

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

СОВРЕМЕННЫЕ ПРОБЛЕМЫ МАТЕМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

**Материалы VI республиканской научной конференции
молодых ученых и студентов
26–28 ноября 2009 года**

Часть I

Брест 2009

УДК 517+519+004
ББК 22.11/.19+32.97

Редакционная коллегия:

В.С. Рубанов, БрГТУ, к.ф.-м.н., доцент – главный редактор
В.А. Головко, БрГТУ, д.т.н., профессор – зам. главного редактора
Р.Х. Садыхов, БГУИР, д.т.н., профессор
Н.В. Лазакович, БГУ, д.ф.-м.н., профессор
А.И. Калинин, БГУ, д.ф.-м.н., профессор
А.А. Дудкин, ОИПИ НАНБ, к.т.н., с.н.с.
В.Ф. Савчук, БрГУ, к.ф.-м.н., доцент
В.М. Ракецкий, БрГТУ, к.ф.-м.н., доцент
С.С. Дереченник, БрГТУ, к.т.н., доцент
Л.П. Махнист, БрГТУ, к.т.н., доцент
С.И. Парфомук, БрГТУ, к.т.н., доцент
Ю.В. Савицкий, БрГТУ, к.т.н., доцент
Д.А. Костюк, БрГТУ, к.т.н., доцент

Рецензенты:

кафедра информатики и прикладной математики БрГУ им. А.С. Пушкина;
зав. кафедрой алгебры и геометрии БрГУ им. А.С. Пушкина **Матысик О.В.**, к.ф.-м.н., доцент.

Современные проблемы математики и вычислительной техники: материалы VI Республиканской научной конференции молодых ученых и студентов, Брест, 26–28 ноября 2009 г. – Брест: БрГТУ, 2009. – Ч. 1. – 120 с.

ISBN 978-985-493-132-6 (Ч. 1)
ISBN 978-985-493-131-9

Представлены тезисы докладов по современным проблемам информационных технологий в научных и прикладных исследованиях, моделирования нелинейных динамических процессов, распознавания образов и анализа изображений, искусственного интеллекта и нейронных сетей, аналитических и численных методов исследований в математике и их приложениях.

Издается в 2-х частях. Часть 1.

УДК 517+519+004
ББК 22.11/.19+32.97

ISBN 978-985-493-132-6

Учреждения образования
© «Брестский государственный технический университет», 2009

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И НЕЙРОННЫЕ СЕТИ

УДК 004.056.57:032.26

АЛГОРИТМЫ ИСКУССТВЕННЫХ ИММУННЫХ СИСТЕМ И НЕЙРОННЫХ СЕТЕЙ ДЛЯ ОБНАРУЖЕНИЯ ВРЕДНОСНЫХ ПРОГРАММ

Безобразов С.В., Рубанов В.С.

УО «Брестский государственный технический университет», г. Брест

Введение. Современные антивирусные продукты представляют собой сложные программные модули, тесно интегрированные в ядро операционной системы и работающие с ней как одно целое. Это не только сканеры, выполняющие простой поиск вирусов по сигнатуре, но и эвристические анализаторы, сетевые экраны, ревизоры и др. Несмотря на это, на сегодняшний день они проигрывают борьбу создателям вредоносных программ. Киберпреступники постоянно придумывают и реализовывают новые пути и методы заражения компьютеров, разрабатывают хитроумные алгоритмы и внедряют свои вредоносные программы.

С момента появления нового вируса до появления ответной реакции на этот вирус со стороны антивирусной индустрии может проходить какое-то, иногда продолжительное время. Как показала практика, за это время вирусы способны заразить сотни тысяч компьютеров, вызвать настоящую вирусную эпидемию и принести огромные убытки. Современные исследования в области защиты информации направлены на создание таких методов и алгоритмов защиты, которые были бы способны обнаружить и нейтрализовать неизвестные вирусы.

Методы искусственных иммунных систем (ИИС) [1] позволяют сконструировать такую систему обнаружения вирусов, которая способна обнаруживать неизвестные вредоносные программы. Основными элементами ИИС, которые выполняют функцию обнаружения вирусов, являются иммунные детекторы. Нами был предложен алгоритм формирования иммунных детекторов на основе искусственных нейронных сетей.

Метод формирования детекторов на основе нейронной сети. Рассмотрим процесс формирования иммунных детекторов на основе нейронных сетей. Вначале генерируется начальная популяция иммунных детекторов, каждый из которых представляет собой искусственную нейронную сеть [2]. Затем формируется набор чистых файлов, состоящий, как правило, из утилит операционной системы, различных документов, файлов разнообразного программного обеспечения. На следующем шаге выбирается несколько компьютерных вирусов или их сигнатур. Набор из чистых файлов и вирусов образует обучающую выборку для нейросетевых детекторов. В процессе обучения нейронная сеть учится распознавать вредоносные программы от «чистого» ПО. Механизм обучения детектора представлен на рисунке 1.

Набор обученных нейронных сетей образует популяцию иммунных детекторов, которые циркулируют в компьютерной системе и производят обнаружение компьютерных вирусов. Наличие разнообразных файлов для обучения и элемента случайности в формировании входных векторов дает возможность получить большое количество различных по своей структуре иммунных детекторов. В процессе сканирования неизвестного файла нейрон-

ная сеть идентифицирует неизвестный образ, в результате чего иммунный детектор принимает решение о принадлежности файла к классу вредоносных программ или к классу чистых файлов.

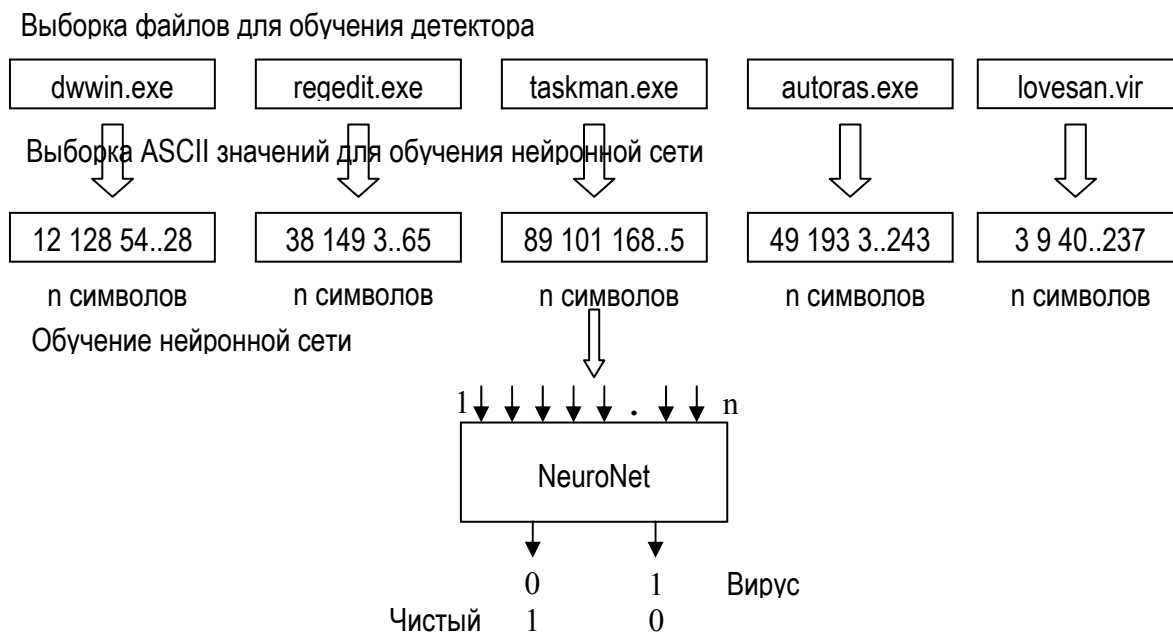


Рисунок 1 – Механизм обучения иммунного детектора на основе нейронной сети

Общий алгоритм функционирования нейросетевых иммунных детекторов можно представить в виде следующих шагов:

1. Генерация начальной популяции иммунных детекторов.
2. Обучение сформированных иммунных нейросетевых детекторов.
3. Отбор (селекция) нейросетевых иммунных детекторов.
4. Функционирование детекторов в компьютерной системе.
5. Уничтожение нейросетевых детекторов по истечению времени.
6. Обнаружение вредоносной программы.
7. Клонирование и мутация нейросетевых иммунных детекторов.
8. Формирование детекторов иммунной памяти.

Отметим основные отличия предложенного алгоритма от существующих. В данном случае каждый нейросетевой иммунный детектор является полностью самостоятельным объектом, т.е. сам выбирает себе область сканирования. Для этого он получает список файлов, хранящихся на жестком диске, и случайным образом выбирает файл из списка для его проверки. После проверки одного файла, детектор переходит к следующему файлу, также выбранному случайным образом из существующего списка. Сканирование файлов нейросетевым иммунным детектором продолжается до тех пор, пока детектор не обнаруживает вредоносную программу, либо до истечения времени, отведенного для функционирования данного детектора.

Популяция нейросетевых иммунных детекторов обеспечивает достаточную область покрытия сканирования файлов на жестком диске для своевременного обнаружения проникшего потенциального компьютерного вируса.

Таким образом, соблюдается принцип децентрализации системы безопасности, построенной на основе комбинации методов нейронных сетей и искусственных иммунных систем, что значительно повышает отказоустойчивость и защищенность системы в целом.

Структура нейросетевого иммунного детектора. На рисунке 2 изображена архитектура нейросетевого иммунного детектора, который состоит из трех слоев нейронных элементов и арбитра.

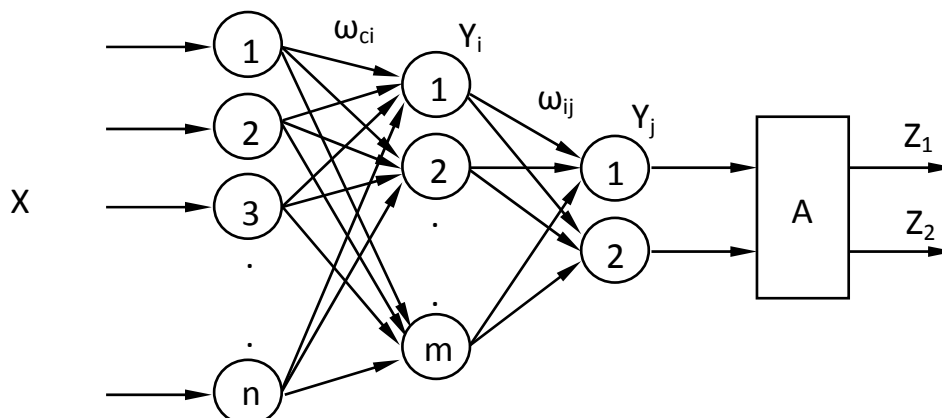


Рисунок 2 – Нейросетевой иммунный детектор

На вход такого детектора в режиме функционирования подаются фрагменты проверяемого файла, которые формируются в соответствии с методом скользящего окна. Первый слой нейронных элементов является распределительным. Он распределяет входные сигналы на нейронные элементы второго (скрытого) слоя. Количество нейронных элементов распределительного слоя равняется размерности скользящего окна.

Второй слой состоит из нейронов Кохонена, которые используют конкурентный принцип обучения и функционирования в соответствии с правилом «победитель берет все».

Третий слой состоит из двух линейных нейронных элементов, которые используют линейную функцию активации. Арбитр осуществляет процедуру окончательного решения о принадлежности сканируемого файла к вирусному или чистому классу.

Результаты экспериментов. На рисунке 3 приведены результаты экспериментальной и теоретической вероятности обнаружения вредоносной программы в зависимости от количества детекторов.

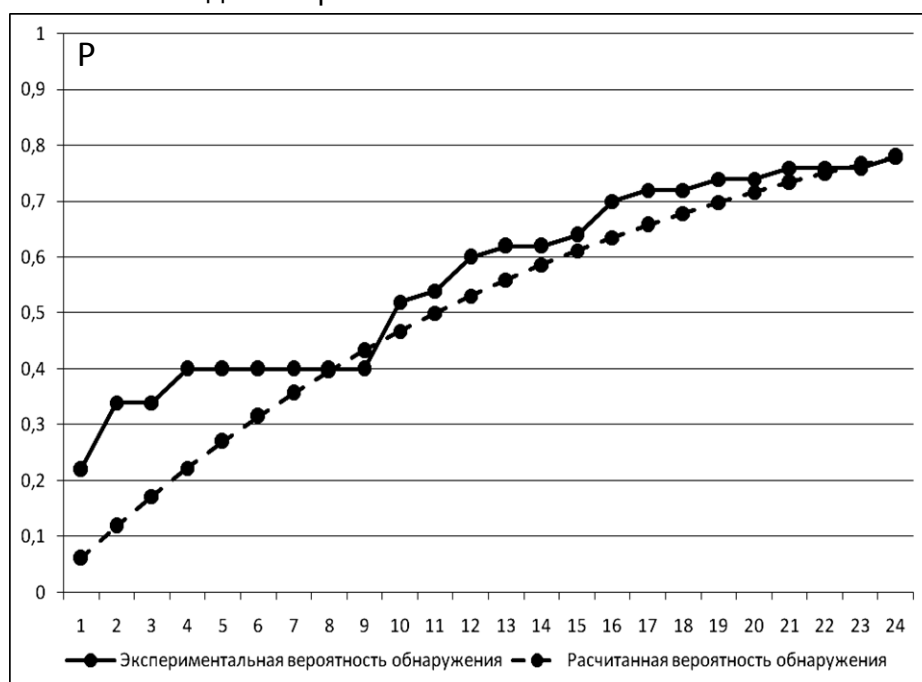


Рисунок 3 – Вероятности обнаружения вредоносных программ нейросетевыми иммунными детекторами

Как следует из рисунка, теоретическая вероятность хорошо аппроксимирует экспериментальную вероятность обнаружения вредоносных программ.

