстрацией для известных компаний, и в тоже время для новых начинающих компаний появляется возможность входа на рынок на равных условиях при существенном сокращении рыночных барьеров.

Тем не менее, если выше рассмотрены современные особенности трансформации в розничной торговле, то, безусловно, определенные изменения должны также происходить в организациях, обеспечивающих их функционирование.

В России формат деятельности оптовых организаций существенно стал видоизменяться с приходом западных торговых сетей и активной экспансией российских сетей в регионы, что привело к уменьшению объемов продаж оптовых организаций.

Данная тенденция сохраняется и сейчас, следовательно, в современных условиях развития сетевой торговли и разновидности – локальных розничных сетей, оптовый посредник и предприятия розничной торговли должны объединять и координировать свою деятельность, исходя не из своих интересов, а как партнеры, заинтересованные в совместном достижении общей цели качественного и своевременного обеспечения покупателей качественными товарами [1].

Этому будет способствовать формирование и развитием коммуникационных платформ на основе современных IT-технологий.

Одна из таких платформ уже функционирует в торговле на базе технологии блокчейн, обеспечивающей возможность формирование распределенных баз данных о происхождении товара.

Таким образом, логистизация деятельности на базе проектного подхода может стать коммуникационной платформой взаимодействия для предприятия сферы торговли, позволяющей организовать современные потоки товародвижения и сократить затраты на ведение деятельности всех партнеров, участвующих в цепочке создания ценности.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Логинов Е.Л., Райков А.Н. Цифровая экономика: уязвимость к сетевым атакам и возможности обеспечения устойчивости управления // Проблемы рыночной экономики, 2017, №4, с. 4-10.

2. Конева Е. 10 причин открыть POP UP магазин [Электронный ресурс]. URL: <https://www.ecwid.ru/blog/10-reasons-to-open-pop-up-store.html> (дата обращения: 30.01.2019).

УДК 004.9

Минашкина А.Н. Медведев Р.Е.

КЛЮЧЕВЫЕ КРИТЕРИИ ВЫБОРА АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ БУХГАЛТЕРСКОГО УЧЕТА АВТОНОМНЫХ УЧРЕЖДЕНИЙ

Рязанский государственный радиотехнический университет,
г. Рязань

В статье рассмотрены основные задачи и предложены ключевые критерии выбора автоматизированных информационных систем бухгалтерского учета автономных учреждений

Развитие современной экономики и изменения, произошедшие в экономической сфере в течении последнего десятилетия создали условия для возникновения новых субъектов экономической деятельности, к числу которых следует отнести автономные учреждения, под которыми понимаются некоммерческие организации, созданные Российской Федерацией, субъектом Российской Федерации или муниципальным образованием для выполнения работ, оказания услуг в целях осуществления предусмотренных законодательством Российской Федерации полномочий органов государственной власти, полномочий органов местного самоуправления в сферах науки, образования, здравоохранения, культуры, средств массовой информации, социальной защиты, занятости населения, физической культуры и спорта, а также в иных сферах в случаях, установленных федеральными законами (в том числе при проведении мероприятий по работе с детьми и молодежью в указанных сферах) [1].

Эффективное ведение финансово-хозяйственной деятельности и дальнейшее развитие автономного учреждения тесно связаны с грамотной политикой в области бухгалтерского учета, под которым в соответствии с федеральным законом «О бухгалтерском учете» № 402-ФЗ от 06.12.2011 г. понимается формирование документированной систематизированной информации об объектах и составление на ее основе бухгалтерской (финансовой) отчетности [2].

Бухгалтерская деятельность автономного учреждения предполагает качественный анализ и структуризацию большого количества хозяйственных операций, необходимых для ведения оперативного, синтетического и аналитического учёта, а также формирования форм бухгалтерской отчёт-

ности и информации, необходимой для принятия грамотных управленческих решений, а также постоянного контроля за ходом производственных процессов учреждения. При этом применение автоматизированных информационных систем учета (далее – АИСУ) существенно уменьшает трудоёмкость учета, в первую очередь за счет автоматизации процессов формирования налоговой и бухгалтерской отчетности, ведения сопутствующей документации, параллельного учета и снижения влияния «человеческого фактора» на качество обработки информации [3]. Таким образом внедрение современных АИСУ в деятельность автономного учреждения является объективной необходимостью, вызванной высокими требованиями к качеству ведения бухгалтерской отчетности и направленной на повышение эффективности принятия управленческих решений.

К числу зада, решаемых АИСУ при ведении бухгалтерского учета в автономном учреждении, целесообразно отнести следующие:

- автоматизированный учет финансово-хозяйственных операций;
- единообразие выполнения операций, позволяющее исключить случаи возникновения случайных ошибок и погрешностей расчетов;
- создание, печать и хранение электронных копий первичных банковских и кассовых документов; разделение функций внутреннего контроля деятельности различных подразделений предприятия;
- автоматический подсчет баланса, развернутого и свернутого сальдо, оборотов, составление главной книги, журналов-ордеров, и прочих отчетных форм; автоматическое выполнение ряда необходимых расчетов и формирование статистической и аналитической отчетности;
- предоставление руководству широкого набора инструментов аналитического и статистического анализа деятельности предприятия.

В настоящий момент на рынке программного обеспечения существует множество АИСУ отечественного производства:

«1С:Бухгалтерия» (<https://1c.ru/rus/products/products.htm>) – программный комплекс представляющий собой набор инструментов, настраиваемых на решение специализированных задач и адаптированных к специфике учета конкретной автономного предприятия;

«АВЕРС: Бухгалтерия» (<http://www.ицаверс.рф/produktsiya/buhgalteriya>) – автоматизированная информационно-аналитическая система предназначенная для автоматизации процессов ведения бухгалтерского учета и информационного обеспечения управления финансово-хозяйственной деятельностью во всех типах учреждений бюджетной сферы;

«ТурбоБухгалтер 6.X» (http://www.pcapital.ru/produkty/turbo_buhgalter) – автоматизированная информационная система, содержащая алгоритмы реализующие мемориально-ордерную форму ведения бухгалтерского учета;

«ПАРУС-Бухгалтерия» (<http://parus.ua/161/>) – автоматизированная информационная система учета персонала, труда и заработной платы,

деятельности планово-экономического отдела, распределения ассигнования и консолидации отчетов от подотчетных организаций;

«БЭСТ-5» (<https://www.bestnet.ru/programs/best-5/>) – система управления предприятием, располагающая развитыми и гибкими средствами регистрации фактов хозяйственной деятельности в оперативном режиме, с одновременным их отражением в бухгалтерском, налоговом и управленческом учете.

Каждая из рассмотренных систем хорошо зарекомендовала себя на рынке и предоставляет широкий набор инструментов бухгалтерского учета и методов формирования сопутствующей отчетности, однако помимо достоинств данные системы имеют специфичные недостатки, поэтому с уверенностью сказать о выборе для автономных учреждений того или иного программного продукта, нельзя, так как выбор должен быть основан на специфичных для учреждения характеристиках и области деятельности. Таким образом при выборе АИСУ для автоматизации деятельности автономного учреждения необходимо учитывать:

- производительность системы (возможность одновременной безпроблемной работы заданного количества пользователей);
- возможность обработки и организацию хранения большого объема разнотипных бухгалтерских документов, охватывающих все разделы бухгалтерского учета в автономном учреждении;
- максимизация предоставляемого системой набора инструментов при минимизации затрат на закупку системы и последующее информационно-техническое сопровождение;
- авторизованный доступ и высокая безопасность хранимых и обрабатываемых системой данных;
- разграничение функциональных обязанностей между различными категориями пользователей;
- соответствие выходных форм отчетных документов, формируемых системой требованиям федерального законодательства РФ;
- совместимость системы с уже внедренными в учреждении программными продуктами;
- обработка информации по различным типам банковских счетов с формированием учета по каждому из них и консолидированной отчетности по учреждению в целом;
- расширяемость функционала в случае возникновения необходимости;
- своевременный учет изменений законодательства Российской Федерации.

При учете вышеперечисленных показателей в процессе выбора АИСУ автоматизация бухгалтерской деятельности автономного предприятия позволит повысить скорость принятия управленческих решений, достоверность аналитической и статистической информации о деятель-

ности учреждения, а также качество ведения бухгалтерского учета, что несомненно повысит эффективность деятельности учреждения и переведет процесс его дальнейшего развития на новый, более качественный уровень.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Приказ Минфина РФ от 16.12.2010 г. № 174н (ред. от 16.11.2016) «Об утверждении Плана счетов бухгалтерского учета бюджетных учреждений и Инструкции по его применению» // СПС «Консультант Плюс».
2. Федеральный закон от 06.12.2011 N 402-ФЗ (ред. от 28.11.2018) «О бухгалтерском учете» // СПС «Консультант Плюс».
3. Одинцов, Б.Е. Информационные системы управления эффективностью бизнеса: учебник и практикум для бакалавриата и магистратуры / Б.Е. Одинцов. – М.: Издательство Юрайт, 2015. 206 с.

УДК 00.004

Крылова М.А., Медведев Р.Е.

АНАЛИЗ СОВРЕМЕННЫХ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ ПЕРСОНАЛОМ ИСПОЛЬЗУЮЩИХ МЕТОДИКУ ОЦЕНКИ ПЕРСОНАЛА ПО КРІ

Рязанский государственный радиотехнический университет,

г. Рязань

В статье проведен сравнительный анализ современных автоматизированных систем управления персоналом, использующий оценку по ключевым показателям эффективности деятельности кадров.