Выводы. Разработан алгоритм построения и функционирования нейросетевой искусственной иммунной системы для обнаружения вредоносных программ, который характеризуется непрерывной эволюцией нейросетевых иммунных детекторов с целью эффективного обнаружения вредоносных программ. Предложенный алгоритм отличается от известных способом клональной селекции, когда мутация детекторов происходит в результате их дополнительного обучения, а отбор клонированных детекторов происходит в соответствии с их значениями суммарной квадратичной ошибки. Это позволяет адаптироваться нейросетевым иммунным детекторам к обнаружению вредоносных программ.

Разработана структура нейросетевого иммунного детектора для обнаружения вредоносных программ, которая состоит из трех слоев нейронных элементов и арбитра. Она характеризуется малым объемом обучающей выборки. Предложенный нейросетевой иммунный детектор способен обнаруживать неизвестные вирусы.

Литература

1. Искусственные иммунные системы для защиты информации: применение LVQ сети // IX Всероссийская научно-техническая конференция «Нейроинформатика - 2007»: сборник научных трудов. – М.: МИФИ, 2007. – Ч. 2.

2. Головкин, В.А. Нейронные сети: обучение, организация, применение / В.А. Головкин // Нейрокомпьютеры и их применение : учеб. пособие / В.А. Головкин. – М., 2001 – 256 с.

УДК 004.8.032.26

МУЛЬТИАГЕНТНЫЙ ПОДХОД К ОБЕСПЕЧЕНИЮ БЕЗОПАСНОСТИ КОМПЬЮТЕРНЫХ СЕТЕЙ

Войцехович Л.Ю.

УО «Брестский государственный технический университет», г. Брест

Высочайший уровень угроз информационной безопасности из внешней среды сделал брандмауэр и *Систему Обнаружения Вторжений (Intrusion Detection System - IDS)* необходимой составляющей защищенной информационной системы. В современном мире развивающихся стремительными темпами компьютерных технологий и телекоммуникаций злоумышленникам стало гораздо легче достичь поставленных целей, благодаря невнимательности и неосведомленности своих жертв о существующих методах защиты.

Простейшим средством сетевой защиты может служить брандмауэр (межсетевой экран, firewall) - реализованное программно или аппаратно средство фильтрации сетевого трафика между двумя сетями или компьютером и сетью (персональный брандмауэр). При этом используются сетевые адреса отправителя и получателя запроса или конкретные службы, а анализа передаваемого трафика не происходит.

Для выполнения анализа передаваемых в сети данных необходимо более сложное и интеллектуальное средство – Система Обнаружения Вторжений [1]. Система обнаружения вторжений – программное и/или аппаратное средство для выявления фактов несанкционированной деятельности (вторжения или сетевой атаки) в компьютерной сети или отдельном узле.

В этой работе для построения системы обнаружения вторжений предлагается использовать *Мультиагентную нейронную сеть* на базе совмещения механизмов *Искусственной иммунной системы* и *Искусственных нейронных сетей*. Предполагается, что такая система обнаружения атак будет способна выполнять обнаружение злоупотреблений и обнаружение аномалий в режиме реального времени.

1. ИММУННАЯ СИСТЕМА

Перед тем как приступить непосредственно к рассмотрению искусственной иммунной системы для построения системы обнаружения атак вкратце остановимся на работе иммунной системы человека. Это описание будет поверхностно, поскольку нас интересуют лишь те механизмы, которые можно использовать в нашей предметной области.

Если так можно выразиться, то основным принципом работы иммунной системы человека является сравнение отдельных “образов” (шаблонов) с телами внутри организма человека. Таким образом, можно обнаружить инородные тела, которые называют антигенами.

В реальной жизни роль вышеупомянутых “шаблонов” выполняют лимфоциты. Они постоянно генерируются спинным мозгом и тимусом в соответствии с информацией, содержащейся в ДНК (эта информация накапливается, и такой процесс называется эволюцией геномной библиотеки). Лимфоциты распространяются в организме через лимфатические узлы. Каждый тип лимфоцитов способен распознать некоторое ограниченное число антигенов. В процессе создания лимфоцитов имеется важный этап – негативная селекция. На этом этапе выполняется специальная процедура проверки на совместимость с родными клетками организма. Если лимфоцит несовместим, то он уничтожается. Иначе он будет бороться с клетками своего же организма. Таким образом, благодаря негативной селекции, “шаблоны” содержат информацию, которая отсутствует внутри организма. Если некоторое внешнее тело соответствует определенному “шаблону”, то оно воспринимается как инородное и должно быть немедленно уничтожено.

В случае если лимфоциты обнаруживают антиген, то на базе соответствующего шаблона создаются новые антитела, которые и уничтожают антиген. Существует также другой важный механизм – клональная селекция. Этот механизм подобен естественному отбору: выживают только те антитела, которые в наибольшей степени соответствуют обнаруженному антигену. Таким образом, данные о сформированных антителах попадают в так называемую иммунную память.

Одна из наиболее подходящих областей применения механизмов иммунных систем – это компьютерная безопасность, где аналогия между защитой человеческого тела и защитой нормально функционирующей компьютерной системы очевидна.

Эксперты, работающие в области искусственных иммунных систем, отмечают три основных свойства таких систем:

- во-первых, они распределенные;
- во-вторых, это самоорганизующиеся системы;
- в-третьих, такие системы не особенно требовательны к вычислительным ресурсам.

По мнению большинства экспертов, эффективная система обнаружения вторжений должна обладать всеми вышеперечисленными свойствами.

2. НЕЙРОСЕТЕВОЙ ДЕТЕКТОР

В рассматриваемой мультиагентной системе обнаружения атак нейросетевой детектор выполняет функции лимфоцита в иммунной системе человека. *Нейронные сети* обладают хорошими обобщающими способностями, могут эффективно решать задачи аппроксимации, классификации и обработки зашумленных данных, что особенно важно в такой области, как обнаружение вторжений.

В данной работе в качестве основного агента системы обнаружения атак (см. рисунок 1) предлагается использовать нейронную сеть, представляющую собой объединение Рециркуляционной нейронной сети (RNN) и Многослойного персептрона (MLP).

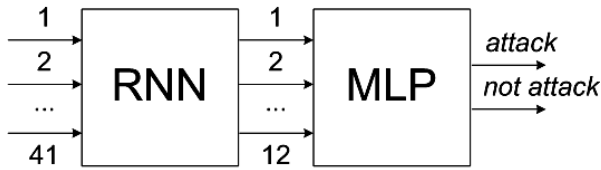


Рисунок 1 – Детектор для мультиагентной нейронной сети

На вход подается 41 параметр, определенный в базе KDD-99 [2]. Эта база содержит информацию о множестве соединений в компьютерной сети. RNN, применение которой с линейной функцией аналогично использованию метода главных компонент, выполняет сжатие 41 параметра входного вектора в 12-размерный выходной вектор. MLP обрабатывает полученные в результате сжатия данные и дает заключение относительно входного вектора: является ли он атакой определенного типа или же это нормальное соединение.

Такой детектор в проектируемой системе будет специализироваться на одном определенном типе атак. На выходе детектора возможны два состояния: “да” – если входной образ принадлежит заданному типу атаки, “нет” – входной образ не является атакой.

В мультиагентной системе можно использовать детекторы другого вида [3].

После выполнения процедуры обучения нейронные сети могут использоваться в задаче обнаружения вторжений.

3. МУЛЬТИАГЕНТНАЯ НЕЙРОННАЯ СЕТЬ

В мультиагентной нейронной сети применяется множество детекторов, специализирующихся в различных областях знаний.

Реальные иммунные системы слишком сложны, чтобы можно было применить все имеющиеся в них механизмы защиты. Но в данном случае не нужны все возможности биологических иммунных систем. В ходе построения мультиагентной системы для обнаружения вторжений использованы лишь основные принципы и механизмы реальных иммунных систем, такие как: генерация и обучение детекторов с различной структурой и специализацией, отбор подходящих детекторов, возможность детекторов обнаруживать аномальную активность, клонирование и мутация детекторов, формирование иммунной памяти.

4. РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ

Результаты эксперимента приведены в таблицах 1 и 2. Записи об атаках класса DoS и Probe распознаны системой в более чем 90% случаев. Несколько хуже результат в случае U2R и R2L. Также присутствуют так называемые ложные срабатывания системы.

Таблица 1 – Обучающая и тестовая выборка

	DoS	U2R	R2L	Probe	Normal	всего
обучающ выборка	3571	37	278	800	1500	6186
тестовая выборка	391458	52	1126	4107	97277	494020

Таблица 2 – Обнаружение атак при помощи мультиагентной НС

класс	кол-во	обнаружено	распознано
DoS	391458	386673 (98.78%)	368753 (94.20%)
U2R	52	47 (90.39%)	45 (86.54%)
R2L	1126	1097 (97.42%)	930 (82.59%)
Probe	4107	4066 (99.00%)	4016 (97.78%)
Normal	97277	---	82903 (85.22%)

Таблица 3 – Обнаружение неизвестных атак

тип	кол-во	обнаружено	тип	кол-во	обнаружено
Normal	75952	74340 (97.88%)	Multihop*	7	5 (71.43%)
Back	2203	2169 (98.46%)	Phf*	4	0 (0.00%)
Land*	1	1 (100.00%)	Spy*	2	0 (0.00%)
Neptune	901	900 (99.89%)	Warezclient	1015	981 (96.65%)
Buffer_overflow	30	26 (86.67%)	Warezmaster	20	19 (95.00%)
Loadmodule	9	9 (100.00%)	Ipsweep	9	9 (100.00%)
Perl*	3	0 (0.00%)	Nmap*	2	2 (100.00%)
Rootkit*	7	3 (42.86%)	Portsweep	15	15 (100.00%)
ftp_write*	6	5 (83.33%)	Satan	10	8 (80.00%)
Guess_passwd	53	53 (100.00%)			

* - атаки, которые отсутствовали в обучающей выборке

Из таблицы 3 следует, что многие записи о неизвестных системе обнаружения вторжений атаках были правильно отработаны как “атака”. Это свидетельствует о том, что такая мультиагентная система обладает способностью к обобщению и может использоваться для обнаружения ранее неизвестных типов активности в сети.

ЗАКЛЮЧЕНИЕ

В данной работе предложена концептуальная модель построения мультиагентной нейронной сети на базе механизмов искусственных иммунных систем и искусственных нейронных сетей.

Такая система характеризуется: i) гибкостью, ii) распределенностью, iii) самоорганизацией, iv) возможностью дообучения в процессе работы.

Литература

1. Войцехович, Л.Ю., Головки, В.А., Кочурко П.А. и Войцехович Г.Ю. Система обнаружения атак как основной элемент защиты компьютерной сети / Л.Ю. Войцехович, В.А. Головки, П.А. Кочурко и Г.Ю. Войцехович // Вестник БрГТУ. – 2008. - №5(53): Физика, математика, информатика. – С. 12-19.

2. 1999 KDD Cup Competition. - Information on: <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>.

3. Vaitsekhovich, L. and V. Golovko. Employment of neural network based classifier for intrusion detection / L. Vaitsekhovich and V. Golovko // Acta Mechanica et Automatica. Bialostok Technical University. Faculty of Mechanical Engineering. – 2008. – Vol. 2, No 4(6). – P. 93-98.

УДК 62-529

ПРИМЕНЕНИЕ ПОСЛЕДОВАТЕЛЬНОГО НЕЙРОКОНТРОЛЛЕРА В АСУТП

Иванюк Д. С., Головки В.А., Шуть В.Н.

УО «Брестский государственный технический университет», г. Брест
ОАО «Савушкин продукт», г. Брест

Нейроуправление – наука относительно молодая, одна из задач которой заключается в построении систем управления (систем принятия решений), которые могут обучаться во время функционирования и таким образом улучшать свою эффективность работы. Для того чтобы алгоритмы управления могли применяться на практике, они должны быть достаточно простыми для реализации и понимания, обладать способностью к обучению,

гибкостью, устойчивостью, нелинейностью. Таким образом, нейронные сети из-за своей способности обучаться на основе соотношения «вход-выход», нелинейными обобщающими способностями пригодны для решения задач управления, которые принципиально связаны с нелинейными характеристиками. В настоящее время существует большое количество подходов к нейронному управлению. В данной статье рассматривается возможность применения последовательного нейроконтроллера для управления технологическим процессом. В такой последовательной схеме управления нейронная сеть непосредственно обучается отображению желаемых (опорных) сигналов в управляющие воздействия, необходимые для получения таких сигналов. Для исследования использовались технологические данные работы проекта «Танки 1-12», созданного в SCADA-системе «EasyServer» на предприятии ОАО «Савушкин продукт». Нейроконтроллер был реализован в виде отдельного модуля (язык C++) для программируемого контроллера управления технологическим процессом ICP CON 7186EXD. Далее он был внедрен в систему управления пастеризационной установкой №2 в проект «Танки 1-12».

Последовательная схема управления

Данная схема наиболее проста, что является как основным достоинством (несмотря на относительную простоту, подходит для решения широкого круга задач), так и недостатком (требует переобучения при изменении параметров объекта управления). Общая схема приведена на рисунке 1.



Рисунок 1 – Последовательная схема нейронного управления

Обучение контроллера происходит следующим образом: предположим, что в момент времени $t+1$ в памяти хранится текущее значение выхода $y(t+1)$, $q+n$ значений того же процесса в предшествующие моменты времени и $p+n$ значений входного процесса u (управляющее воздействие) в предшествующие моменты времени (таким образом, p, q определяют размер окна для выходного и управляющего сигнала соответственно, n – количество обучающих выборок).

Обучающую выборку нейронной сети (нейроконтроллера) можно представить в виде матрицы, строки которой характеризуют векторы, которые подаются на вход сети. Пусть только что получена величина $y(10)$, а $y(11)$ еще не известна, и $p=3, q=3, n=3$. Также величины $y(9), y(8), \dots, y(5)$ и $u(9), u(8), \dots, u(5)$ имеются в памяти. Тогда получаем:

$$X = \begin{bmatrix} y(7) & y(6) & y(5) & u(7) & u(6) & u(5) \\ y(8) & y(7) & y(6) & u(8) & u(7) & u(6) \\ y(9) & y(8) & y(7) & u(9) & u(8) & u(7) \end{bmatrix}$$

Эталонные значениями для обучения соответственно будут величины $u(8), u(9), u(10)$. То есть контроллер прогнозирует значение управляющего воздействия на основе предыдущих значений входа и выхода.

Таким образом, если имеются сохраненные данные о работе системы за некоторый интервал, то их можно использовать для предварительного обучения контроллера.

Нейроконтроллер объекта управления на основе прямого инверсного обучения.

В проекте «Танки 1-12» реализовано управление пастеризационной установкой №2. Схема установки приведена на рисунок 2.

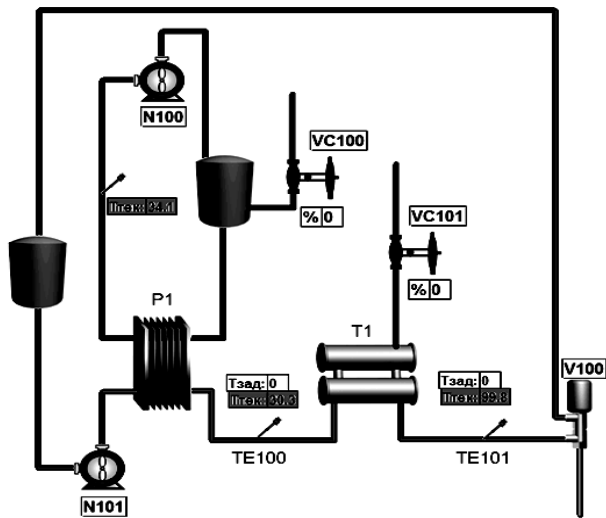
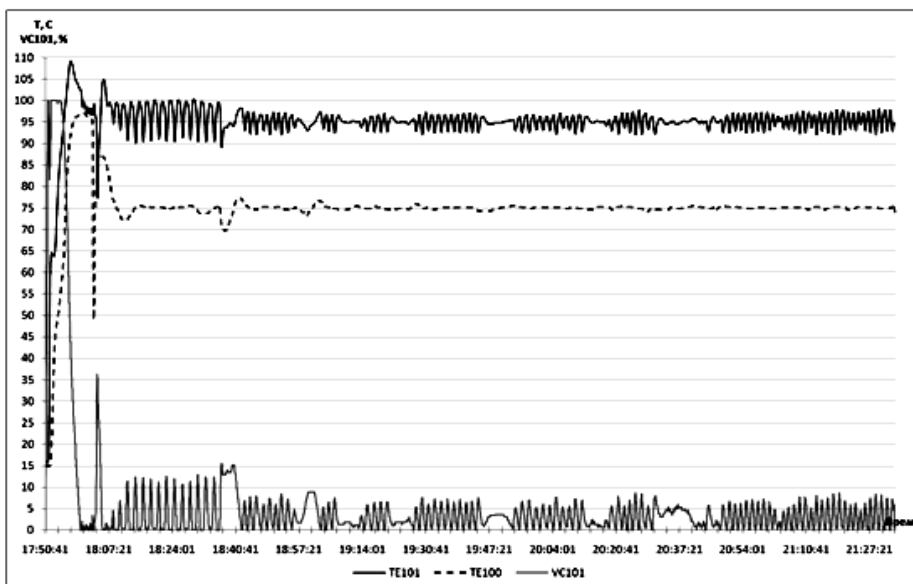


Рисунок 2 – Схема пастеризационной установки №2

Установка в режиме пастеризации работает следующим образом: молоко из бачка насосом N101 подается в пластинчатый теплообменник P1, где оно подогревается, далее идет на гомогенизатор, затем пастеризуется в трубчатом теплообменнике T1. Управление заключается в поддержании температур TE100 (температура гомогенизации) и TE101 (температура пастеризации) в заданных пределах путем открытия управляемых паровых клапанов VC100 и VC101. Температура TE100 должна поддерживаться в пределах 75 ± 2 °C, TE101 – 95 ± 2 °C.

В качестве нейроконтроллера был выбран многослойный персептрон со следующей структурой: 30 входных элементов, 15 – скрытых, 1 – выходной. В качестве функции активации скрытых нейронов использовалась сигмоидная функция, выходных – линейная. Для обучения использовались сохраненные данные работы установки № 2. Обучение производилось на первых 300 точках данных (рисунок 3), размер окна – 10. Далее обученная сеть прогнозировала вход системы.

Для обучения использовались сохраненные данные работы установки № 2. Обучение производилось на первых 300 точках данных (рисунок 3), размер окна – 10. Далее обученная сеть прогнозировала вход системы.



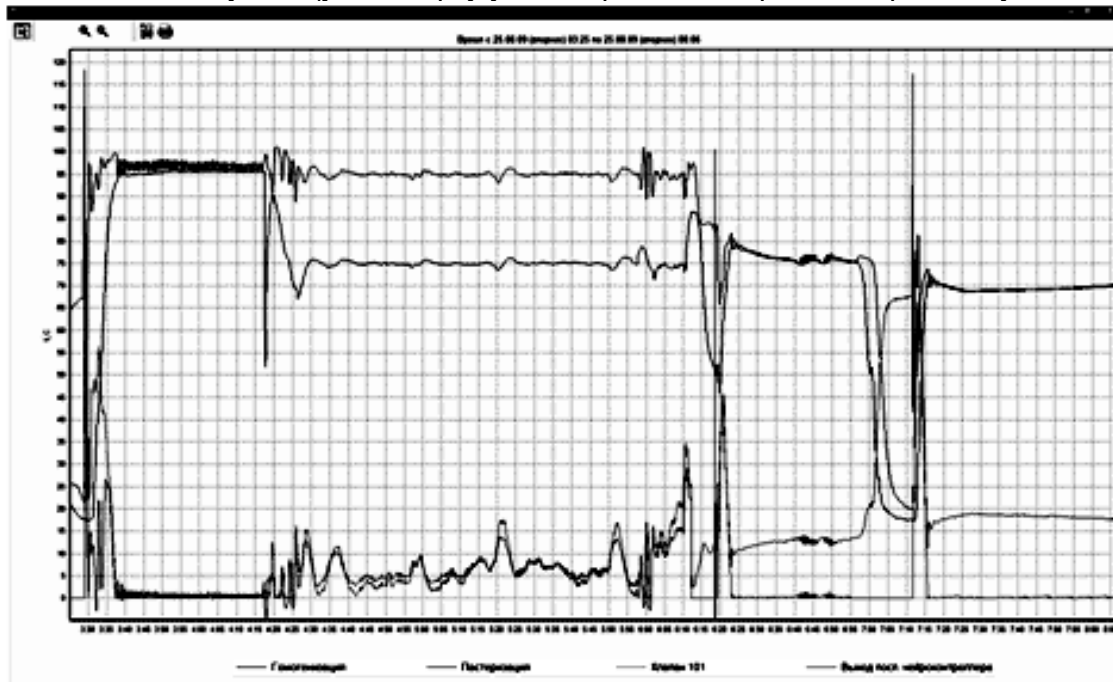
производилось на первых 300 точках данных (рисунок 3), размер окна – 10. Далее обученная сеть прогнозировала вход системы.

Рисунок 3 – График работы установки №2

Обучение происходило на персональном компьютере со следующими параметрами: процессор E6550 2.33ГГц, память 2Гб, операционная система Windows Vista Business. Таблица 1 отражает ход обучения. Обучение до точности 0.02 произошло за 19517 итераций (140 сек), до точности 0.012 – 188385 итераций (21 мин). После обучения такой нейроконтроллер повторяет действия обычного ПИД-контроллера с достаточной точностью и может использоваться как его замена.

Тестирование нейроконтроллера в составе проекта

Для тестирования нейроконтроллер был встроен в программу управления технологическим проектом «Танки 1-12». Как видно из графика, на участке, где проводилось обучение (режим работы – пастеризация), нейроконтроллер практически точно повторяет управление ПИД-контроллера (Рисунок 4). На остальных участках (режим работы – мойка) нейроконтроллер управляет с некоторым постоянным смещением относительно ПИД-контроллера, величина которого зависит от режима. Это объясняется тем, что установка переходит в новый режим, меняются параметры ее работы, и выход нейроконтроллера уже не соответствует требованиям управления. Для устранения данного недостатка необходимо включить в обучающую выборку участки различных режимов работы установки.



**Рисунок 4 – Работа последовательного нейроконтроллера.
Полный цикл работы установки (работа, мойка)**

Выводы

Последовательный нейроконтроллер может успешно использоваться в качестве замены обычного ПИД-контроллера. На его основе могут быть построены любые нелинейные системы управления, если имеются необходимые для обучения сохраненные данные работы системы (например, работа системы в режиме ручного управления). Если в системе используются несколько ПИД-контроллеров (для разных режимов работы), то нейроконтроллер может использоваться как единый регулятор работы системы во всех режимах (после соответствующего обучения).

Литература

1. Neurocontrol: An Overview for the Practitioner. – http://www.werbos.com/Neural/Neurocontrollers_1999.htm.
2. Сигеру Омату. Нейроуправление и его приложения. Книга 2 / Сигеру Омату, Марzuki Халид, Рубия Юсоф. – М.: Издательское предприятие редакции журнала «Радиотехника», 2000.
3. Усков, А.А. Интеллектуальные технологии управления. Искусственные нейронные сети и нечеткая логика / А.А. Усков, А.В. Кузьмин. – М.: Горячая линия – Телеком, 2004.

УДК 004.056.57:032.26

КОЛЛЕКТИВНОЕ ПОВЕДЕНИЕ В МНОГОАГЕНТНЫХ СИСТЕМАХ НА ОСНОВЕ ПОДКРЕПЛЯЮЩЕГО ОБУЧЕНИЯ

Кабыш А.С.

УО «Брестский государственный технический университет», г. Брест

Введение. Многоагентное обучение – новая область исследований в машинном обучении (machine learning). Также подкрепляющее обучение – это последнее, самостоятельно выделившееся, направление машинного обучения, делающее только первые шаги. На пересечении этих двух областей находится множество нерешенных проблем. Одна из них – это коллективное обучение, или задача адаптации одиночного обучения на многоагентный подход. Агент – сущность, находящаяся в какой-то среде и способная действовать в ней. Многоагентная система (МАС) – это группа взаимодействующих агентов. Более формально[1], многоагентный подход – это целостная парадигма к построению сложных систем, состоящих из взаимодействующих агентов, которые, оперируя локальными знаниями и ограниченными возможностями (ресурсами), способны, благодаря своему взаимодействию, достигать ожидаемого глобального поведения системы. Так, муравьи образуют муравейники – превосходно организованные общества с разделением труда; взаимодействие множества людей порождает мировую экономику, а взаимодействие живых клеток создает функционирующий организм. Обучение многоагентных систем имеет свои базовые принципы:

1. Один агент не может обучаться отдельно от других агентов.
2. Структуры взаимодействий внутри многоагентной системы должны быть включены в алгоритм обучения.
3. Многоагентное обучение – это во многом принципиально новый вид обучения, не сводящийся к разделению обучения отдельных агентов[1].

Цель данной работы – адаптация подкрепляющего обучения для многоагентных систем. Как результат – мы получаем новый класс алгоритмов обучения для широкого круга задач. Рассмотрим более подробно подкрепляющее обучение.

Подкрепляющее обучение

Подкрепляющее обучение (Reinforcement Learning, RL) – это область искусственного интеллекта и способ машинного обучения, предназначенная для обучения автономных агентов путем их взаимодействия с внешней средой для достижения в ней оптимального поведения [2]. Подкрепляющее обучение возникло на пересечении таких областей наук, как динамическое программирование, машинное обучение, исследование рефлексов, когнитивные процессы и биология живых организмов. Подкрепляющее обучение применяется в робототехнике, теории адаптивного управления, теории адаптивного обучения и в теории игр.

Стандартная модель подкрепляющего обучения имеет вид [2,3]:

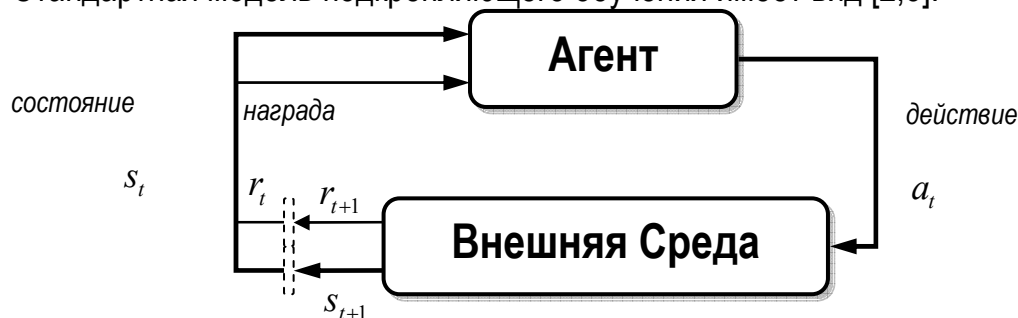


Рисунок 1 – Стандартная модель подкрепляющего обучения

Агент взаимодействует с внешней средой в дискретные моменты времени $t = 0, 1, 2, 3, \dots$. В каждый момент времени t агент получает некоторое представление внешней среды – внутреннее состояние $s_t \in S$, где S – это множество всех возможных состояний. На основании текущего состояния выбирается действие $a_t \in A(s_t)$, где $A(s_t)$ множество возможных действий в состоянии s_t . В следующий момент времени, после выполнения действия, агент и переходит в новое состояние s_{t+1} , и получает численную награду $r_{t+1} = \mathfrak{R}(s_t, a_t, s_{t+1})$, где $\mathfrak{R}(s_t, a_t, s_{t+1})$ – функция награды. Имея значение награды, агент может оценить качество или полезность перехода $s_t \xrightarrow{a_t} s_{t+1}$ и произвести обучение. В каждый момент времени агент осуществляет отображение из состояния в вероятности выбора каждого возможного действия. Это отображение называется *политикой* π агента и обозначается $\pi(s, a)$, где при условии, что $s = s_t$, выбирается некоторое $a = a_t$.

Награда играет роль обратной связи, говорящей о том, какое действие было «плохим» или «хорошим». По сути, подкрепляющее обучение – это метод проб и ошибок. Агент исследует среду и, если находит положительное действие, получает положительную награду, и наоборот – отрицательную; алгоритм подкрепляющего обучения устроен таким образом, что вероятность в будущем выбрать действие с положительной наградой возрастет.