В современной экономике уровень конкурентоспособности, а значит и эффективность деятельности предприятий напрямую зависит от качества трудовых ресурсов, процессов их формирования, использования и развития, поэтому одно из первых мест, в процессе достижения стратегических целей, занимает эффективное управление персоналом организации. В современной литературе существует множество трактовок, однако наиболее популярное определение дает Маслова В.М., а именно: управление персоналом – это совокупность принципов, средств и методов целенаправленного воздействия на персонал организации, позволяющие максимально использовать его интеллектуальные и физические способности при выполнении трудовых функций для достижения организационных целей.

Управление персоналом является сложным процессом, характеризующимся специфическими особенностями и закономерностями и предусматривающим, в качестве основных направлений деятельности планирование, перераспределение, формирование и рациональное использование трудовых ресурсов.

Управление персоналом тесно связано с процессом качественного развития организации и заключается в: проектировании организационной

структуры; определении потребности и расчете численности персонала; разработке системы сбора, использования и хранения кадровой информации; анализе деятельности работников организации; определении критериев оценки для подбора персонала; анализе кадровой ситуации на рынке труда и сопутствующем формировании стратегии управления персоналом в рамках комплексной стратегии развития организации.

Одним из основных инструментариев формирования комплексной стратегии управления персоналом организации являются автоматизированные системы управления персоналом (далее – АСУП), которые позволяют: автоматизировать расчёты заработной платы работников; упорядочить расчеты, связанные с управлением персоналом (командировочные расчеты, начисления и удержания, наградные); усовершенствовать процесс управления мотивацией кадров; осуществлять грамотную оперативную своевременную оценку персонала; осуществлять грамотное планирование использования персонала; усовершенствовать процесс обучения и управление карьерой работников (своевременное определение необходимости прохождения обучения); осуществлять оперативное управление системой аттестации персонала; производить автоматизированный анализ эффективности персонала, анализ соответствия работников занимаемым должностям; моделировать штатное расписание и оптимизировать работу с ним; наладить постоянное «информационное обслуживание» сотрудников компании.

В основе своей АСУП используют различный набор современных методов оценки персонала с учетом приоритетов и потребностей конкретной организации, позволяющих на необходимом качественном уровне измерить достижимость поставленных перед работниками организации целей. Анализ наиболее популярных методов оценки персонала представлен в табл. 1.

По результатам представленного анализа целесообразно сделать вывод, что наиболее эффективным и многопрофильным методом оценки персонала является оценка по ключевым показателям эффективности (оценка КРІ). Данный метод позволяет проводить анализ результатов деятельности каждого работника предприятия по отдельности и работу предприятия в целом, а также позволяет выявлять необходимость проведения ряда мероприятий (мотивация, стимулирование персонала, ротация, управление карьерой сотрудников, пересмотр кадровой и общей стратегии). Важным преимуществом метода оценки по КРІ является возможность его автоматизации. Среди наиболее популярных современных АСУП, использующих метод оценки по КРІ целесообразно выделить следующие (табл. 2):

ELMA КРІ – автоматизированная система, которая позволяет строить карту стратегических и операционных целей компании, а после декомпозировать их до отдельных сотрудников. Использование BPM-платформы для создания матриц результативности позволяет выделить, измерить и

интерпретировать любые показатели работы сотрудника (количественные и качественные) [1].

Т а б л и ц а 1. Сравнительный анализ методов оценки персонала с учетом потребностей организации

Приоритеты (потребности) \ Метод	Assessment- центр	Оценка по KPI	Метод «360 градусов»	Тестирование	Метод РМ
Повышение производительности труда	+	+	—	—	+
Объективная оценка результатов работы	+	+	—	—	+
Невысокая стоимость реализации	—	—	+	+	—
Невысокие временные затраты	—	+	+	+	—
Возможность оценки большого числа работников	—	+	+	+	+
Связь со стратегией развития предприятия	—	+	—	—	+
Ориентация на обучение и управление карьерой работников	+	+	—	—	+
Возможность использования результатов оценки для построения системы мотивации и стимулирования	+	+	—	—	+
Возможность автоматизации процесса оценки	—	+	—	+	—

HighPer – программный продукт, позволяющий собирать данные из других сервисов учёта персонала, используемых на предприятии, оптимизировать управление возложенными на работниками задачами и получать оперативный доступ к отслеживанию показателей их выполнения [2].

KPI-Drive – авторский программный продукт с собственной методикой внедрения системы оценки по KPI. Предусматривает такие модули как: показатели, задачи, оценка, оплата (расчет премии, зарплаты). Вся информация по сотрудникам, показателям и фактам доступна к загрузке и выгрузке из внешних программ [3].

KPI Monitor – готовое программное решение для оперативной оценки эффективности работы организации посредством автоматизации системы управления ключевыми показателями эффективности. Программный продукт может быть реализован в компаниях любого вида сложности и отраслевой специализации. KPI Monitor предоставляет гибкие средства интеграции с всевозможными информационными системами (WMS, ERP и CRM-системы, бухгалтерские программы) в целях сбора данных, необходимых для создания показателей. Также, данный программный продукт поддерживает автоматическую синхронизацию аналитических справочников с корпоративной информационной системой, что обеспечивает компании единую систему аналитики [6].

1С:Управление по целям и KPI – программный продукт, позволяющий автоматизировать постановку целей и контроль результативности (стратегический и операционный цикл), управлять вовлеченностью и мотивацией персонала, отслеживать динамику показателей деятельности по отдельным функциональным сферам.

Т а б л и ц а 2. Сравнительный анализ современных программных обеспечений для автоматизации KPI

Название ПО	Параметры						
	Разработчик		Кадровый функционал	Технологическая платформа	Интерфейс	Масштабируемость	Стоимость (тыс. руб.)*
HighPer	Специалисты по KPI	Компания «Проект «Дельфы»	3	Microsoft .NET Framework + Microsoft SQL	5	2	345
KPI-Drive		А. Литягин	4	Apache	4	3	337
ELMA KPI	IT-разработчики	EleWise	3	Microsoft .NET Framework + Microsoft SQL	3	4	307
KPI Monitor		ПроФитПроект	5		5	5	≈ 300
1С:Управление по целям и KPI		Волгасофт	5	1С: Предприятие	4	5	≈ 250

По результатам приведенного анализа целесообразно выявить лидера, а именно автоматизированную систему KPI Monitor, позволяющую с наименьшими затратами контролировать выполнение требуемых задач сотрудниками организации и автоматизировать создание оценочных таблиц деятельности сотрудников с возможностью применения шаблонов матриц KPI и синхронизации результатов с целями и приоритетами организации. Применение программы KPI Monitor облегчит систему оценки сотрудников предприятия, позволит контролировать выполнение установленных задач, быстро рассчитывать премии по каждому работнику и повышать результативность работы предприятия.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Управляйте движением к цели с помощью понятных ключевых показателей эффективности в системе ELMA KPI [Электронный ресурс] URL: <https://www.elma-bpm.ru/product/kpi/> (Дата обращения: 30.01.2019);
2. HighPer Автоматизация оплаты труда по KPI [Электронный ресурс] URL: <http://highper.ru/> (Дата обращения: 30.01.2019);
3. KPI Drive. Система Управления и Мотивации [Электронный ресурс] URL: <https://kpi-drive.ru/> (Дата обращения: 30.01.2019).

УДК 591.685

Блохина И.О.

АВТОМАТИЗАЦИЯ БИЗНЕС-ПРОЦЕССОВ НА ПРЕДПРИЯТИИ

Рязанский государственный радиотехнический университет,
г. Рязань

В статье рассмотрены методы управления бизнес-цепочками внутри компании, роль и методика автоматизации бизнес-процессов на промышленных предприятиях.

Методы управления бизнесом в современном мире резко отличаются от опыта прошлых поколений. В настоящее время для достижения поставленных задач используются различные способы, среди которых наиболее распространенным и действенным становится автоматизация бизнес-процессов.

Процесс автоматизации бизнес-процессов на предприятии заключается в том, что выполнение типовых и стандартных задач переходит под контроль программно-аппаратного комплекса. Актуальность данной темы очевидна, ведь при правильном подходе применение бизнес-процессов способствует высвобождению ресурсов, что, несомненно, ведет к повышению эффективности производства.

С развитием предприятия автоматизация бизнес-процессов становится неотъемлемой частью его деятельности. Использование на практике автоматизированных систем, ориентированных на обработку документов, значительно упрощает процесс ввода больших массивов данных, осуществление контроля остатков материальных ценностей и т.д. Автоматизированный контроль за соблюдением нормативных требований к выпускаемой продукции позволяет повысить ее качество [4]. С помощью данных систем руководитель имеет возможность быстро выявить наиболее слабые места в кадровом составе фирмы, что приведет к снижению затрат за счет исключения этих звеньев [1].

В нормативной документации каждого предприятия обычно закреплены основные функции, которые должны осуществлять сотрудники. В идеале эти процессы выполняются четко и бесперебойно. Однако на практике не все так гладко. Если задачи, поставленные руководителем, не исполняются должным образом, если зачастую наблюдаются перебои в работе, а также происходит потеря и искажение информации, то это является сигналом о необходимости автоматизации бизнес-процессов. Правильная организация структуры управления позволяет сотрудникам любых уровней работать максимально эффективно. На рис. 1 показаны этапы, на которых основана методика автоматизации бизнес-процессов.

Перед началом процесса автоматизации бизнес-процессов у каждого руководителя должны быть намечены конкретные цели. Он должен четко представлять, что он хочет увидеть в результате внедрения. Цели могут

быть разными. Они зависят, во-первых, от характера деятельности фирмы. Также необходимо получить сведения о наиболее уязвимых местах предприятия (например, логистика, управление запасами и т.д.) [3]. От того, насколько правильно поставлены задачи, зависит эффективность использования бизнес-процесса.



Рис. 1. Этапы автоматизации бизнес-процесса на предприятии

Суть формализации бизнес-процесса состоит в определении располагаемых ресурсов, наличие и описание подпроцессов, последовательность движения информации и выполнения мероприятий. На основе представленных выше данных специалистами выстраивается модель оптимизации бизнес-процессов. Для этого на основе проведенного анализа намечают план по достижению намеченных результатов, который исключает лишние простои и задержки. В ходе разработки технического задания по согласованию с руководителем определяются точные задачи, которые будут включены в бизнес-процесс.

Организация, занимающаяся автоматизацией, выполняет кодирование информации. Для того, чтобы бизнес-процесс был доступен для применения на практике, разрабатывают должностные инструкции, регламентирующие порядок действий при использовании информационных систем.

Оценить, насколько эффективным оказалось внедрение бизнес-процесса на предприятии возможно только опытным путем. В процессе работы будут выявлены как положительные результаты, так и недостатки, которые устранят в процессе корректировки. В результате внедрения бизнес-процесса на предприятии возможно добиться следующих результатов [2]:

1. Упрощение учетных процедур;
2. Возможность обработки bigdata;
3. Оптимизация издержек;
4. Увеличение скорости работы;
5. Улучшение качества продукции;
6. Упрощение процесса контроля за ходом деятельности предприятия.

Не стоит забывать о том, что необходима интеграция бизнес-процесса на все уровни организационной структуры, занятой в конкретной

сфере. Нельзя автоматизировать лишь часть процесса, так как положительного результата данная система не принесет.

При всей простоте и наглядности данной методики, на практике не всегда удастся быстро и легко осуществить переход к новой системе. Поэтому необходимо выявить переходный период, в течение которого фирма будет осваивать новый процесс.

Таким образом, автоматизация бизнес-процессов на предприятии является достаточно продолжительной процедурой, состоящей из отдельных этапов и включающей в себя слаженную работу практически всей компании. Предпринимателям не стоит откладывать их внедрение в долгий ящик, ведь от того, насколько рационально используются трудовые и материальные ресурсы, зависит успех и эффективность деятельности предприятия.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Громов А.И. Управление бизнес-процессами: современные методы. - Люберцы: Юрайт, 2016, – 367с.
2. Джестон Д. Управление бизнес-процессами. Практическое руководство по успешной реализации проектов. – М.: Символ, 2015. – 512 с.
3. Репин В.В. Бизнес-процессы. Моделирование, внедрение, управление. – М.: Манн, Иванов и Фербер, 2016. – 512 с.
4. Тельнов Ю.Ф. Инжиниринг предприятия и управление бизнес-процессами. Методология и технология. – М.: ЮНИТИ, 2015. – 176 с.

УДК 51-74

Коротаев А.Н., Макашкин Я.О., Благодарова Т.А.

ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ЛЕТНОЙ ЭКСПЛУАТАЦИИ ВОЗДУШНЫХ СУДОВ ГРАЖДАНСКОЙ АВИАЦИИ

Рязанский государственный радиотехнический университет,

г. Рязань

В статье рассматривается предугадывание наступления событий или явлений, отличающиеся принципиальностью свойств.

Моделирование ситуаций ставит перед собой разные цели, главной из которых является предугадывание наступления событий или явлений, отличающиеся принципиальностью свойств. Зачастую их относят к событиям, которые вероятнее всего произойдут в будущем, но так же бывает, что добиться наступления желаемых условий, для наступления предполагаемых событий, экспериментальным методом невозможно. В такой ситуации в дело вступает методика математического регулирования, с помощью которой создается множество прогнозов наступления тех или иных явлений. Часто при описании математической модели элемент контроля остается без внимания, но, например, при рассмотрении особенностей полета и параметров движения летательных аппаратов

фактор контроля становится важнейшим элементом исследования. С помощью Математической модели упрощается реальная картина происходящего, это нужно для того что бы отбросить ненужные факторы и сконцентрироваться на исходной задаче, такая задача становится идеализированной и легко поддается анализу с помощью мощной вычислительной технике. Современный этап развития авиастроения характеризуется повышенным вниманием к динамике летательных аппаратов (ЛА), это связано со стремлением уменьшить лобовое сопротивление воздушного судна, что позволило бы снизить расходы топлива, улучшить характеристики маневренности дальности и продолжительности полета. Таким образом расчетные методы должны соответствовать такому уровню точности, который смог бы позволить достичь высокой эффективности при эксплуатации ЛА, и конечно немалую важность при эксплуатации ЛА является достижение достаточного уровня безопасности полетов (БП). (1)

В развитие авиации явно просматривается постепенное уменьшение количества летных экспериментов за счет внедрения исследований с применение математических моделей, что позволяет отказаться от дорогостоящих, опасных, а порой и невыполнимых задач при проведении экспериментов. Кроме того, большой ряд условий при проведении летных испытаний (ЛИ) бывает трудно, а порой и опасно, реализовывать, к таковым относятся особые случаи в полете, связанные с отказом техники, ухудшением метеоусловий, а так же этапы взлета и посадки.

Таким образом, использование математических моделей (ММ) движения самолета в особых случаях взлета и посадки становится не только удобным, но и необходимым средством решения задач летных экспериментов (ЛЭ). (2) В связи с этим ЛИ уступают место гораздо более безопасному и экономически более выгодному, но не менее эффективному методу ММ движения ЛА. Такая модель реализована в виде системы математического моделирования динамики полета летательных аппаратов (СММ ДП ЛА). СММ представляет собой развитый комплекс программного обеспечения с методиками планирования, проведения и обработки результатов вычислительных экспериментов (ВЭ). Эта методика позволяет оперативно получать информацию, а главное гарантирует ее точность. Таким образом ЛИ, требующие больших финансовых и человеческих ресурсов, уступает место методу ММ, оставляя за собой лишь функцию подтверждения результатов.(3)

Повышение уровня БП и эффективности ЛЭ ЛА достигается путем последовательного, всестороннего анализа всех факторов, оказывающих влияние на поведение ВС, а, следовательно, и на БП. Первоочередно рассматриваются такие факторы, как: отказы авиационной техники, состояние погодных условий, а также действия экипажа. Таким образом эффект от использования системы математического моделирования динамики полета летательного аппарата, позволяющее учесть факторы и условия,

влияющие ЛЭ при эксплуатации ВС, а так же на БП, невозможно переоценить ввиду дешевизны, доступности, а главное эффективности данного средства анализа поведения ВС. (4)

Разработка методики вычислительного эксперимента позволила:

- внести ряд рекомендаций в РЛЭ ВС – уменьшить длину разбега и пробег за счет выбора оптимального положения закрылков;
- определить оптимальные методы пилотирования, рассматриваемого типа ВС, в сложных условиях полета;
- сократить уровень шума на местности путем оптимизации приемов пилотирования ВС;
- обеспечить экономию ресурсов за счет сокращения объема ЛИ;
- осуществлять более детальную и качественную работу в рамках исследования, проектирования, а так же эксплуатации воздушных судов и аэродромов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Жулев В.И., Иванов В.С. Безопасность полетов летательных аппаратов. – М.: Транспорт, 1986. 223 с.
2. Бекмуханбетов М.Д., Ципенко В.Г. Обоснования выбора оптимальных углов отклонения закрылков самолета Ил-96-300 на различных этапах полета Научный Вестник УВАУГА, № 1. - Ульяновск : РИО УВАУГА, 2008. – с. 6-16.
3. Байкулова Н.И., Кузьмина Ю.Е., Полякова И.Ф., Ципенко В.Г. Математическое моделирование взлета транспортного самолета в сложных метеоусловиях // Методы инженерного обеспечения безопасности полетов. – М.: МИИГА, 1985. С. 95-102.
4. Котик М.Г., Павлов А.В., Пашковский И.М. Летные испытания самолетов. 2-е изд. М.: Машиностроение, 1968. – 423 с.

УДК 65.011.56

Байкин П.С.

РАЗРАБОТКА СИСТЕМЫ КРАТКОСРОЧНОГО ПРОГНОЗИРОВАНИЯ СПРОСА НА ПРОДУКЦИЮ С ИСПОЛЬЗОВАНИЕМ ПРИНЦИПА САМООРГАНИЗАЦИИ

Рязанский государственный радиотехнический университет,

г. Рязань

В статье рассмотрен алгоритм разработки системы, осуществляющий прогноз с использованием принципа самоорганизации, применены методы простого экспоненциального сглаживания, метод Тригга-Лича

Решение задач краткосрочного прогнозирования вызывают особые затруднения, поскольку они требуют высокую точность. Широко известные методы статистического прогноза не обеспечивают достаточную точность прогнозирования в условиях быстро меняющихся условий рынка товаров и услуг. Поэтому необходимо разработать систему, которая будет

осуществлять прогноз с использованием принципа самоорганизации, который отражает изменяющиеся во времени условия [3].

Только организация систем управления предприятием во взаимосвязи всех функций управления, включая прогнозирование, на основе информационно-аналитических аспектов является одним из важнейших направлений повышения эффективности деятельности предприятий в целом.

Разработка системы прогнозирования временных рядов динамики, предназначенной для определения краткосрочного прогноза на примере решения задачи прогнозирования спроса.

Современные условия на рынке управления и бизнеса характеризуются ужесточением конкурентной борьбы на рынках сбыта, трансформируются в конкуренцию организаций, в соперничество знаний и управление ими в целях достижения рыночного преимущества. А это предполагает применение гибких систем управления, которые помогают принять эффективные управленческие решения в условиях неопределенности.