Математическое обоснование подкрепляющего обучения получено из динамического программирования, а также статистических методов моделирования (метод Монте-Карло), и основано на итеративном варианте формулы Беллмана. Подкрепляющее обучение работает на Марковских процессах принятия решений (МППР). Другими словами, среда, в которой действует агент, должна описываться МППР; как следствие, подкрепляющее обучение хорошо подходит к теории игр.

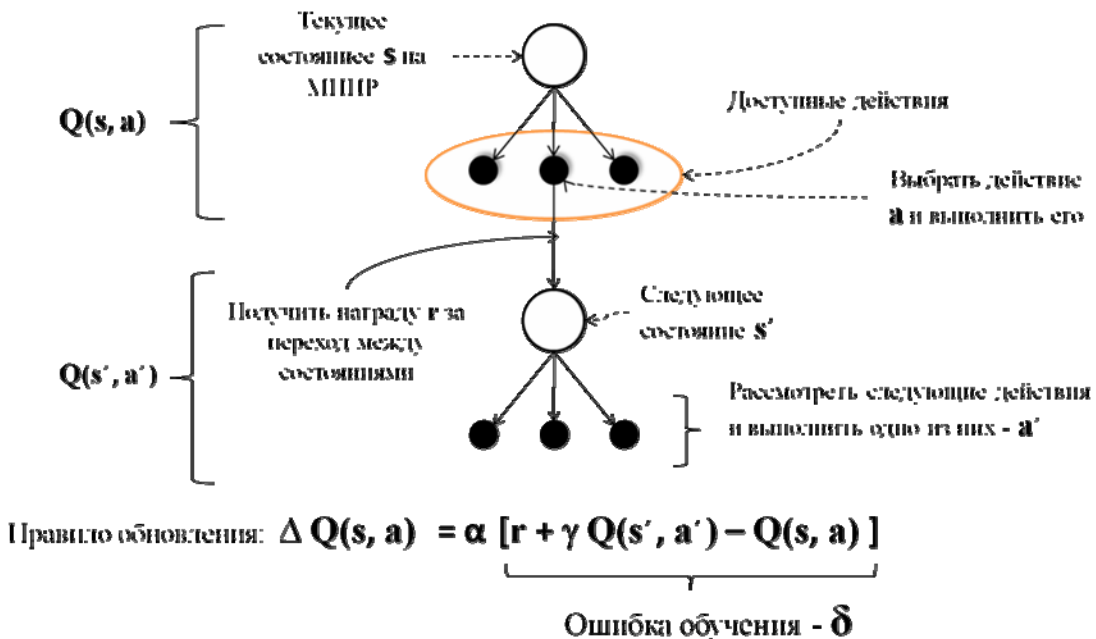


Рисунок 2 - Итеративное подкрепляющее обучение по методу временной разности

Где $Q(s, a)$ – это значение Q-функции, функции полезности некоторого действия a в некотором состоянии s . Разность ценности двух пар состояние-действие с учетом награды составляет ошибку обучения δ , которая может быть использована для обновления значений ценности:

$$Q(s, a) = Q(s, a) + \alpha [r_t + \gamma Q(s', a') - Q(s, a)] \quad (1)$$

Другой, наиболее известный алгоритм подкрепляющего обучения называется *Q-Learning* и выполняет обновление по следующей формуле:

$$Q(s, a) = Q(s, a) + \alpha[r_t + \gamma \max_{a'} Q(s', a') - Q(s, a)] \quad (2)$$

Обучение закончено, когда ошибка обучения δ становится равной 0, или меньше заданного порога, тогда оптимальная политика выбирается при помощи следующего выражения:

$$\pi^* = \max Q(s, a) \quad (3)$$

Для хранения Q-функции используется либо таблица (для небольших пространств), либо аппроксиматор. Основное преимущество аппроксиматора – способность проводить обобщение в пространстве состояний-действий. Аппроксиматором может быть любая параметрическая функция, например, деревья решений, нейронные сети, RBF-функция и другие.

Основным преимуществом подкрепляющего обучения является способность адаптироваться к обучаемой среде; она может быть полностью неизвестна, но и даже в этом случае агент может достичь оптимального поведения.

Многоагентное подкрепляющее обучение

В первых работах по коллективному подкрепляющему обучению [5-6] был использован подход, при котором вся Многоагентная система рассматривалась как один агент – объединенное обучение. Такой подход позволял успешное обучение, но оно ничем не отличалось от стандартной RL модели. Сходимость алгоритма была медленной, а пространство состояний-действий характеризовалось большой размерностью. Возникла потребность в многоагентном обучении, которое решало бы вышеназванные проблемы подкрепляющего обучения. Эта модель была разработана и представлена в [7]. Основные её положения сводятся к следующим принципам:

1. Любую сложную задачу всегда можно разложить на подзадачи и поручить её решение отдельным агентам. Награда задает некоторую цель и корректирует процесс её достижения. Следовательно, *выделив награды, можно выделить цели*. Но способ начисления наград зависит от структуры многоагентной системы; один агент может начислять награду другому и т.д. Данный принцип получил название принципа декомпозиции и лежит в основе предложенного проекционного подхода к подкрепляющему обучению. Он позволяет декомпозировать задачу обучения на подзадачи, чем во многом решает проблему проклятья размерности. Для полноценного обучения декомпозиции недостаточно.

2. Тесная взаимосвязь агентов, выраженная в направленности на решение общей задачи, делает их обучение взаимосвязанным. Аналогичное решение было применено и к подкрепляющему обучению. В многоагентной системе действия агентов направлены на других агентов, не только во внешнюю среду, как показано на рисунке 1. Если предположить, что значение $Q(s', a')$ в формулах (1) и (2) может быть получено от другого агента, то становится возможным обучение агентов относительно друг друга. Такое разделение отражает тот факт, что текущее состояние одного агента может зависеть от действий другого. Таким образом, можно определить ошибку действий одного агента, относительно другого.

Данные принципы были опробованы на задаче управления многозвенным роботом. Многозвенный робот - это робот с N степенями свободы, где N - количество узлов. Каждый узел робота представляет собой одного агента. Каждый узел робота, кроме последнего, может изменять положение, ориентацию всех последующих узлов относительно своей позиции на 360 градусов. Другими словами, произвольный узел может вращать всю последующую структуру робота по всей окружности с центром в точке с данным узлом (радиус вращения зависит от количества сегментов робота и размерности среды). Центральный (корневой) узел робота не изменяет своего местоположения. Последний, терминальный узел, не изменяет своего местоположения; его положение зависит от со-

гласованных действий предыдущих агентов. При обучении необходимо научиться согласовывать движение частей робота таким образом, чтобы терминальный элемент попадал в целевую точку.

В задаче управления пятизвенным роботом, благодаря декомпозиции, пятимерное пространство состояний действий было преобразовано в пять одномерных, обучение в которых происходит значительно быстрее. Относительное обучение позволило обучаться с учетом структуры многоагентной системы.

Моделирование многоагентной системы, построенной на вышерассмотренных принципах, показало следующие результаты. Во первых, сходимость обучения была на порядок быстрее, чем при объединенном обучении (60-100 эпизодов против 1000). Во вторых, была выявлена способность многоагентной системы к синхронизации действий. В третьих, был наглядно продемонстрирован принцип, что «обучение – это обобщение», т.к. робот легко перестраивался с цели на цель. В четвертых, стратегия поведения робота значительно изменялась в зависимости от выбранного алгоритма обучения. В пятых, моделирование показало наиболее оптимальное поведение для данного робота, которое значительно отличалось от ожидаемого.

Литература

1. Hosc M. Vidal. *Fundamentals of Multiagent Systems with Net Logo Examples*. (www.multiagent.com)
2. Richard S. Sutton, Andrew G. Barto. *Reinforcement Learning: An Introduction* Cambridge : MIT Press., 1998
3. Tesauro, G. J. (1994). TD-gammon, a self-teaching backgammon program, achieves master-level play. *Neural Computation*, 6(2):215--219. (<http://www.research.ibm.com/massive/tdl.html>)
4. Dr. Florentin Woergoetter, Dr. Bernd Porr. Статья *Reinforcement Learning* на ресурсе <http://www.scholarpedia.org>. (http://www.scholarpedia.org/article/Reinforcement_learning).
5. Кабыш, А.С. *Коллективное поведение агентов на основе подкрепляющего обучения*. Нейроинформатика / А.С. Кабыш, В.А. Головко. – 2009. – Часть 1. – С. 191-200.
6. Kabysh, A.S., Golovko V.A., *Collective Behavior in Multiagent Systems Based on Reinforcement Learning*, PRIP-2009: Proceedings of the Tenth International Conference (19-21 May, Minsk, Republic of Belarus) / Kabysh, A.S., Golovko V.A. – Minsk, 2009. – С. 260-264.
7. Kabysh, A.S. *Collective Behavior in Multi-Agent Systems*, OWD 2009 Ph.D. workshop, Eastern Europe Summer School, 12-24 October, Silesian University of Technology. – Poland. – 2009. – P. 92-97.

УДК 004.89

ИСПОЛЬЗОВАНИЕ ИСКУССТВЕННЫХ ИММУННЫХ СИСТЕМ И НЕЙРОННЫХ СЕТЕЙ ДЛЯ ОБНАРУЖЕНИЯ КОМПЬЮТЕРНЫХ АТАК

Комар М.П.

Тернопольский национальный экономический университет, г. Тернополь, Украина

В настоящее время обеспечение безопасности информации является одной из ключевых задач. Развитие компьютерных сетей и их объединение в глобальную сеть Интернет привело к росту числа преступлений, связанных с нарушением основополагающих принципов информационной безопасности: доступности, целостности и конфиденциальности информации. Несмотря на развитие средств защиты, таких как брандмауэры, количество

вторжений в информационные системы компаний возрастает с каждым годом. Для обнаружения компьютерных атак используются различные классы инструментальных средств, такие как системы обнаружения атак, системы предотвращения атак, сканеры уязвимостей, комплексные системы управления безопасностью. Однако использование этих средств сегодня ограничено рядом факторов [1]:

- высокая стоимость;
- сложность в использовании;
- низкая эффективность функционирования при наличии неизвестных атак;
- высокая нагрузка на компоненты сети.

Традиционные методы обнаружения атак, такие как сигнатурный метод или метод обнаружения аномалий, не позволяют достичь оптимальных характеристик обнаружения атак. Сигнатурный метод анализа основан на том, что большинство атак на систему известны и развиваются по схожим сценариям. В данном подходе сигнатуры вторжений определяют характерные особенности, условия, устройства и взаимосвязь событий, которые ведут к попыткам или собственно к вторжению.

К недостаткам сигнатурного анализа можно отнести:

- зависимость масштабируемости и производительности от размера базы данных сигнатур;
- обновление базы данных сигнатур затруднительно ввиду отсутствия общепринятого языка описания, а добавление собственных сигнатур требует высокой квалификации;
- обновление базы данных сигнатур требуется при обнаружении нового типа атак, период обновления базы должен быть невелик.

В настоящее время исследования в области защиты информации ведутся в направлении разработки средств, которые позволяют решить часть этих проблем за счет использования интеллектуальных технологий. Вместе с тем, многие вопросы при построении этих систем, связанные с эффективностью применения новых методов и технологий и их реализацией в режиме реального времени, остаются открытыми и не до конца исследованными, поэтому разработка интеллектуальных систем обнаружения атак с использованием таких перспективных направлений, как искусственные иммунные системы и нейронные сети, является актуальной задачей.

Модели, основанные на принципах функционирования систем иммунитета, применяются в различных областях науки и техники. Сфера их применения включает следующие области (но не ограничивается ими): методы вычислений; когнитивные модели; искусственные иммунные системы для распознавания образов; методы обнаружения аномалий и неисправностей; мультиагентные системы; модели самоорганизации; модели коллективного интеллекта; системы поиска и оптимизации; модели автономных распределенных систем; модели искусственной жизни; системы компьютерной и интернет-безопасности; модели обучающихся систем; методы извлечения информации; искусственные иммунные системы для выявления подделок; методы обработки сигналов и изображений [2].

Для построения системы обнаружения компьютерных атак предлагается использовать методы искусственных иммунных систем и нейронных сетей.

Для достижения поставленной цели необходимо решить следующие задачи:

- предложить структуру системы обнаружения атак на основе искусственных иммунных систем и нейронных сетей;
- разработать методы и алгоритмы функционирования системы обнаружения компьютерных атак;
- разработать программное обеспечение системы.

Предполагается, что разработка системы обнаружения компьютерных атак, основанной на применении методов искусственных иммунных систем и нейронных сетей, позволит существенно повысить вероятность обнаружения неизвестных сетевых вторжений.

Литература

1. Кашаев, Т.Р. Применение искусственной иммунной системы для решения задачи обнаружения атак / Материалы 3-й Всероссийской зимней школы – семинара аспирантов и молодых ученых / Т.Р. Кашаев. – Уфа: УГАТУ, 2008. – С. 326-332.

2. Дасгупта, Д. Искусственные иммунные системы и их применение / Д. Дасгупта; пер. с англ. под ред. А.А. Романюхи. – М.: ФИЗМАТЛИТ, 2006. – 344 с.

УДК 004.8.032.26

НАСТРОЙКА ПОРОГОВ НЕЙРОСЕТЕВЫХ ДЕТЕКТОРОВ ДЛЯ РАСПОЗНАВАНИЯ КЛАССОВ СЕТЕВЫХ АТАК

Кочурко П.А.

УО «Брестский государственный технический университет», г. Брест

При решении задач обнаружения попыток несанкционированного доступа к системе можно выделить два основных подхода: обнаружение аномалий и обнаружение злоупотреблений.

Нелинейные рециркуляционные нейронные сети (РНС) способны выступить в качестве детекторов СОА, реализующей обе технологии. Известно [1], что именно объединение обеих технологий в рамках одной системы может позволить повысить качество обнаружения и снизить уровень ложных срабатываний.