Для реализации поставленной задачи необходимо средство, которое позволит разработать систему удобную и мощную, а также модифицируемую и отвечающую современным требованиям к программному обеспечению.

Для разработки системы прогнозирования я выбрал инструментальное средство разработки Microsoft Visual Studio, Microsoft Access 2003.

В большинстве приложений применяются два типа прогностических моделей: экспоненциальное сглаживание и регрессия. Методы, основанные на экспоненциальном сглаживании, предназначены для краткосрочного прогнозирования. Они применяются, как правило, к данным как, месячный спрос, сумма продаж за квартал и.т.д. Методы регрессионного выравнивания применяются в среднесрочном прогнозировании. В системе выбраны следующие методы решения прогнозных задач: Метод простого экспоненциального сглаживания, адаптивное сглаживание прогноза Брауна, метод Тригга – Лича, самоорганизующийся метод прогнозирования. Данные методы были выбраны для краткосрочного прогнозирования по временному ряду [1].

1. Метод простого экспоненциального сглаживания.

Если ряд фактических значений показателя и константа сглаживания, то экспоненциально сглаженным рядом будет ряд, получаемый по формуле [4]

$$f_{t+\tau} = \alpha d_t + (1 - \alpha)f_t$$

где $f_{t+\tau}$ – прогноз на момент времени $t + \tau$; t – текущий момент времени; τ – период упреждения прогноза; d_t – фактическое значение наблюдаемого показателя; α – константа сглаживания ($0,1 \leq \alpha \leq 1$).

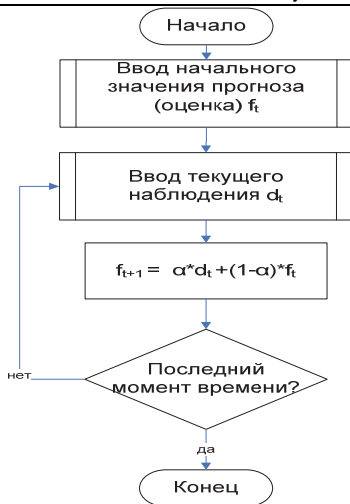


Рис. 2. Блок-схема вычисления прогноза методом экспоненциального сглаживания

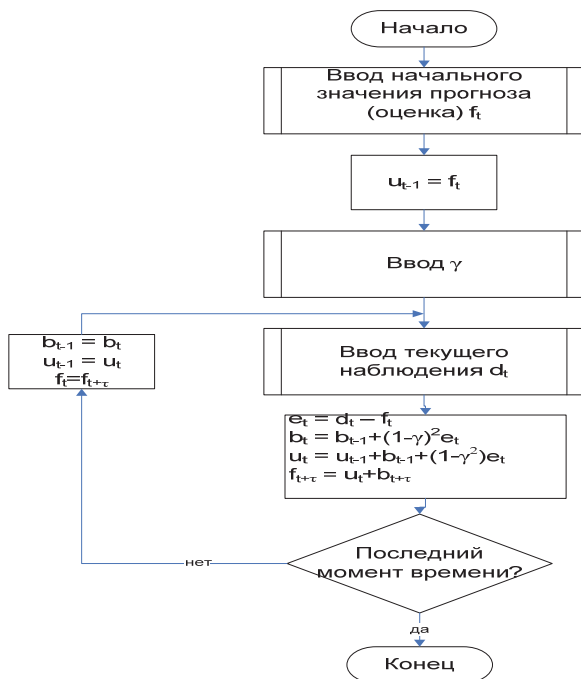


Рис.2. Блок-схема вычисления адаптивного сглаженного прогноза Брауна

2. Адаптивное сглаживание прогноза Брауна.

Этот метод основывается на вычислении оценок по методу взвешенных наименьших квадратов d_t [1].

$$\begin{aligned}f_{t+\tau} &= u_t + b_t \tau \\u_t &= u_{t-1} + b_{t-1} + (1 - \gamma^2) e_t \\e_t &= d_t - f_t \\b_t &= b_{t-1} + (1 - \gamma)^2 e_t\end{aligned}$$

3. Метод Тригг-Лича.

Для стационарных показателей прогноз на любой момент времени по модели адаптивной скорости реакции Тригга и Лича вычисляется по формуле

$$f_{t+\tau} = \tilde{u}_t = |T_t| d_t + (1 - |T_t|) \tilde{u}_{t-1}$$

где $T_t = \bar{e}_t / MAD_t$; $\bar{e}_t = \alpha e_t + (1 - \alpha) \bar{e}_{t-1}$;

$$MAD_t = \alpha |e_t| + (1 - \alpha) MAD_{t-1}$$

где $f_{t+\tau}$ – прогноз на момент времени $t + \tau$; t – текущий момент времени; τ – период упреждения прогноза; d_t – фактическое значение наблюдаемого показателя; e_t – экспоненциально взвешенная ошибка; e_t – ошибка прогноза; T_t – следящий контрольный сигнал; MAD_t – среднее абсолютное отклонение; α – константа сглаживания ($0,1 \leq \alpha \leq 1$)

4. Самоорганизующийся алгоритм прогнозирования.

Процесс построения самоорганизующегося алгоритма (САП) состоит из нескольких шагов.

Шаг 1. Формирование первоначального множества простых алгоритмов прогнозирования.

На этом шаге формируется множество простых алгоритмов, которые будут использоваться при конструировании самоорганизующегося алгоритма. Под определением «простой алгоритм» мы в дальнейшем будем понимать известные алгоритмы (методы, мат. модели) прогнозирования. В первоначальное множество можно включить простые, но в тоже время, научно обоснованные статистические методы прогнозирования.

Шаг 2. Выбор критерия отбора лучшего алгоритма для формирования прогноза.

В качестве критерия отбора можно выбрать относительную или абсолютную ошибку прогнозирования, рассчитываемую в каждой точке или некоторую агрегированную ошибку за предыдущий период.

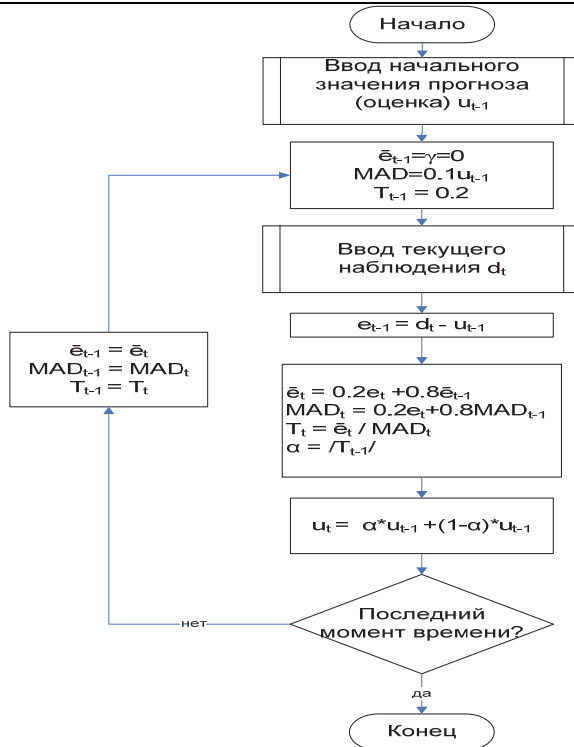


Рис. 4. Блок-схема вычисления прогноза методом Тригга-Лича

Шаг 3. Стадия обучения САП.

Проводится прогнозирование на настоящий момент (или предыдущие моменты) времени, где известно (или известны) фактические значения наблюдаемого показателя по алгоритмам из множества, сформированного на шаге 1.

Шаг 4. Оценивание качества прогноза.

Оценивание качества прогноза проводится с использованием критерия, выбранного на шаге 2. Алгоритм, показывающий минимальное значение критерия (ошибки прогнозирования), будем считать лучшим.

Шаг 5. Прогнозирование на будущий момент времени с использованием САП.

Прогнозное значение наблюдаемого показателя на следующий момент будет определяться с помощью алгоритма, отобранного на шаге 4.

Таким образом, при определении будущего значения наблюдаемого показателя, САП должен переключаться на тот алгоритм, который показал наименьшее значение критерия оценки качества прогнозирования на текущий период [2].

По итогу работы нами была разработана система прогнозирования, которая обеспечивает пользователя возможностью:

Ввода, изменения, просмотра введенных статистических данных;

Выбора метода прогнозирования и получение справочной информации о методе;

Выбора критерия оценки качества прогноза;

Просмотра результатов прогнозирования (прогноз и ошибка на заданный период);

Сохранение прогнозных данных в файле с выбранным форматом.

Система определяет прогноз, ошибку прогноза и накапливает статистические данные наблюдаемого показателя. Она может быть использована для краткосрочного прогноза бизнес-процессов: курса доллара, спроса на промышленные и сельскохозяйственные товары и т.д.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Афанасьев В.Н. Анализ временных рядов и прогнозирование: Учебник/В.Н. Афанасьев, М.М. Юзбашев. – М.: ФиС, ИНФРА-М, 2012. – 320 с.

2. Мюллер Г. Анализ временных рядов и прогнозирование: Учебники / Г. Мюллер и др. – М.: Финансы и статистика, 2012. – 320 с.

3. Лукашин Ю.П. Адаптивные методы краткосрочного прогнозирования временных рядов / Ю.П. Лукашин. – М.: Финансы и статистика, 2003. – 416 с.

4. Шурыгин, А.М. Математические методы прогнозирования / А.М. Шурыгин. – М.: ГЛТ, 2009. – 180 с.

УДК 519.68

Проказникова Е.Н., Карпущина А.А.

ИНФОРМАЦИОННЫЕ СИСТЕМЫ ВЕДЕНИЯ УПРАВЛЕНЧЕСКОГО УЧЕТА

Рязанский государственный радиотехнический университет,
г. Рязань

Для удержания конкурентных позиций на рынке необходимо принимать управленческие решения в режиме «онлайн». Помочь в обработке массива данных позволяет использование информационной системы для ведения управленческого учета.

Рассмотрим информационные системы управленческого учета (далее – ИСУУ).

Классификация ИСУУ:

1. ИСУУ начального уровня;
2. ИСУУ среднего уровня;
3. ИСУУ высшего уровня.

ИСУУ начального уровня используют организации небольшого размера. Отличительной чертой подобных ИСУУ является ограниченный охват бизнес – процессов предприятия. Помимо этого в ИСУУ начального уровня как правило обладают широкими возможностями, что позволяет

интегрировать их с другими системами. Однако, готовые решения подобных информационных систем удовлетворяют узкий круг потребностей организации и могут отличаться по целевому назначению, что нельзя отметить в ИСУУ следующего уровня [2].

ИСУУ среднего уровня включают разного рода подсистемы, например, бухгалтерия и производство. Подобные системы обеспечивают потребности в учете организации одновременно по нескольким направлениям [2].

ИСУУ высшего уровня дает возможность скооперировать базы данных больше двух областей ведения управленческого учета организации, что может привести к реорганизации деятельности компании и усовершенствованию организации работы между структурными подразделениями [3].

Несмотря на разнообразие информационных систем управленческого учета перед решением перехода на внедрение подобной системы возникает вопрос: «Каким образом использование ИСУУ поможет в обработке информации и ускорит процесс принятия управленческих решений в организации?».

В практической деятельности ведения хозяйственной деятельности организации существуют фундаментальные отличия компьютерного и ручного способа управленческого учета информации. Рассмотрим некоторые из них [1].

Первое отличие заключается в количестве регистрации одного и того же показателя. В компьютерной программе достаточно одного раза внесения цифровых значений показателя для того, чтобы они отобразились в разных разделах отчета, в ручном учете необходимо вносить одни и те же значения показателей многократно, поскольку не существует автоматической привязки массива данных между разделами.

Второе отличие гласит о неограниченной аналитичности учета. Компьютерная программа хранит вводимую накопленную информацию в базе данных, что позволяет сотруднику составить аналитический отчет за определенный период времени, при этом структуру отчета можно сформировать в удобном и необходимом виде для пользователя.

Далее можно отметить такое отличие как повышение достоверности учетных данных. Компьютерная программа минимизирует человеческий фактор в арифметическом подсчете цифровых значений.

Таким образом, можно сделать вывод, что разработка, внедрение и использование ИСУУ в организации дает ее пользователям в каждый момент времени контролировать важнейшие финансовые показатели работы организации, а также моделировать финансовые результаты в зависимости от варианта проведения той или иной хозяйственной операции.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Балдин К.В. Информационные системы в экономике. – М.: Дашков и К, 2015. – 395 с.
2. Степанова Е.Е. Информационное обеспечение управленческой деятельности. – М. : ИНФРА-М, 2015. – 192 с.
3. Трофимов В.В. Информационные системы и технологии в экономике и управлении. – М. : Юрайт, 2015. – 522 с.

УДК 365.246

Горюнов Е.А.

СОВРЕМЕННОЕ ИНФОРМАЦИОННО-ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ КАК ОСНОВА УСПЕШНОГО РАЗВИТИЯ РИЭЛТЕРСКОГО БИЗНЕСА

Рязанский государственный радиотехнический университет,
г. Рязань

В работе рассматриваются перспективы применения современного информационно-программного обеспечения для развития риэлтерского бизнеса.

В современном мире сложно найти такое направление бизнеса, которое бы не изменялось и адаптировалось под текущие реалии. В свою очередь не исключением является и риэлтерский бизнес, который функционирует в одной из ведущих сфер экономики – недвижимости [3].

Необходимо отметить, что рынок недвижимости и, в частности, риэлтерский бизнес достаточно интересен для различных смежных видов бизнеса, поскольку всегда существует стремление у крупнейших игроков занять нишу, в которой есть возможность потеснить конкурентов [4].

В свою очередь существует ряд потенциальных инструментов, с помощью которые возможно несколько усиливать активность на рынке недвижимости и помогать людям решать их жилищные вопросы. Один из самых популярных – это снижение ставки ЦБ и, соответственно, ипотечных ставок в коммерческих банках. Поскольку неоспоримым лидером по выданным кредитам в настоящее время является Сбербанк, то выглядит логичным его проникновение в риэлтерский бизнес. Одним из первых решений стало создание платформы Домклик для сбора базы объектов недвижимости, а также предложения клиентам уже от самого Сбербанка, что в перспективе создает возможность создания полноценного агентства недвижимости на основе Сбербанка, причем реализуемого на цифровой основе.

Глобально платформенный принцип в настоящее время приобретает все большее распространение, что вызвано адаптацией технологий и моделей распространения цифровой экономики, включая не только трансформацию производственных процессов и процессов управления на производственных предприятиях [6], но также и трансформацию форматов в торговле [7]. В связи с этим необходимость создания коммуникационных

платформ в риэлтерском бизнесе целесообразна в связи и с необходимостью обработки больших объемов данных, а также возможность создавать коммуникационные площадки для взаимодействия продавцов недвижимости, потенциальных и реальных покупателей, посредников, аналитиков и других участников сделок купли-продажи недвижимости. Следует отметить, что описанная выше активность выхода банков на поле купли-продажи недвижимости не единична, так как наблюдаются еще новые игроки в данном бизнесе в виде различных агрегаторов и платформ для рекламы объектов недвижимости («Авито», «Юла», «Домофонд», «ЦИАН»).

На основании рассмотренных выше предпосылок можно сформировать основные тенденции развития риэлтерского бизнеса:

- во-первых, рынок недвижимости меняется, и все возрастающее количество сделок по ипотеке стимулирует создание и развитие своих ипотечных брокеров непосредственно в офисе;
- во-вторых, активное создание современных сайтов, адаптированных под использование со смартфонов, поскольку объекты недвижимости все чаще выбирают с телефонов и планшетов (рис. 1);
- в-третьих, создание своих электронных площадок между агентствами недвижимости для альтернативы существующих рекламных площадок.

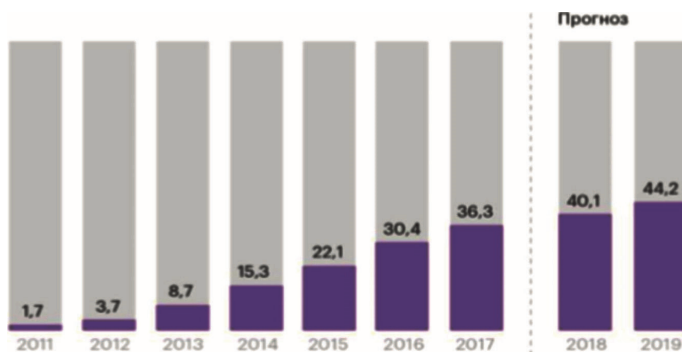


Рис. 1. Доля пользователей, использующих смартфоны для поиска недвижимости, % [2]

Таким образом, агентствам недвижимости, которые хотят продолжить свое развитие и укреплять свою долю на рынке, необходимо принимать вызов активного развития цифровой экономики и IT-технологий в данном бизнесе, что в дальнейшем обеспечит выход на новый качественный уровень.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Критик А.Б Экономика недвижимости: Учебник. – М.:МГУ, 2017.

2. Брюханова Г.А., Яременко Н.Н. Состояние рынка недвижимости на современном этапе // Гуманитарные, социально-экономические и общественные науки, 2016, №1.
3. Прокофьев К.Ю. Рынок недвижимости: понятие, анализ (на примере рынка городской жилой недвижимости) // Экономический анализ: теория и практика, 2017, №3.
4. Стерник Г.М. Методология прогнозирования российского рынка недвижимости // Экономика и финансы, 2017, №8.
5. Ковальчук Ю.А., Степнов И.М. Цифровая экономика: трансформация промышленных предприятий // Инновации в менеджменте, 2017, № 1 (11), с. 32-43.
6. Степнов И.М., Ковальчук Ю.А. Организационные платформы как признак циклических изменений в отрасли (на примере «pop up store») / В сборнике: Управление инновациями – 2017. Материалы международной научно-практической конференции. Под редакцией Р.М. Нижегородцева, Н.П. Горидько. – М., 2017. – с. 99-104.
7. Логинов Е.Л., Райков А.Н. Цифровая экономика: уязвимость к сетевым атакам и возможности обеспечения устойчивости управления // Проблемы рыночной экономики, 2017, №4, с. 4-10.

УДК 519.86

Гутарова Е.А.

АВТОМАТИЗАЦИЯ ОРГАНИЗАЦИИ УЧЕТА ПРОДАЖ ГОТОВОЙ ПРОДУКЦИИ ПО РСБУ И МСФО

**Рязанский государственный радиотехнический университет,
г. Рязань**

В данной работе проведен сравнительный анализ организации учета готовой продукции российской системы бухгалтерского учета (РСБУ) и международных стандартов финансовой отчетности (МСФО) и пути автоматизации учета готовой продукции.

В деятельности любой производственной организации важное место занимают производство и реализация готовой продукции.