В случае применения нелинейных рециркуляционных сетей (РНС) в качестве детектора аномалий [2] обучение РНС производится на нормальных соединениях таким образом, чтобы входные вектора на выходе восстанавливались в себя, при этом, чем соединение более похоже на нормальное, тем меньше ошибка реконструкции:

$$E^k = \sum_j (\bar{X}_j^k - X_j^k)^2, \quad (1)$$

где X_j^k – j -й элемент k -го входного вектора, \bar{X}_j^k – j -й элемент k -го выходного вектора. Если $E^k > T$, где T – некий заданный для данного детектора порог, то соединение признаётся аномалией, или атакой, иначе – нормальным соединением.

Таким образом, одна РНС может применяться для определения принадлежности входного вектора к одному из двух классов – тому, на котором обучалась (класс A), или ко второму (класс \bar{A}), которому соответствуют далеко отстающие вектора. Объединив в одной системе N подобным образом обученных детекторов, каждый из которых отвечает за анализ принадлежности входного вектора к одному из классов A_i , можно успешно решать задачу распознавания типа или класса атаки. Для этого необходимо анализировать относительную ошибку реконструкции

$$\delta_i^k = \frac{E_i^k}{T_i}, \quad (2)$$

где T_i – порог i -го детектора, изначально $T_i = \text{mean} \delta_i^k$. Чем меньше δ_i^k , тем более вероятна принадлежность входного k -го образа к классу A_i .

После обучения нейродетекторов всех классов необходимо решить задачу настройки порогов T_i таким образом, чтобы уменьшить ошибки неверной классификации. Для этого применяется следующий алгоритм:

1) вычисляется матрица результатов классификации C , где C_{ij} – количество векторов класса A_i , определенных как вектора класса A_j ; при этом если $i \neq j$, то это данное значение указывает, сколько векторов классифицированы ошибочно. Для того, чтобы уменьшить данное значение, необходимо уменьшить для каждого k -го вектора δ_j^k и увеличить δ_i^k . Исходя из (2), для этого вычисляются суммарные относительные ошибки для всех векторов, принадлежащих к классу A_i , но определенных как вектора класса A_j :

$$ER_{ij} = \sum_{k=1}^{C_{ij}} \left(\frac{E_i^k}{E_j^k} T_j - T_i \right), \quad (3)$$

для $\forall i = 1..N, j = 1..N, i \neq j$;

2) для $\forall i = 1..N$ вычисляется новое значение порога:

$$T_i = T_i + \alpha \frac{\sum_{j=1}^{N, i \neq j} ER_{ij}}{\sum_{j=1}^{N, i \neq j} C_{ij}} - \beta \frac{\sum_{j=1}^{N, i \neq j} ER_{ji}}{\sum_{j=1}^{N, i \neq j} C_{ji}}, \quad (4)$$

где $\alpha = \beta = 0,01$;

3) если не достигнуто максимальное количество итераций, то – переход к пункту 1).

На рисунках 1 и 2 изображены графики суммарного изменения порогов и изменения ошибки неверной классификации MC .

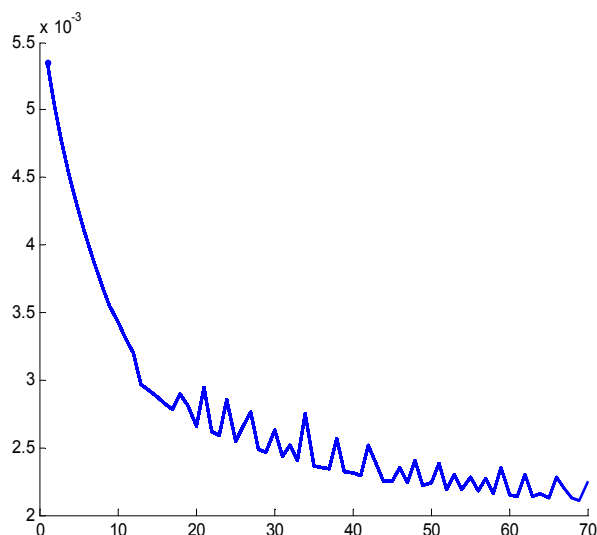


Рисунок 1 – Суммарное изменение

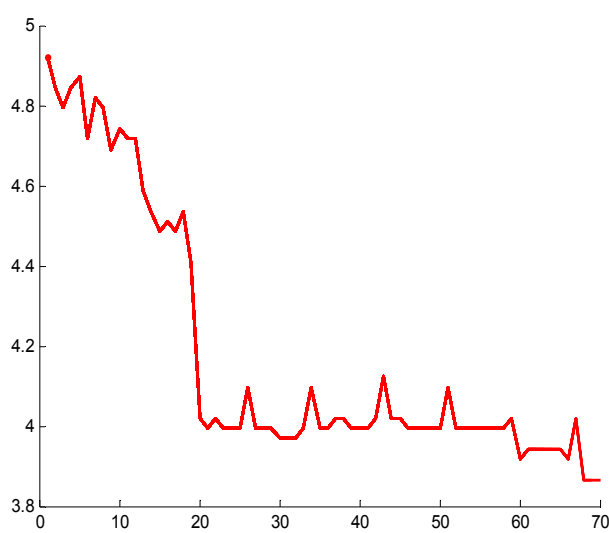


Рисунок 2 – Ошибка неверной значения порогов классификации

В таблицах 1–3 приведены результаты распознавания 23 типов атак [3] (в том числе 1-й тип – нормальные соединения) до и после настройки порогов, а также распознавания типов на тестовой выборке. Результаты представлены парой значений, выраженных в процентах: DR – процент корректно распознанных атак данного типа, FAR – процент ложных срабатываний, а также третьей синтетической характеристикой SE – суммой ошибок первого и второго рода.

Таблица 1 – Результаты распознавания типов атак до настройки порогов на обучающей выборке (3880 соединений), $MC = 5,05\%$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
DR	91,7	96,4	100	96,7	95	100	89,5	100	83,3	76,7	98,9	97,1	100	100	100	96,6	95	86,5	100	100	100	74	98
FAR	2,91					0,03		0,16				0,84			0,16	0,11	0,13	0,03				0,16	0,58
SE	11,2	3,6		3,3	5	0,03	10,5	0,16	6,7	23,3	1,1	3,74			0,16	3,51	5,13	13,5				26,2	2,6

Таблица 2 – Результаты распознавания типов атак после настройки порогов на обучающей выборке (3880 соединений), $MC = 3,87\%$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
DR	87,1	98,2	100	96,7	97,5	100	93,2	100	100	85,3	99,1	82,9	100	100	100	95,4	95	97	100	100	100	93,3	76
FAR	1,07					0,03	0,38			0,64		0,26			0,16	0,08	0,13	0,38				0,24	0,58
SE	14	1,8		3,3	2,5	0,03	7,18			15,3	0,9	17,4			0,16	4,68	5,13	3,4				6,94	24,6

Таблица 3 – Результаты распознавания типов атак после настройки порогов на тестовой выборке (505291 соединений), $MC = 1,92\%$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
DR	94,2	85,2	93,8	87,5	98,1	10,3	23,3	95,3	100	85,7	99,7	44,6	100	100	99,2	96,6	80	97,6	99,8	100	100	80,3	85
FAR	0,41	0,32		0,01	0,12		0,04		0,01		0,03	0,02				0,06	0,01	0,64				0,34	

Сравнивая результаты распознавания типов атак в первой и второй таблицах, можно сделать вывод, что на различных детекторах настройка порогов отражается по-разному: ухудшается качество распознавания (детекторы №№ 1, 12 и 23), остается на том же уровне или значительно улучшается (детекторы №№ 2, 5, 7 и, особенно, 10, 18 и 22). Это значит, что система пытается за счет проигрыша в небольшом количестве детекторов достичь выигрыша в большем количестве детекторов и в целом по системе.

Литература.

1. Giacinto, G. et al. Selection of image classifier // G. Giacinto, F. Roli, G. Fumera. – Electron. – №26(5), 2000. – P. 420-422.
2. Кочурко, П.А. Нейросетевой детектор аномалий / П.А.Кочурко // Известия Белорусской инженерной академии. – 2005. – № 1(19) – С. 78-81.
3. KDD Cup 1999 Data / The UCI KDD Archive, Information and Computer Science. – University of California, Irvine, 1999.

УДК 004.8.032.26

ОПТИМИЗАЦИЯ НЕЙРОСЕТЕВОЙ СИСТЕМЫ ДЛЯ АНАЛИЗА ЭЛЕКТРОЭНЦЕФАЛОГРАММ

Лаврентьева С.В.

УО «Брестский государственный технический университет», г. Брест

Введение

Автоматическое обнаружение эпилептиформной активности в сигналах электроэнцефалограмм (ЭЭГ) является актуальной задачей. Результаты исследования динамики значения старшего показателя Ляпунова для сигналов электроэнцефалограмм (ЭЭГ) показывают неточность обнаружения интервала, содержащего эпилептиформную активность, а также высокий процент ложных срабатываний алгоритма [1]. Причиной таких результатов работы метода является анализ нестационарного сигнала. Когда в обучающую выборку

попадает граница сегментов, при переходе которой меняются характеристические параметры сигнала, результаты расчета старшего показателя Ляпунова будут сильно зависеть от обучения нейронной сети. Обучение может привести к усреднению значения показателя на интервале обучения, также может быть рассчитано с большим влиянием одного из сегментов либо привести к резкому возрастанию показателя. Для устранения данного недостатка работы алгоритма предлагается до его использования проводить сегментацию сигнала на условно-стационарные сегменты и расчет старшего показателя Ляпунова производить отдельно на каждом сегменте. Такое решение позволит также точнее определять границы и длительность каждого сегмента, тем самым выделить участки с эпилептиформной активностью.

1. Нейросетевая система

Приведем алгоритм работы системы с кратким описанием используемых методов и результатов обработки сигнала ЭЭГ [2]:

1) Исходная группа сигналов обрабатывается при помощи метода независимых компонент (The Independent Component Analysis – ICA) [3]. Этот метод позволяет отфильтровать электрическую активность головного мозга от шумов и артефактов. В результате анализа выделяется один полезный ЭЭГ сигнал для последующей обработки.

2) Полученный на этапе 1) сигнал подвергается адаптивной сегментации при помощи прогнозирующей искусственной нейронной сети (ИНС). Так как сигналы ЭЭГ являются нестационарными, то данный подход позволяет получить последовательность сегментов, где в рамках каждого сегмента данные можно считать стационарными.

3) При помощи прогнозирующей ИНС производится расчет старшего показателя Ляпунова на каждом из сегментов. В результате получаем детерминированный ряд $\lambda(t) = (\lambda_1, \lambda_2, \dots, \lambda_N)$, где N – количество сегментов.

4) Согласно критерию (1) по значениям $\lambda(t)$ определяется наличие эпилептической активности в сигнале ЭЭГ:

$$\begin{cases} \lambda > 0, \text{ нормальная активность;} \\ \lambda \leq 0, \text{ эпилептиформная активность.} \end{cases} \quad (1)$$

Такой подход позволяет работать с малыми объемами данных, однако зависит от эффективности смоделированной ИНС. По результатам наших предыдущих исследований [1,4] наилучшие результаты для рассматриваемой задачи получаются при использовании многослойного персептрона с сигмоидной функцией активации в скрытом слое.

2. Описание экспериментальных данных

Различные эпилептиформные активности, которые могут присутствовать на ЭЭГ, характеризуются амплитудой, частотой сигнала, длительностью участка. Наиболее важным свойством для автоматического обнаружения участка эпилептиформной активности при помощи предложенного алгоритма является длительность аномальной активности на ЭЭГ. Согласно этой длительности можно определить размер минимального сегмента, который предполагается выделять на ЭЭГ и при этом корректно идентифицировать его принадлежность к классу нормальной либо аномальной активности. Если проанализировать возможную длительность эпилептиформной активности и минимальный период повторения (T_{\min}) таких активностей в случае образования комплексов на ЭЭГ, тогда размерность сегмента N ограничится исходя из неравенства (6) [2].

$$N \leq \frac{T_{\min} \cdot F}{\tau}, \quad (2)$$

где F – частота дискретизации сигнала, τ – временная задержка.

Для проведения исследований была сформирована база тестовых образов, состоящая из 60 фрагментов ЭЭГ сигналов с частотой дискретизации 250 Гц. При этом 30 фрагментов содержат в себе различные формы эпилептической активности и 30 представляют собой фрагменты ЭЭГ, характеризующиеся только нормальной активностью головного мозга.

При помощи программного продукта TESEAN 2.1 для экспериментальных данных была рассчитана величина $\tau = 1$. Таким образом, согласно условию (2) получим $N \leq 50$ при минимальном периоде повторения эпилептических вспышек в комплексах на ЭЭГ $T_{\min} = 0.2$ секунды [2]. С увеличением размерности обучающей выборки возрастает и время обучения до желаемой ошибки, при этом приемлемым можно принять интервал $N = [50, .. 70]$.

3. Экспериментальное исследование

Необходимо определить граничное значение старшего показателя Ляпунова, согласно которому мы будем классифицировать сегмент как нормальную или аномальную активность. Рассмотрим вариации значения минимального сегмента в интервале $N = [50, .. 70]$ и рассчитаем значение показателя для каждого образа тестовой базы и определим диапазон изменения данной величины $[\min, \max]$ для образов с нормальной активностью и для образов с аномальной активностью отдельно. Результаты анализа сведены в таблицу 1.

Таблица 1 – Результаты расчета старшего показателя Ляпунова при различных размерах обучающей выборки

N	Значение старшего показателя Ляпунова			
	Эпилептиформная активность		Нормальная активность	
	min	max	min	max
50	-0.12	-0.01	-0.04	0.13
55	-0.20	0.01	-0.02	0.17
60	-0.21	0.01	-0.01	0.16
65	-0.25	0.00	0.00	0.18
70	-0.31	-0.01	0.01	0.21

Результаты экспериментальных расчетов, приведенные в таблице 1, показывают, что при $N = 70$ диапазоны значений показателя не пересекаются, а следовательно, можно определить граничное значение $\epsilon = 0$ (среднее значение между \max для эпилептиформной активности и \min для нормальной активности сигнала).

Для $N = 70$ рассчитаем относительную величину K_r , характеризующую точность выполненной классификации:

$$K_r = \frac{C_r}{C_p} \cdot 100\%, \quad (3)$$

где C_r – количество правильно классифицированных образов, $C_p = 60$ – общее количество тестовых образов.

Результаты предложенного экспериментального анализа представлены на рисунке 1.

Как показано на рисунке 1, наилучший результат достигается при достижении ошибки обучения $E_w = 0.01$, так как все образы классифицируются правильно и при этом снижение данного значения приведет к нежелательному увеличению времени обучения.

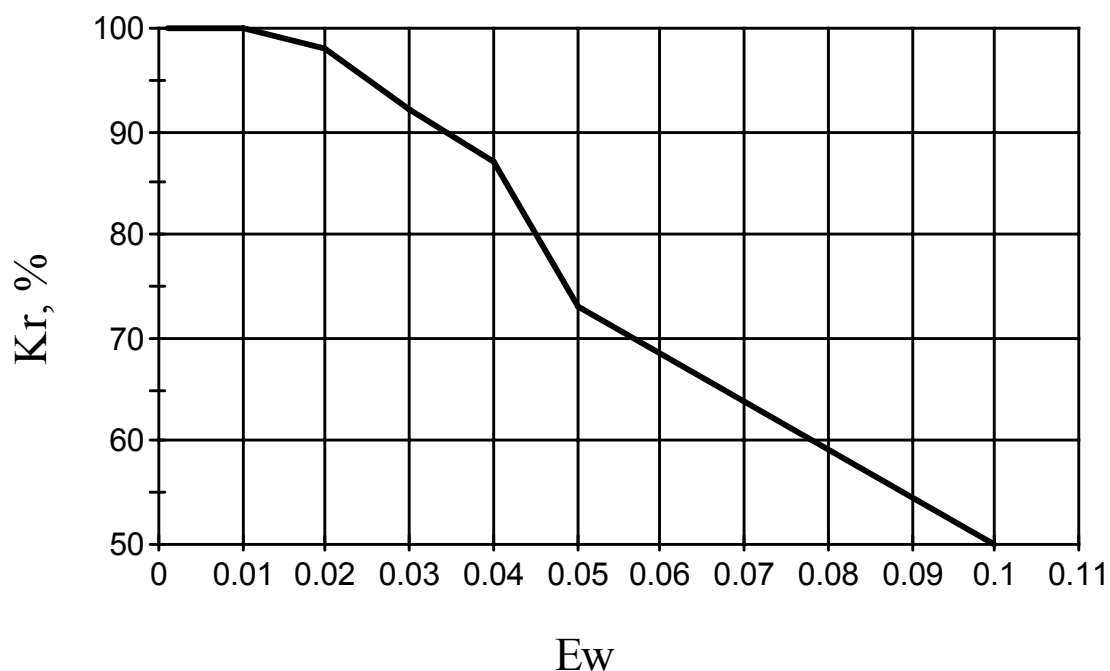


Рисунок 1 – Зависимость точности классификации исходных образов от значения достигаемой ошибки обучения нейронной сети E_w

Выводы

Представлено описание нейросетевой системы для анализа электроэнцефалограмм с целью обнаружения и выделения длительности эпилептиформной активности.

Проведены экспериментальные исследования для настройки параметров предложенного алгоритма с целями оптимизации его работы и повышения точности выделения участков и классификации их.

Исследования проводятся по заданию Министерства образования Республики Беларусь в соответствии с темой «Нейросетевая система анализа электроэнцефалограмм для обнаружения и распознавания эпилептиформной активности» (№ госрегистрации 20090433).

Литература

1. Golovko, V.A. Neural networks for chaotic signal processing: application to the electroencephalogram analysis for epilepsy detection / V.A. Golovko, S.V. Bezobrazova // Proceedings of the 4th International Conference on Neural Networks and Artificial Intelligence (ICNNAI'2006) / BrSTU. – Brest, 2006. – P.136-139.

2. Laurentsyeva, S. Electroencephalogram Analysis Based on Artificial Neural Network and Adaptive Segmentation / S. Laurentsyeva, V. Golovko, V. Evstigneev // Proceedings of the Tenth International Conference of Pattern Recognition And Information Processing (PRIP'2009), Minsk, Belarus, 19 – 21 May, – Minsk, 2009 – P. 327–331.

3. Hyvaerinen A., Oja E. Independent component analysis: algorithms and applications // Neural Networks. – 2000. – № 13. - P. 411-430.

4. Bezobrazova S. Forecasting neural networks and chaos theory in the application for epilepsy detection // IX International PhD Workshop OWD'2007: Conference Archives PTETiS. – 2007. – Vol. 23.

УДК 681.3

ОЦЕНКА АНТИКРИЗИСНОЙ УСТОЙЧИВОСТИ ПРОМЫШЛЕННОГО ПРЕДПРИЯТИЯ НА ОСНОВЕ НЕЙРОСЕТЕВОГО МОДЕЛИРОВАНИЯ

Масалитина Н.Н.

*УО «Гомельский государственный технический университет
имени П.О. Сухого», г. Гомель*

С целью исследования закономерностей изменения состояния промышленного предприятия под влиянием кризисных изменений различного происхождения и противостоящих им механизмов было выполнено моделирование системы поддержания устойчивости и антикризисного управления предприятия. В результате получена нейронная сеть, отражающая изменение состояния предприятия в зависимости от степени развития его антикризисных механизмов.

В качестве входных значений нейронной сети используются показатели, характеризующие степень развития отдельных элементов подсистем устойчивости и антикризисного управления, соответствующие выявленным наиболее значимым для белорусских промышленных предприятий кризисным процессам (числу наиболее значимых кризисных изменений отнесены те факторы, по которым зафиксированы согласованные высокие оценки экспертов по показателю частоты или уровня ущерба, либо по обоим показателям (см. табл. 2)).

Обучение нейронной сети проведено на основе выборочной совокупности, содержащей 55 предприятий, обладающих различным уровнем устойчивости к кризисным изменениям (устойчивые; неплатежеспособные; испытывающие сложности, но сохраняющие платежеспособность). Процентное соотношение числа предприятий каждой группы соответствует структуре промышленности Республики Беларусь в отношении указанных характеристик. В состав обучающего множества случайным образом отбирался 91% эталонов, в тестовое – 9%.

При помощи алгоритма обратного распространения ошибки (BP) и алгоритма эластичного распространения ошибки (RP) с использованием различных активационных функций построено 23 нейронные сети (табл. 1). Соотношение элементов по слоям 20-2-1. Полученные нейронные сети построены с помощью различных активационных функций, характеризуются различными требованиями к точности получаемых результатов и различными параметрами обучения).

В результате оценки степени точности полученных нейронных сетей на основе анализа таблиц сопряженности, отражающих степень совпадения фактических и поставленных в процессе обучения и тестирования нейронных сетей диагнозов (рис. 1), отобраны наиболее точные нейронные сети – НС1 и НС2. Для дальнейших исследований выбрана НС2, так как она получена на основе более устойчивого к ошибкам алгоритма обучения (BR).

Таблица 1 – Параметры построенных нейронных сетей

Обозначение	Алгоритм обучения	Активационная функция	Параметр крутизны
НС1	RP	сигмоид	1
НС2	BP	сигмоид	1
НС3	RP	гипертангенс	1
НС4	BP	гипертангенс	0,9
НС5	BP	гипертангенс	0,8
НС6	BP	гипертангенс	0,7
НС7	BP	арктангенс	–

Продолжение таблицы 1

НС8	RP	арктангенс	–
НС9	BP	логарифмическая	–
НС10	RP	логарифмическая	–
НС11	BP	гауссиан	1
НС12	BP	гауссиан	0,9
НС13	BP	гауссиан	0,8
НС14	BP	пороговая	0,7
НС15	RP	пороговая	0,7
НС16	BP	пороговая	0,09
НС17	RP	пороговая	0,09
НС18	RP	пороговая	2
НС19	RP	гауссиан	0,35
НС20	BP	сигмоид	1
НС21	BP	сигмоид	0,8
НС22	BP	сигмоид	0,64
НС23	BP	сигмоид	0,71

Во входном слое сеть НС2 сети содержит 20 величин, соответствующих экономическим показателям, характеризующим степень развития отдельных элементов системы антикризисной устойчивости промышленного предприятия (табл. 2). Скрытый слой состоит из двух нейронов. Выходной слой содержит один нейрон, соответствующий диагнозу предприятия: квазиустойчивое состояние, самостоятельное антикризисное управление, потенциальное банкротство. Активационной функцией данной сети является сигмоид с параметром, равным 1.

$$D = \frac{1}{1 + e^{-U_1}} \quad (1)$$

$$U_1 = v_{11}^2 \frac{1}{1 + e^{-N_1}} + v_{21}^2 \frac{1}{1 + e^{-N_2}} - S^2 \quad (2)$$

$$N_j^k = S_j^k + \sum_{i=1}^n v_{ij}^k I_i, \quad (3)$$

где D – диагноз предприятия; N_j^k – реакция j -го нейрона k -го слоя; v_{ij}^k – вес связи от i -го входа к j -му нейрону k -го слоя; S_j^k – смещение j -го нейрона k -го слоя; I_i – реакция i -го входа.

Анализ степени точности полученной модели позволил установить стопроцентное распознавание эталонов с различными диагнозами (рис. 1).

Оценка устойчивости полученной математической модели к изменению входных параметров позволила установить, что допустимые диапазоны их изменения, обеспечивающие необходимую точность распознавания, являются достаточно широкими: от 75 до 100% промышленных предприятий Республики Беларусь по рассмотренным параметрам соответствуют диапазонам устойчивости модели. Следовательно, построенная модель может быть использована в управлении широким кругом субъектов хозяйствования.

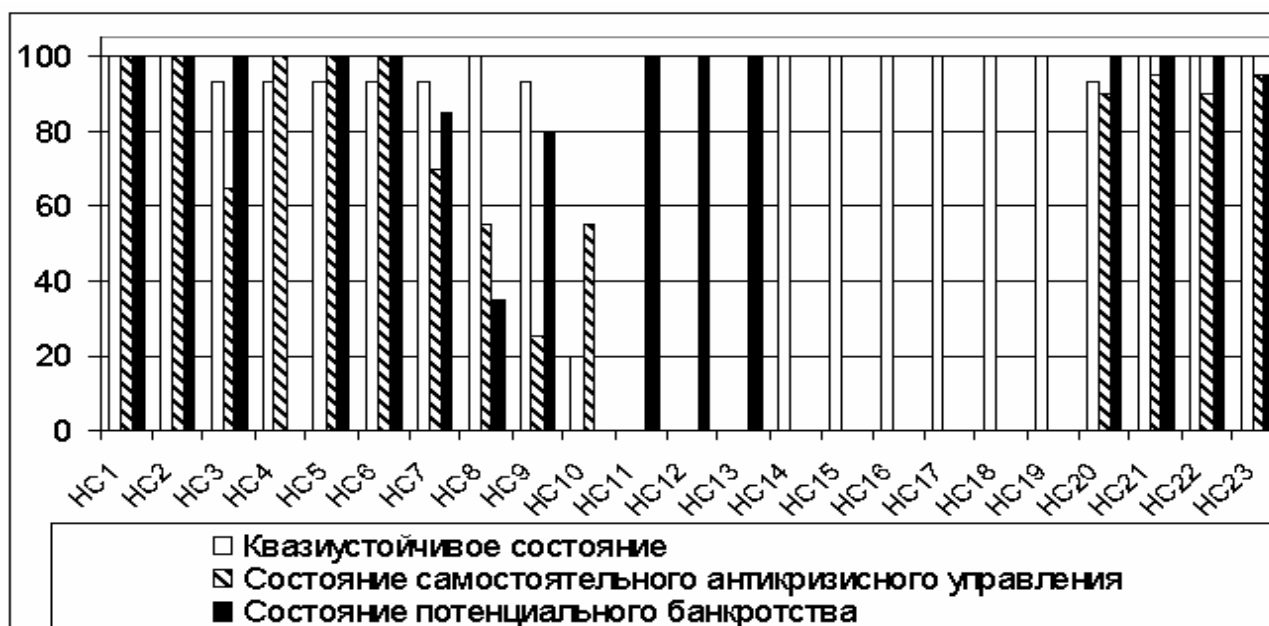


Рисунок 1 – Процент верно распознанных эталонов с различными диагнозами, полученный при оценке точности построенных нейронных сетей

Таблица 2 – Пределы устойчивости разработанной модели системы антикризисной устойчивости промышленного предприятия к изменению входных параметров

Показатель	Мин. значение	Макс. значение
Коэффициент физического износа основных средств, отн. ед.	0	0,904
Коэффициент морального износа основных средств, отн. ед.	0,2	1
Коэффициент выбытия основных средств, отн. ед.	0	0,368
Коэффициент обновления основных средств, отн. ед.	0	4,851
Коэффициент использования оборудования (общий), отн. ед.)	0,21	0,625
Среднее отношение объема запасов сырья и материалов к объему потребления, отн. ед.	1,1	1,2
Просроченная дебиторская задолженность, млн.р.	0	6608,4
Среднее число поставщиков основных видов сырья, материалов, полуфабрикатов, ед.	3	-
Среднее число покупателей по основным видам продукции, ед.	10	-
Процент экспорта в страны СНГ в объеме реализации продукции, %	0	32
Процент экспорта в страны дальнего зарубежья в объеме реализации продукции, %	0	25
Процент производства рыночных новинок, %	-	20
Процент производства товаров со стабильным спросом и прибылью, %	20	-
Процент производства убыточных товаров, %	-	40
Отношение запасов готовой продукции среднемесячному объему реализации, отн. ед.	0,013	681,231
Среднеквадратическое отклонение прогнозных показателей от фактических, отн. ед.	0,22	0,52
Доля затрат на НИОКР в затратах на производство продукции, отн. ед.	0	0,62
Процент производства новой продукции, %	0	39,8
Отношение платежей по страхованию к объему затрат на производство продукции, отн. ед.	0	0,418
Отношение объема страховых и резервных фондов к затратам на производство продукции, отн. ед.	0	1,1

УДК 004

ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ В ЭКОНОМИКЕ

Новик Е.Ю., Семенюк Д.И.*УО «Белорусский государственный экономический университет», г. Минск*

Нейросетевая технология находит применение во многих сферах жизни человека, в том числе – и в экономике. В странах запада это превратилось в высокодоходную индустрию. Многие хедж-фонды, банки, пенсионные фонды, страховые компании покупают у разработчиков пакеты программ, основанные на ИНС. Основная причина – ИНС предоставляют возможность анализа прошлой информации, выявления зависимостей и закономерностей между различными начальными данными. Это ПО позволяет «думать» вместо человека, принимать решения. Также, относительная «свежесть» данной технологии и биологическое основание делает её в глазах большинства обывателей волшебной пиллюлей, магическим граалем. На самом ли деле нейронные сети позволяют механизировать (хотя бы частично) мозг человека и смоделировать мыслительные процессы? В этой статье мы попробуем разобраться в этом (хотя тот факт, что в НС уже вложены огромные средства, и спрос на ПО, основанное на ИНС, всё ещё растёт, звучит позитивно).