Реализация выпущенной продукции является одной из центральных задач для хозяйствующего субъекта, основной деятельностью которого является производство. Это обусловлено тем, что темпы роста объема производства, улучшение качества продукции сильно влияют на величину затрат, прибыли и рентабельности организации, а в основе данных финансовых показателей лежит исчисление себестоимости и учет продаж готовой продукции.

Учетная система, принятая в РФ, отличается от международных стандартов финансовой отчетности. Согласно концепции развития бухгалтерского учета и отчетности в Российской Федерации на среднесрочную перспективу РСБУ в дальнейшем будут все больше сближаться с МСФО [4].

В настоящее время некоторые виды организаций уже законодательно обязаны публиковать консолидированную финансовую отчетность по МСФО [1].

Рассмотрение положений МСФО по учету готовой продукции и сравнение их с РСБУ позволяет наиболее оптимально выработать эффективные способы представления информации о деятельности производственного предприятия в системе бухгалтерского учета и отчетности, а также создать автоматизированную систему учета готовой продукции.

Учет готовой продукции по РСБУ подчиняется Федеральному закону «О бухгалтерском учете», ПБУ 5/01 «Учет материально-производственных запасов», а также методическим указаниям по бухгалтерскому учету материально-производственных запасов, так как в них закреплены основные методы учета готовой продукции [6].

Учет готовой продукции в соответствии с международными стандартами ведется по МСФО (IAS) 2 «Запасы», а признание выручки от реализации готовой продукции происходит согласно МСФО (IAS) 18 «Выручка».

Одно из главных различий МСФО и РСБУ состоит в методах оценки готовой продукции на балансе, так что при составлении отчетности для целей МСФО на основании данных российского бухгалтерского учета может возникнуть необходимость корректировки стоимости остатков готовой продукции на конец периода.

Также отличается влияние права собственности на продукцию для учета ее на балансе. В основе РСБУ лежит данное право, а в основе МСФО – концепция приоритета содержания над формой. Таким образом, РСБУ рассматривает право собственности с юридической точки зрения, а МСФО – с экономической. С юридической точки зрения готовая продукция продана и списывается с баланса продавца тогда, когда к покупателю перешло права собственности на нее, что определяется договором, а с экономической – когда покупатель принимает на себя риск, и получает возможность извлекать экономическую выгоду, а этот момент может не совпадать с моментом перехода права собственности.

При сравнении требований к раскрытию информации в бухгалтерской финансовой отчетности в МСФО 2 и ПБУ 5/01 видно, что информация о готовой продукции в МСФО отражается подробнее, чем в РСБУ, из-за чего требуются дополнительные усилия и трудовые затраты по составлению отчетности от организаций, отчитывающихся по МСФО.

В российской практике МСФО 18 «Выручка» соответствует ПБУ 9/99 «Доходы организации». Требования к определению дохода в них подобны. Критерии признания выручки от продажи продукции близки, но оба подхода не всегда будут давать одинаковое значение в отчетности, так как ранее уже отмечалось, что моменты перехода значительных рисков и перехода юридических прав не всегда одинаковы. В этом случае доходы

от продажи готовой продукции могут попасть в различные периоды отчетности по МСФО и РСБУ.

Автоматизированную информационную систему для учета готовой продукции можно разделить на три блока:

- первый блок данных показывает постоянную информацию, реализованную в виде справочников. В данной системе выделено два основных и три вспомогательных справочника. Справочник Материалы и Справочник Поставщики – это основные, они являются источником информации при учете поступления материалов, а также при формировании отчетной сущности Остатки.

- второй блок – это блок учета движения материалов и готовой продукции. Этот блок состоит из следующих сущностей: Приход, ПриходМЦ, Требование-Накладная, Расход, РасходГП. Для решения задач данного блока используется промежуточная информация, которая в последствии используется для выполнения соответствующих расчетов. В результате обработки данной информации формируются итоговые документы, итоговые ведомости, а также итоговые таблицы, необходимые для формирования итоговых отчетов. Вся эта информация необходима для контроля процесса учета и анализа его результатов.

- третий блок – это блок отчетной информации. С учетом изменения входных данных формируются сводные сущности, в которых все эти изменения учитываются. Это Журнал и Остатки, они необходимы для формирования отчетной информации и удобны тем, что фиксируют каждое изменение данных.

Основное назначение создаваемой АИС – это автоматизация учета складских операций и реализации продукции. Следовательно, структуру программ можно описать следующими основными блоками, представленными на рис. 1:



Рис. 1. Блок-схема основных модулей программы

Рассмотрим архитектуру решения АИС, создаваемой для целей РСБУ и МСФО. Данные первичных источников информации (базы данных для ведения учета РСБУ) с помощью универсального ETL-средства (в нашем случае это внешняя обработка для 1С) трансформируются в единый план

счетов по МСФО и загружаются в консолидационный пакет в виде двух таблиц: сводные остатки и сводные обороты по счетам в разрезе всех аналитик (субконто). Принцип работы прост. Пользователь-ответственный за заполнение пакета на предприятии создает настройку – соответствие между своим планом счетов и планом счетов по МСФО (в соответствие могут входить не только счета, но и субконто). На основании этого соответствия с помощью внешней обработки выгружаются таблицы остатков и оборотов с привязкой к двум планам счетов одновременно: плану счетов РСБУ локальной базы и единому плану счетов консолидационной базы МСФО.

В пакете через формулы на основании этих двух таблиц строятся проверочные отчеты, в т.ч. и пробный баланс по РСБУ в формате счетов МСФО (таблица остатков – для баланса, таблица сводных оборотов – для отчета о доходах и расходах). Данные пакетов (таблицы остатков и оборотов) загружаются в консолидационную систему.

Консолидационная система представляет собой хранилище данных, построенное на платформе 1С Предприятие 8. Система представлена на рис. 2. В предложенной системе есть две центральные таблицы:

- таблица для Данных РСБУ (содержит остатки по счетам РСБУ и сводные обороты, загружается из пакетов);
- таблица данных МСФО (Главная книга МСФО = агрегированные данные РСБУ + Корректировки) с обвязкой справочников, часть из которых являются централизованной аналитикой.

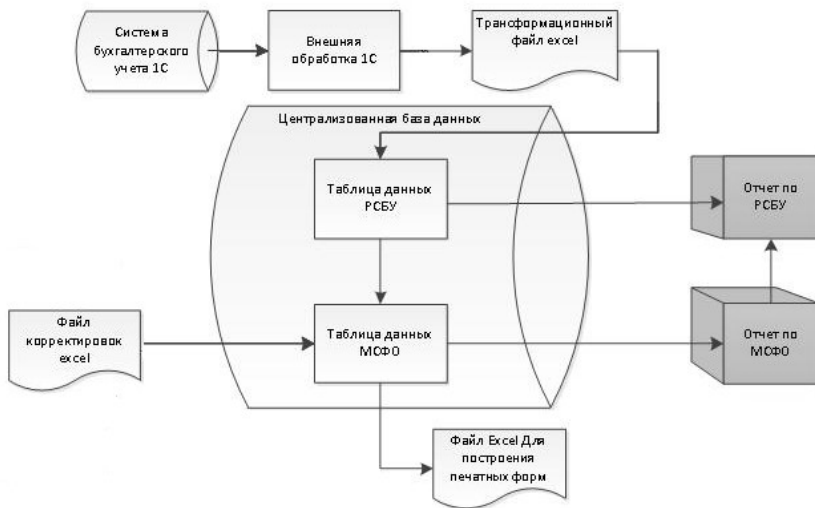


Рис. 2. Архитектура решения АИС

Данные для таблицы данных РСБУ загружаются только из пакетов. Далее эти данные агрегируются и переносятся в сводном виде в таблицу данных по МСФО. Происходит это при загрузке из пакета автоматически. В таблице данных по МСФО также имеется возможность добавлять строки-корректировки. Каждая корректировка привязывается к файлу excel – источнику расчетов. Таким образом, итоговый баланс по МСФО может быть разложен по счетам на строки корректировки с привязкой к файлу-расчету и строки, загруженные из пакета. Первичные данные не изменяются, и имеется возможность проследить всю цепочку расчетов от конечного результата до первоначального источника данных.

Используется иерархическая англо-саксонская модель плана счетов по МСФО (без корреспонденции). Счета отчета о Доходах и расходах подчиняются счету Нераспределенной прибыли. Таким образом, обеспечивается связь между балансом и отчетом о доходах и расходах. План счетов вы можете разработать для себя сами или взять любой рекомендованный. Баланс и отчет о доходах и расходах получается с помощью инструмента сводной таблицы на основании таблицы данных МСФО.

Хранилище данных имеет два отчета: Отчет по данным МСФО и Отчет по Данным РСБУ. Отчеты построены на компоновке данных 1С и по своей сути представляют собой два OLAPкуба: куб для данных МСФО и куб для данных РСБУ. Строятся эти отчеты очень быстро, т.к. по сути используется платформенное средство 1С – компоновка данных. Таким образом, отчет позволяет получать расшифровку данных МСФО до первичной аналитики. Изучив принцип работы сводных таблиц, пользователь сам сможет строить для себя необходимые отчеты.

В последние годы российские стандарты по учету готовой продукции согласно концепции развития бухгалтерского учета и отчетности в Российской Федерации на среднесрочную перспективу начали сближаться с международными принципами учета. Однако часть положений РСБУ по-прежнему имеют существенные различия с МСФО, что требует дополнительных уточнений, разработки положений, внедрения программ и алгоритмов по параллельному ведению учета для организаций. В настоящее время проводится разработка новых федеральных стандартов РФ, которые должны будут соответствовать нормам МСФО. Внедрение нового федерального стандарта по учету запасов назначено на 2019 год.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Федеральный закон «Об акционерных обществах» от 26.12.1995 N 208-ФЗ (ред. от 31.12.2017) [Электронный ресурс]: – Режим доступа: КонсультантПлюс.

2. Об утверждении положения по бухгалтерскому учету «Учет материально-производственных запасов» (ПБУ 5/2001) [Электронный ресурс]: –

Приказ Минфина России от 09.06.2001 N 44н (ред. от 16.05.2016). – Режим доступа: КонсультантПлюс.

3. Об утверждении положения по бухгалтерскому учету «Доходы организации» (ПБУ 9/99). [Электронный ресурс]: Приказ Минфина РФ от 06.05.1999 N 32н (ред. от 06.04.2015). – Режим доступа: КонсультантПлюс.

4. Об одобрении Концепции развития бухгалтерского учета и отчетности в Российской Федерации на среднесрочную перспективу [Электронный ресурс]: Приказ Минфина РФ от 01.07.2004 N 180 – Режим доступа: КонсультантПлюс.

5. Международный стандарт финансовой отчетности (IAS) 2 «Запасы» [Электронный ресурс]: Приказ Минфина России от 25.11.2011 N 160н. (ред. от 27.06.2016) – Режим доступа: КонсультантПлюс.

6. Международный стандарт финансовой отчетности (IAS) 18 «Выручка» [Электронный ресурс]: Приказ Минфина России от 28.12.2015 N 217н) (ред. от 27.06.2016) – Режим доступа: КонсультантПлюс.

УДК 519.688

Косилова Е.Р.

АВТОМАТИЗАЦИЯ РАСЧЕТА ПОТРЕБНОСТИ В ХИМИЧЕСКИХ МАТЕРИАЛАХ НА КОЖЕВЕННОМ ПРОИЗВОДСТВЕ ПРИ ПОМОЩИ VASIAL BASIC FOR APPLICATION

**Рязанский государственный радиотехнический университет,
г. Рязань**

Автоматизация аналитических расчетов стала объективной необходимостью в связи с трудоемкостью разнообразных экономических вычислений.

Современные информационные технологии позволяют целиком автоматизировать обработку всех экономических данных, в том числе и по анализу хозяйственной деятельности. Роль автоматизации аналитических расчетов заключается в следующем:

1. Повышается продуктивность работы экономистов-аналитиков. Они освобождаются от технической работы и больше занимаются творческой деятельностью, что позволяет делать более глубокие исследования, вести постановку более сложных экономических задач.

2. Более глубоко и всесторонне исследуются экономические явления и процессы, более полно изучаются факторы и выявляются резервы повышения эффективности производства,

3. Повышаются оперативность и качество анализа его общий уровень и действенность.

В целом на производственных предприятиях планирование потребности в сырье и материалах происходит следующим образом: исходными данными для расчета потребности производства в материальных ресурсах являются предусматриваемые объемы выпуска продукции (или оказываемые

мых услуг), а также нормативная база потребностей в материальных ресурсах на единицу продукции (норма расхода). Потребность в материальных ресурсах определяется методом прямых расчетов, т.е. путем умножения нормы расхода материалов на соответствующие объемные показатели (1):

$$ПРi = Нрi * N, \quad (1)$$

где $ПРi$ – потребность в i -ом материале за определенный промежуток времени, $Нрi$ – норма расхода i -ого материала на единицу производимой продукции, N – объем производства продукции за данный интервал времени.

Кожевенное производство отличается своей специфичностью в области потребляемых производственных материалов. Для производства кожи основным материалом являются химические материалы. Кроме того, номенклатура потребляемых химических материалов достаточно обширна, каждый химический материал потребляется на определенной стадии производственного процесса в определенном количестве, поэтому точное и своевременное планирование и учет количества химических материалов на складах и необходимого количества для всего производственного объема.

Исходной информацией для расчета необходимого количества химических материалов в разрезе каждого вида химматериала является: объем производства в плановый период и, соответственно, нормы расхода; остатки химматериалов в разрезе момент расчета в пути.

Введем обозначения: X – объем остатков химических материалов (кг.) на складе на $xx.xx.xxxx$ г.; Y – объем химических материалов (кг.), находящихся в пути, которые поступят на завод к началу планируемой недели; $Нрi$ – норма расхода i -ого химического материала (кг./ед.); N – объем производства на планируемый интервал времени (ед.)

Построим блок-схему расчета потребности в химических материалах для определенного количества производимой продукции на плановый период – неделя (показана на рис. 1)

Так как кожевенное производство характеризуется своей непрерывностью, то учет и заказ химматериалов в производство должно осуществляться в оперативном режиме, и согласно вышеописанному алгоритму данные расчеты будут осуществляться с периодичностью в 3 дня, поэтому на оптимальным вариантом для автоматизации расчетов будет написание кода в VBA в одном из приложений MS Office – Exel. Кроме того, данные расчеты на предприятии осуществляет экономист, а VBA изначально был ориентирован на пользователей, а не на профессиональных программистов, поэтому создавать программы на нем можно достаточно быстро и легко, также в Office встроены мощные средства, облегчающие работу пользователя: подсказки по объектам и по синтаксису, макрорекордер и прочее.

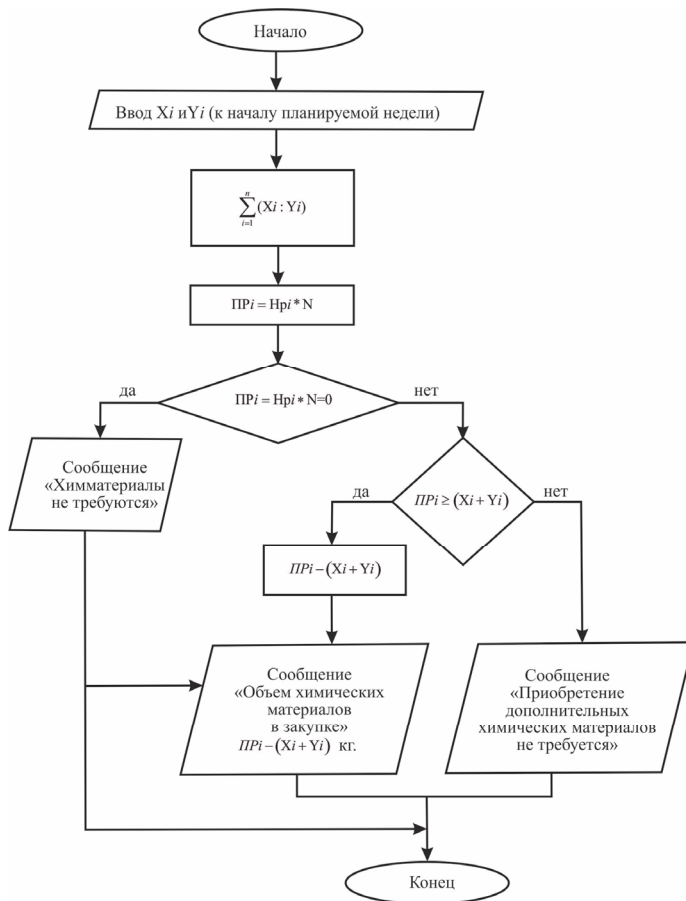


Рис. 1. Алгоритм расчета потребности в химических материалах

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Комолова Н.В., Яковлева Е.С. Программирование на VBA в Excel. – М.: Самоучитель, 2016. – 432 с.
2. Гордина С.В. VISUAL BASIC FOR APPLICATION как система объектноориентированного программирования // Интеграция образования. – 2006. – № 4. – С.60-65.
3. Костин Л.Ю. Использование возможностей EXEL для решения задач, создания наглядных пособий и разработки шаблонов для урока математики // Развитие математических теорий и методов для компьютерных приложений. – 2013. – № 27. – С.200-202.

4. Лещукова И.В. Оперативное планирование производства // Инновационная наука. – 2016. – № 5. – С.128-129.
5. Шерстяных Н.С. Эффективное использование материальных ресурсов // Экономинфо. – 2010. – № 14. – С.55-57.
6. Блок-схемы алгоритмов [Электронный ресурс] <https://pro-prof.com/archives/1462>.
7. Кожевенное производство [Электронный ресурс] // Режим доступа: <https://www.ronl.ru/stati/marketing/135769/>.
8. Учет материалов на производстве [Электронный ресурс] // Режим доступа: <https://leksii.org/12-87597.html>.

СОДЕРЖАНИЕ

Овечкин Г.В., Овечкин П.В., Чан Т.З.

ВЫСОКОСКОРОСТНОЙ МНОГОПороговый ДЕКодЕР
ДЛЯ ВОССТАНОВЛЕНИЯ СТИРАНИЙ
НА БАЗЕ СОВРЕМЕННЫХ ГРАФИЧЕСКИХ ПРОЦЕССОРОВ4

Бубнов А.А., Бубнов С.А.

О ВОЗМОЖНОСТИ ИСПОЛЬЗОВАНИЯ ДСМ-МЕТОДА
ДЛЯ ОЦЕНКИ НАДЁЖНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
НА ЭТАПЕ ТЕСТИРОВАНИЯ9

Кабанов А.Н., Матаев К.Р., Фоломкин Д.Н.

ФОРМИРОВАНИЕ ГЛАВНЫХ КОМПОНЕНТ
ПРИ МНОГОКРИТЕРИАЛЬНОМ АНАЛИЗЕ СИСТЕМЫ12

Лощинин А.Б.

ИНФОРМАЦИОННЫЕ МЕДИЦИНСКИЕ СИСТЕМЫ15

Горкин А.В.