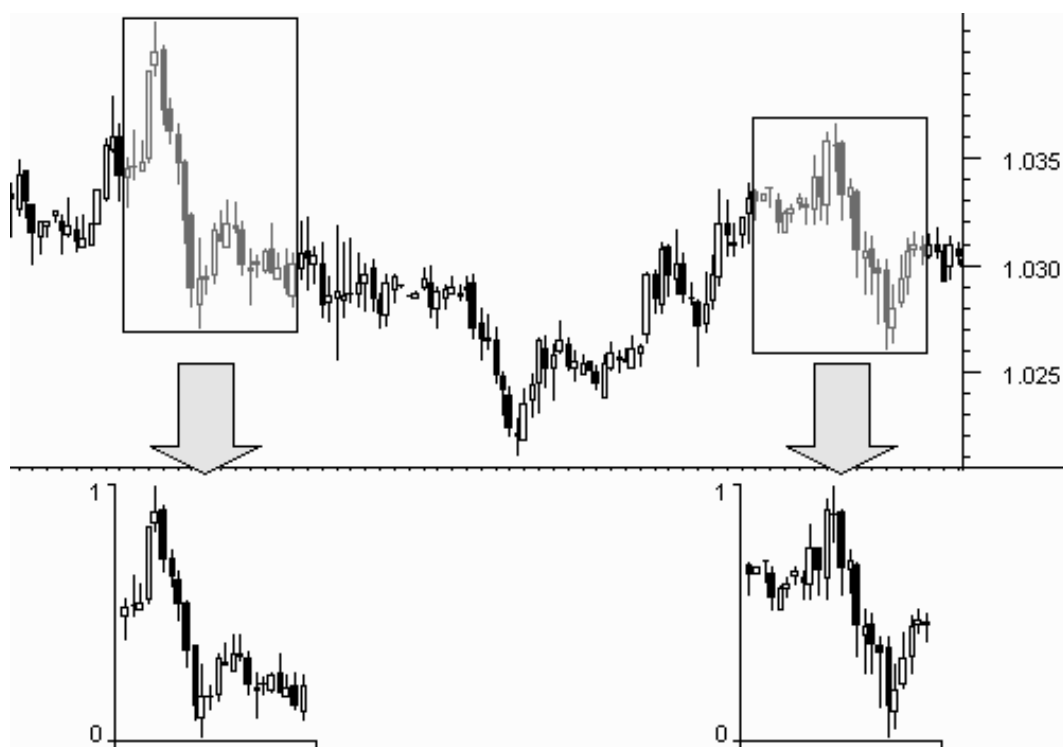
Кора головного мозга человека содержит 10–20 миллиардов нейронов. Один нейрон может иметь связи с 20-ю тысячами других нейронов. Смоделировать человеческий мозг и достигнуть его производительности пока технически невозможно. Еще совсем недавно IBM создало сеть суперкомпьютеров, производительность которых достигла 10% от производительности мышиного мозга. Конечно, моделировать работу всего головного мозга не нужно, достаточно лишь малой части, чтобы решить именно нужную проблему. Но пока и это осуществить очень сложно.

Проблема состоит в том, что при всяческом упрощении ИНС качество (и правдивость) её результата (прогноза) неуклонно падает. Чтобы не быть голословным, мы написали программу на языке MQL4 (встроенный язык программы-терминала торговых сделок MetaTrader 4 фирмы Metaquotes).

У нас есть временной ряд (цена на инструмент за год), набор технических индикаторов. В качестве основы мы выбрали метод окон. Метод окон состоит в том, что на входы нейронной сети подаётся образ, сформированный не за все значения временного ряда (а мы выбрали период длительностью один год), а лишь часть (т.н. окно). На входы нейронной сети мы выбрали значения нижней границы бара (наименьшее значение цены за час), верхней границы (наивысшее значение цены за час), и значение стохастического осциллятора. Глубина погружения – то есть размер окна – 3 часа. Получается, что входных элементов у нас получилось 9. Так как выходной сигнал каждого нейрона лежит в границах отрезка [0;1] (область значений сигмоида), то наиболее логичным и рациональным было бы подавать на входы ИНС значения, лежащие на отрезке [0;1]. Поэтому, предварительно сигнал нужно обработать (нормировать). Для значений котировок это осуществляется по формуле

$$\bar{x} = \frac{x - \text{Min}}{\text{Max} - \text{Min}},$$

где Min – минимальное значение цены в окне, Max – соответственно максимальное, x – первоначальная цена, \bar{x} – нормированное значение цены.



Так как стохастик лежит в границах $[0;100]$, то предварительно мы просто делим значение на 100.

Топология сети следующая: четырёхслойный персептрон (входной и выходной слои не считаются), без обратных связей. Количество весов в ИНС составило более сотни. Диапазон изменения каждого веса мы выбрали совсем небольшой – 20 (-10 ... 10). Шаг – 1. Но всё равно, в идеале мы должны перебрать $9,53e+5340$ комбинаций. Это гигантское число. Даже при использовании генетического алгоритма времени на перебор значений весов (на обучение ИНС) может попросту не хватить. Мы выявили закономерность работы нашей ИНС: для успешного применения программы, скажем, на полсутки вперёд, ИНС требуется обучать на историческом промежутке длительностью два месяца. Другими словами, эта программа работает, и качество довольно высокое (прибыльность примерно 320% годовых при просадке не более 5%). Но обучение сети по времени занимает больше, чем прогнозный горизонт (у нас – половина суток). Таким образом, приходится упрощать ИНС, сокращать количество входных параметров, слоёв ИНС. Однако главное, что ИНС могут применяться на практике.

На волне либерализации экономики РФ возможна ситуация, что, к примеру, вырастет число частных пенсионных фондов. Эти фонды будут заинтересованы в рациональном управлении свободными средствами. Посему, будут вкладываться деньги в развитие нанотехнологий, а также в создание мощнейших суперкомпьютеров, способных качественно и быстро обучать ИНС.

Литература

1. Уоссермен, Ф. Нейрокомпьютерная техника: Теория и практика.
2. Розенблатт, Ф. Принципы нейродинамики. Перцептроны и теория механизмов мозга. – М., 1965.
3. Барский, А.Б. Нейронные сети: распознавание, управление, принятие решений.

УДК 004.032.26:330.131.7

ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ ДЛЯ АНАЛИЗА СТАБИЛЬНОСТИ ЭКОНОМИЧЕСКОЙ СРЕДЫ

Перетрухина В.В., Лобик В.Г., Путицкая И.И.

УО «Белорусский государственный экономический университет», г. Минск

Уже сегодня искусственные нейронные сети используются в разнообразных областях: распознавание текста и речи, семантический поиск, экспертные системы и системы поддержки принятия решений, предсказание курсов акций, системы безопасности, предсказание банкротства предприятия, анализ текстов и другие.

Цель работы: изучить применение нейронных сетей для формализации процессов управления, рассмотреть непосредственные примеры. Для ее достижения были решены следующие задачи: выполнение обзора применения нейронных сетей; анализ применения нейронных сетей на конкретном примере предсказания банкротства предприятия.

Если в проблеме рейтингования задача нейросети - воспроизведение мнения экспертов о надежности корпорации, то нейросетевое предсказание банкротств основано на статистической обработке конкретных примеров банкротств. В такой постановке задача нейросети - самой стать экспертом, определяющим финансовую стабильность корпорации, основываясь исключительно на объективной информации - показателях финансовой отчетности. Обычно от нейросети требуется оценить вероятность банкротства через определенный промежуток времени (например, через год или через два года) по доступной на данный момент финансовой отчетности. В качестве входов используют финансовые индикаторы - отношения балансовых статей, наиболее полно отражающие определенные стороны финансового положения фирмы [2]. При желании можно повысить "подозрительность" нейросети, обеспечив точность выявления банкротов вплоть до 99% - за счет снижения требований к ошибкам второго рода (классификации нормальной фирмы как банкрота). Банкротства можно уверенно предсказывать за несколько лет до их фактического наступления, причем точность предсказания за два года практически не отличается от точности предсказания за год. Таким образом, неявные сигналы неблагополучия присутствуют в финансовой отчетности фирмы задолго до ее краха [1].

Для построения нейронной сети была разработана ее топология, определён механизм обучения и процедура тестирования. Кроме того, для обучения определены входные данные – выборка компаний с достоверной финансовой отчетностью и рассчитанные на ее основе коэффициенты. Наиболее привлекательным в данном случае представляется выбор трехслойного персептрона и алгоритма обратного распространения в качестве обучающего.

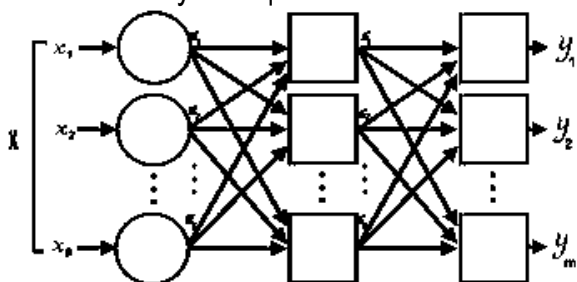


Рисунок 1 – Трехслойный персептрон

Таблица 1 – Конфигурация разработанной сети

Общие параметры	Параметры скрытого слоя	Параметры выходного слоя
Количество узлов во входном слое: 10	Коэффициент обучения: 0.25	Коэффициент обучения: 0.01
Число узлов в скрытом слое: 10	Коэффициент инерции: 0.6	Коэффициент инерции: 0
Число узлов в выходном слое: 1	Затухание: 0	Затухание: 0
Алгоритм обучения: обратное распр.		
Функция активации: сигмоидальная		

Точность подбора этих параметров определяет способность сети к обучению. Значение единственного узла выходного слоя – показатель финансовой состоятельности предприятия. Единичное значение соответствует состоянию банкротства, нулевое – полному финансовому благополучию. В качестве активационной функции выбрана сигмоидальная:

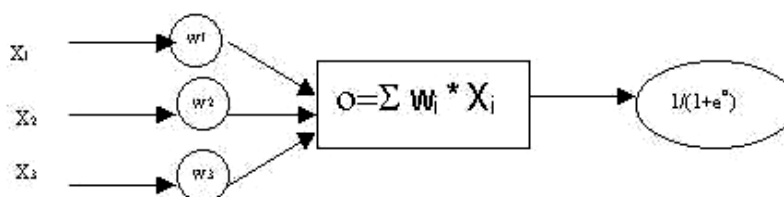


Рисунок 2 – Изображение сигмоидальной функции активации

В рассматриваемой задаче прогнозирования банкротства наряду с выбором модели и конфигурации важным моментом является сбор статистики по предприятиям. В качестве исходных данных взята финансовая отчетность 100 компаний (данные о 50 компаниях выступали в качестве данных для обучения, для тестирования модели 100 компаний были разбиты на 2 множества – обучающее множество и тестирующее множество). Деятельность организаций рассматривалась в период с 2004 по 2008 год, что позволило включить в модель влияние различных внешних факторов. Соотношение обанкротившихся и финансово устойчивых предприятий в выборке было оптимальным: 50/50, что является исключительно важным для корректности обучения нейронной сети. Если бы в исходных данных преобладали благополучные предприятия, сеть могла бы “подумать”, что высокую вероятность правильного прогноза легко получить простым предположением о том, что каждое предприятие жизнеспособно в рассматриваемом периоде [3], [2]. Для оценки правильности модели была собрана статистика по фирмам за 3 года до их банкротства и нейронная сеть обучена на данных финансовой отчетности каждого года. Целью теста было увидеть насколько хорошо модель прогнозирует состояние организации через год, через 2 года и через 3 года. На основании финансовой информации об оставшихся 50 предприятиях было сформировано тестирующее покрывающее множество, которое использовалось для оценки “обобщающей способности” модели. Были получены следующие результаты.

Таблица 2 – Корректность прогноза

	Обучающее множество			Тестирующее множество		
	Общий показ.	Для благоп. компаний	Для несост.	Общий показ.	Для благоп. компаний	Для несостоят.
Год 0	98%	100%	96%	88%	81%	96%
Год 1	88%	88%	88%	69%	58%	81%
Год 2	90%	100%	81%	62%	65%	58%

Как и ожидалось, модель показала высокие результаты для обучающего множества и значительно более низкие для тестирующего. Особенно интересным в анализе результатов прогнозирования для компаний из тестирующего множества представляется факт высокой вероятности предсказания банкротства в первые 2 года. Требуется дальнейшее исследование для определения причин этого явления [1].

В данной работе было представлено, каким образом нейронные сети способны помочь людям в генерации знаний, которые основывались бы на всех первоначальных данных. Исследования в области нейронных сетей в основном достаточно наглядны. По сравнению с другими вычислительными методами в статистике и науке они имеют значительные преимущества. Но они никогда не смогут полностью заменить людей в процессе решения задачи. Нейронные сети должны использоваться для обобщения данных, а не для определения, атрибуты и критерии которого весьма важны при сборе данных. Нейронные сети адаптивны по своей природе, они могут подражать решению проблемы человеком, но они не сообщат, какой из критериев решения задачи должен быть принят во внимание перед сбором данных. Также были рассмотрены различные методики анализа и оценки рисков. Нейронные сети не являются панацеей, во многих случаях более эффективным будет использование традиционных методов статистики. Несмотря на это, во многих областях управления рисками нейронные сети зарекомендовали себя как более рациональное решение.