РАБОТА С МАТЕМАТИЧЕСКИМИ ВЫРАЖЕНИЯМИ
В РАЗРАБОТКЕ ПОД ANDROID (JAVA)19

Гринченко Н.Н., Корябкин В.В.

ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ИСПРАВЛЕНИЯ СТИРАНИЙ
С ПОМОЩЬЮ КАСКАДНЫХ СХЕМ КОРРЕКЦИИ ОШИБОК22

Кузин И.Н.

ИССЛЕДОВАНИЕ ЯВЛЕНИЯ «ПЕРЕСТАНОВКИ РАНГОВ»
В МЕТОДЕ АНАЛИЗА ИЕРАРХИЙ26

Власов П.А.

КОМПЬЮТЕРНЫЕ ВИРУСЫ И ИХ КЛАССИФИКАЦИЯ28

Матаев К.Р., Кабанов А.Н.

ТЕХНОЛОГИЯ ПРОГНОЗИРОВАНИЯ
МНОГОМЕРНЫХ ВРЕМЕННЫХ РЯДОВ30

Азязова Е.А., Коровин А.А.

КРИПТОСТОЙКОСТЬ ШИФРОВ32

Власов П.А., Гагаринский В.Ю.

ОДНОМЕРНАЯ ДИНАМИКА.
ОПРЕДЕЛЕНИЕ ОПТИМАЛЬНОГО РЕШЕНИЯ.
ЗАДАЧА О КУЗНЕЧИКЕ, КОТОРЫЙ СОБИРАЕТ МОНЕТЫ34

Магомадов Р.У., Коровин А.А.

АЛГОРИТМ ОТРИЦАТЕЛЬНОГО ОТБОРА
В РАМКАХ ЗАЩИТЫ ИНФОРМАЦИИ37

Гагаринский В.Ю.

ОДНОМЕРНАЯ ДИНАМИКА. ПОДСЧЕТ КОЛИЧЕСТВА ВАРИАНТОВ РЕШЕНИЯ	39
--	----

Сальников А.П., Магомадов Р.У.

ИМИТОСТОЙКОСТЬ ШИФРА	40
----------------------------	----

Чернышов А.С.

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СИСТЕМЫ КРЕДИТНОГО БРОКЕРА	42
---	----

Васильев И. В.

РАЗРАБОТКА КОНФИГУРАЦИИ 1С ДЛЯ САНАТОРИЯ	44
--	----

Беляев Д.Ю., Дмитриева Т.А.

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ РЕШЕНИЯ ПРОБЛЕМЫ РУЧНОГО СОСТАВЛЕНИЯ ОТЧЕТНЫХ ДОКУМЕНТОВ.....	46
---	----

Ермаков Д.С.

АЛГОРИТМЫ КОДИРОВАНИЯ И ДЕКОДИРОВАНИЯ ПО ХЭММИНГУ	48
--	----

Дмитриева Т.А., Палагин Д.А.

РАЗРАБОТКА ИНТЕРАКТИВНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ОБУЧЕНИЯ АЛГОРИТМИЧЕСКОМУ ЯЗЫКУ	50
---	----

Щелконогов А.Н.

РАЗРАБОТКА ПРОСТЕЙШИХ НЕЙРОСЕТЕЙ В KERAS.....	51
---	----

Дмитриева Т.А., Колунтаева Е.С.

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ИМПОРТА ДАННЫХ О СОТРУДНИКАХ ИЗ EXCEL В 1С.....	53
--	----

Перевезенцев Е.Е., Ромашкова В.В.

ГИБРИДНЫЕ ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ.....	54
---	----

Титов Д.Д.

АЛГОРИТМЫ ВЫЧИСЛЕНИЯ ЦЕНТОРИДОВ ИНТЕРВАЛЬНЫХ НЕЧЕТКИХ МНОЖЕСТВ ВТОРОГО ТИПА	58
--	----

Скачков А.М.

АНАЛИЗ ПРОБЛЕМЫ ПОДБОРА ЗНАЧЕНИЙ ГИПЕРПАРАМЕТРОВ В НЕЙРОННЫХ СЕТЯХ НА ПРИМЕРЕ БИБЛИОТЕКИ KERAS	66
---	----

Балабанов Н.Р.

ПРОБЛЕМЫ ЗАЩИТЫ ИНФОРМАЦИОННЫХ СИСТЕМ ОТ ЧЕЛОВЕЧЕСКОГО ФАКТОРА	68
---	----

Чернобродова Л.А., Кабанов А.Н., Моргунов Д.Р.

РАЗРАБОТКА ИНФОРМАЦИОННОГО ОБЕСПЕЧЕНИЯ
МНОГОПАРАМЕТРИЧЕСКОЙ ОПТИМИЗАЦИИ УПРАВЛЕНИЯ
ИНВЕСТИЦИОННОЙ ДЕЯТЕЛЬНОСТИ ОРГАНИЗАЦИИ
НА ОСНОВЕ ЭКСПЕРТНОГО АНАЛИЗА70

Чернобродова Л.А., Кабанов А.Н., Иешкин А.Ю.

ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ МНОГОКРИТЕРИАЛЬНОЙ
АДАПТИВНОЙ ОПТИМИЗАЦИИ В ИНВЕСТИЦИОННОЙ
ПОЛИТИКЕ ПРЕДПРИЯТИЙ74

Романов А.И. Медведев Р.Е.

ИСПОЛЬЗОВАНИЕ ДИСТАНЦИОННЫХ ОБРАЗОВАТЕЛЬНЫХ
ТЕХНОЛОГИЙ ПРИ ОРГАНИЗАЦИИ ПОЫШЕНИЯ КВАЛИФИКАЦИИ
СОТРУДНИКОВ ПРОИЗВОДСТВЕННЫХ ПРЕДПРИЯТИЙ78

Сидорова М.А.

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ BLOCKCHAIN
ПРИ СОВЕРШЕНСТВОВАНИИ ВЗАИМОДЕЙСТВИЯ
ПРЕДПРИЯТИЙ АВИАЦИОННОЙ ПРОМЫШЛЕННОСТИ80

Воронина О.А. Медведев Р.Е.

СТРУКТУРА АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ
СИСТЕМ ФИНАНСОВО-ЭКОНОМИЧЕСКОГО АНАЛИЗА82

Рябых А.Н.

ЦИФРОВЫЕ ПЛАТФОРМЫ КОММУНИКАЦИЙ
ДЛЯ ФОРМИРОВАНИЯ ПРОДАЖ85

Минашкина А.Н. Медведев Р.Е.

КЛЮЧЕВЫЕ КРИТЕРИИ ВЫБОРА АВТОМАТИЗИРОВАННЫХ
ИНФОРМАЦИОННЫХ СИСТЕМ БУХГАЛТЕРСКОГО УЧЕТА
АВТОНОМНЫХ УЧРЕЖДЕНИЙ87

Крылова М.А., Медведев Р.Е.

АНАЛИЗ СОВРЕМЕННЫХ АВТОМАТИЗИРОВАННЫХ СИСТЕМ
УПРАВЛЕНИЯ ПЕРСОНАЛОМ ИСПОЛЬЗУЮЩИХ МЕТОДИКУ
ОЦЕНКИ ПЕРСОНАЛА ПО КРІ90

Блохина И.О.

АВТОМАТИЗАЦИЯ БИЗНЕС-ПРОЦЕССОВ НА ПРЕДПРИЯТИИ94

Коротаев А.Н., Макашкин Я.О., Благодарова Т.А.

ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ЛЕТНОЙ ЭКСПЛУАТАЦИИ
ВОЗДУШНЫХ СУДОВ ГРАЖДАНСКОЙ АВИАЦИИ96

Байкин П.С.

РАЗРАБОТКА СИСТЕМЫ КРАТКОСРОЧНОГО ПРОГНОЗИРОВАНИЯ
СПРОСА НА ПРОДУКЦИЮ С ИСПОЛЬЗОВАНИЕМ
ПРИНЦИПА САМООРГАНИЗАЦИИ98

Проказникова Е.Н., Карпухина А.А.

ИНФОРМАЦИОННЫЕ СИСТЕМЫ
ВЕДЕНИЯ УПРАВЛЕНЧЕСКОГО УЧЕТА 103

Горюнов Е.А.

СОВРЕМЕННОЕ ИНФОРМАЦИОННО-ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
КАК ОСНОВА УСПЕШНОГО РАЗВИТИЯ РИЭЛТЕРСКОГО БИЗНЕСА 105

Гутарова Е.А.

АВТОМАТИЗАЦИЯ ОРГАНИЗАЦИИ УЧЕТА ПРОДАЖ
ГОТОВОЙ ПРОДУКЦИИ ПО РСБУ И МСФО 107

Косилова Е.Р.

АВТОМАТИЗАЦИЯ РАСЧЕТА ПОТРЕБНОСТИ
В ХИМИЧЕСКИХ МАТЕРИАЛАХ НА КОЖЕВЕННОМ ПРОИЗВОДСТВЕ
ПРИ ПОМОЩИ VASIAL BASIC FOR APPLICATION 112

МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

межвузовский сборник научных трудов

Подписано в печать 05.02.2019 г.
Формат 60х84/16. Гарнитура «Times New Roman».
Бумага офсетная. Печ. л. 7,5.
Тираж 80 экз. Заказ № 1992

Издательство ИП Коняхин А.В. (Book Jet)

Отпечатано в типографии Book Jet
390005, г. Рязань, ул. Пушкина, д.18
Сайт: <http://bookjet.ru>
Почта: info@bookjet.ru
Тел.: +7(4912) 466-151

ISBN 978-5-6042175-7-3