Литература

1. Короткий, С. Нейронные сети: алгоритм обратного распространения / С. Короткий. – 231 с.
2. Блум, Ф. Мозг, разум и поведение / Ф. Блум, А. Лейзерсон, Л. Хофстедтер. – М.: Мир, 1988. – 198 с.
3. Уоссермен, Ф. Нейрокомпьютерная техника: Теория и практика / Ф. Уоссермен. – Мир, 1992. – 154 с.

УДК 004.8.032.26

НЕЙРОСЕТЕВОЙ ПОДХОД К АРХИВАЦИИ ФАЙЛОВ

Шевеленков В. В.

УО «Брестский государственный технический университет», г. Брест

Введение

На сегодняшний день известно множество методов архивации файлов. Большинство персональных компьютеров используют в повседневной работе такие приложения, как WinRAR и WinZip, которые позволяют пользователям сжимать файлы в меньшие по размеру архивы. Особенно сжатые файлы удобно передавать и скачивать с удаленных ПК и распределенных систем.

Каждый известный компрессионный метод характеризуется такими параметрами, как скорость и максимально достижимый коэффициент сжатия. Целью написания данной работы является создание алгоритма сжатия файлов с коэффициентом, недостижимым распространенными на сегодняшний день приложениями.

Алгоритм, описанный ниже, основан на использовании алгоритма обратного распространения ошибки с использованием адаптивного шага обучения и некоторыми модификациями, для достижения необходимых результатов. Предложенный метод сжатия может быть применен к файлам любого типа и позволит достичь нового уровня информационной безопасности.

1. Общие принципы архивации

Первые работы, касающиеся сжатия информации были опубликованы Клодом Шенноном и Хартли. Они не были конструктивными, поскольку не содержали точного описания алгоритмов сжатия, а лишь описывали такую возможность. Практическое применение стало возможным лишь с появлением алгоритмов Шеннона-Фано (Shannon-Fano) и Хаффмана (Huffman), которые заключались в использования для каждого значения байта кода различной длины.

Наиболее известные архиваторы используют в процессе сжатия избыточность информации, арифметическое кодирование, энтропийное кодирование и др. Каждый из существующих методов имеет как сильные, так и слабые стороны. Например, методы, использующие в своей работе избыточность информации, показывают хорошую скорость работы, но при этом обеспечивают средний уровень сжатия (0,7-0,8 от исходного размера файла), а методы, использующие математическое кодирование, наоборот, работают медленнее, но и коэффициент сжатия у них выше.

Методика, предложенная в этой работе, основана на использовании алгоритма обратного распространения ошибки. Она показывает хорошие результаты по коэффициенту сжатия, но также имеет и весомые недостатки, выражающиеся в низкой скорости архивации. Причина написания данной работы – достижение коэффициента сжатия, равного 10 и выше.

2. Методика архивации

Процедура трансформации информации состоит из следующих шагов:

- нормализация информации;
- сжатие информации;
- сохранение информации.

В данной работе автор предлагает методику сжатия файлов любого типа, так же как и большинство используемых архивационных приложений.

2.1. Нормализация информации

Содержимое каждого файла представляется набором ASCII символов. Код каждого символа представляется числом в диапазоне от 0 до 255. Алгоритм обратного распространения использует при обучении и прогнозировании числа в диапазоне от 0 до 1, таким образом, необходимо нормализовать информацию, т.е. привести исходные данные к требуемому диапазону.

2.2. Сжатие информации

Для сжатия информации мы используем многослойную нейронную сеть, обучаемую при помощи алгоритма обратного распространения ошибки с адаптивным шагом обучения и варьированием количества нейронов в скрытых слоях. На рис.1 показана архитектура данной нейронной сети. [1]

Причем $X_1...X_N$ -входы нейронной сети, именно на эти входы будет подаваться нормализованная информация. W_1 , W_2 и W_3 – матрицы весовых коэффициентов. Y -выходной сигнал нейронной сети.

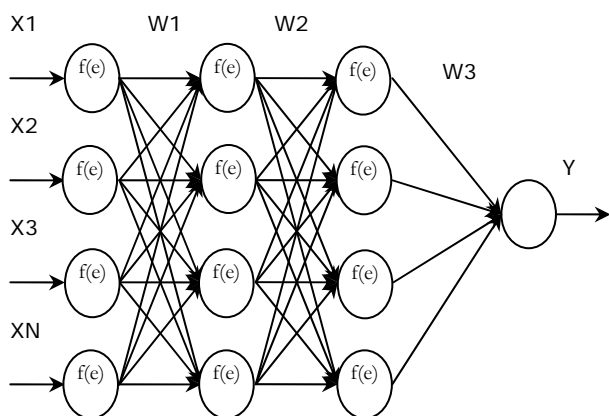


Рисунок 1 – Архитектура нейронной сети

Процедура обучения или тренировки нейронной сети – это итерационный процесс, который может быть разделен на следующие шаги:

- Каждый шаг обучения начинается с подачи входных образов из обучающей выборки. Для каждого образа вычисляются выходные значения для каждого нейрона скрытых слоев сети и выходного нейронного элемента.

- На следующем шаге алгоритма выходной сигнал сети сравнивается с желаемым (эталонным), который выбирается следующим, после поданного входного образа. Разница является ошибкой выходного нейрона, и основная идея методики заключается в прогнозировании требуемой информации, т.е. в получении значения выходного сигнала нейронной сети, равного желаемому последующему из обучающей выборки. Ошибки нейронов скрытых слоев невозможно вычислить, т.к. выходные значения этих нейронов неизвестны. Только в середине 80-х годов появился алгоритм обратного распространения ошибки, который заключался в распространении ошибки выходного нейрона на все нейроны скрытых слоев. Весовые коэффициенты W , которые используются для обратного распространения ошибки, равны весовым коэффициентам при прямом распространении информации.

- После этого необходимо скорректировать весовые элементы и пороговые значения для каждого нейронного элемента.

- Также мы изменяем шаг обучения, который влияет на скорость обучения сети.

- При достижении желаемой ошибки обучения мы можем использовать нашу сеть для прогнозирования.

Таким образом, обладая информацией о значениях весовых коэффициентов, первых 10 элементов, пороговых элементов и некоторых других параметрах, мы можем прогнозировать нашу обучающую выборку с высокой точностью, достигающей значений порядка 0,0001 и выше [2].

2.3. Сохранение информации

Обученная нейронная сеть представляет собой определенное число параметров, необходимых для сохранения: весовые коэффициенты (210 при архитектуре 10-10-1), стартовый набор эталонных значений (10) и т.д. Для восстановления архитектуры сети нужно сохранить эти данные в файле, требующем сжатия. Последовательность сохранения информации следующая:

- Параметры нейронной сети – количество слоев, количество элементов в каждом слое;
- Весовые коэффициенты;
- 10 первых элементов обучающей выборки;
- Количество прогнозируемых элементов.

После сохранения данных числовых вещественных значений процесс сжатия можно считать законченным.

3. Модификация нейронной сети

Изложенный выше алгоритм успешно архивирует большинство файлов. На некоторых из них нейронная сеть обучается довольно быстро, т.е. скорость высока, однако некоторые файлы не поддаются сжатию, т.е. нейронная сеть на обучающей выборке, полученной из этих файлов, не может в процессе обучения достичь требуемой среднеквадратичной ошибки. Для решения данной проблемы автором была разработана методика сжатия почти каждого информационно массива (файла). Если нейронная сеть не может в процессе обучения достичь желаемой ошибки, тогда алгоритм изменяет параметры нейронной сети, такие как количество нейронов в скрытых слоях и количество слоев сети. Если эти модификации не помогают достичь минимальной ошибки, нам остается дообучать нейронную сеть в процессе прогнозирования, включением некоторого числа эталонных элементов в спрогнозированный ряд.

Таким образом, варьируя параметры нейронной сети и дообучая сеть в процессе прогнозирования, мы можем достичь высокой точности тренировки сети.

4. Экспериментальные результаты

На рисунке 2 мы видим архивационную программу, созданную автором по алгоритму, предложенному выше. Она позволяет выбирать файлы, обучать сеть, сохранять сжатые файлы. Также представляется информация о текущем состоянии нейронной сети, количестве нейронов в слоях и текущая ошибка. Если программа не может достичь требуемой ошибки, она автоматически производит модификацию сети.

Программа тестировалась на файлах различных типов и результаты сравнились со стандартным приложением ПК WinRAR. При сжатии некоторых типов файлов (*.jpg, *.mp3, *.Rar) коэффициент сжатия достигал 5-9, причем приложение WinRAR эти файлы сжать не могло, и их размер после применения только увеличивался. Недостатком данного алгоритма является низкая скорость сжатия, равная 10 минутам при файле размером 10 Мб. Однако основная цель работы – достижение высоких коэффициентов сжатия без учета скорости. Простейшее решение проблемы избыточности времени – распараллеливание алгоритма и использование другой архитектуры нейронной сети.

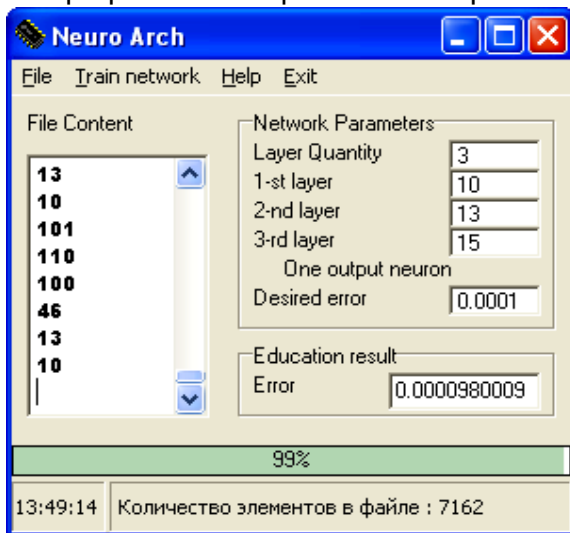


Рисунок 2 – Архиватор

Заключение

В данной работе представлена новая методика архивации файлов. Результаты экспериментов показывают положительный результат применения данного алгоритма к файлам многих типов. Более того, данный алгоритм имеет множество возможностей для усовершенствования. Конечно, алгоритм не лишен недостатков, но в следующих работах автор попытается от них избавиться.

Литература

1. Rumelhart D., Hinton G., Williams R. Learning representation by backpropagation errors// Nature.-1986.-N323.-P.533-536.
2. Golovko V., Savitsky J., Gladischuk. Predicting Neural Net// Proceedings Int. Conf. CM NDT-D5.-Berlin:DGZfP.-1995.-P348-353.

РАСПОЗНАВАНИЕ ОБРАЗОВ И АНАЛИЗ ИЗОБРАЖЕНИЙ

УДК 004.8.032.26

ИДЕНТИФИКАЦИЯ ЧЕЛОВЕКА ПО ФОТОПОРТРЕТУ С ПРИМЕНЕНИЕМ НЕЙРОСЕТЕВЫХ МЕТОДОВ

Войцехович Г. Ю.

УО «Брестский государственный технический университет», г. Брест

ВВЕДЕНИЕ. Каждый человек обладает рядом уникальных биометрических характеристик, таких как изображение сетчатки глаза, узор радужной оболочки, отпечатки пальцев, голос, лицо и прочие. Часть из этих характеристик успешно используется для идентификации человека в криминалистике или при решении задач контроля доступа [1]. Методы, основанные на применении биометрических параметров, имеют множество преимуществ по сравнению с традиционными, такими как ключ, пароль и др., которые можно подделать, забыть или потерять. Однако имеется и ряд недостатков. Один из них заключается в том, что сравнение полученных биометрических параметров с имеющимися в базе данных является весьма сложной и трудоемкой задачей. Вторым недостатком – сложность собственно получения характеристик. Идентификация человека по изображению лица в этом смысле предпочтительнее, чем методы, основанные на других биометрических характеристиках: системы контроля доступа по изображению лица состоят из обычной фото- или видеокамеры и персонального компьютера. Однако поиск человека, изображенного на фотографии, в базе данных – первостепенная задача, решаемая при создании системы контроля доступа, основанной на анализе изображений лиц. До сих пор эта задача остается не решенной на 100%, и, ввиду ее сложности, можно с уверенностью говорить о том, что в ближайшее время она будет оставаться весьма актуальной.

ПРЕИМУЩЕСТВА НЕЙРОННЫХ СЕТЕЙ. Одним из способов частичного решения проблемы поиска человека по изображению лица в имеющейся базе данных является использование нейронных сетей. Такой подход, так же как и абсолютно все существующие методы, не дает 100%-ной гарантии правильной идентификации человека по его фотографии. Тем не менее, нейронные сети обеспечивают очень высокие результаты при распознавании. Благодаря их обобщающим способностям, нейросетевые системы распознавания способны корректно выдавать результаты даже при наличии возрастных изменений, эмоций, очков и пр., что является наиболее сложной задачей при идентификации человека по изображению лица. Кроме того, обученная сеть мгновенно выдает результаты при подаче на нее изображения, что позволяет использовать такие системы в режиме реального времени, например, для поиска людей, находящихся в розыске, с использованием камер наблюдения в местах массового скопления. Подобные системы могли бы найти применение также на таможенных и иных пунктах пропуска (в том числе автоматических), где по изображению человека, полученному в данный момент с фото- или видеокамеры, осуществлялся бы поиск в предварительно сформированной базе данных. Использование подобных систем может значительно облегчить работу сотрудников таможенных пунктов пропуска и повысить ее качество в связи с устранением влияния психологического состояния (усталости и других факторов) сотрудника на результаты его работы. Кроме того, использование таких систем может полностью заменить человека при контроле доступа к какому-либо объекту.

ОБЩАЯ СТРУКТУРА СИСТЕМЫ РАСПОЗНАВАНИЯ. Была разработана система распознавания лиц, основанная на применении нейронных сетей, реализация которой проводилась в среде Borland C++ Builder с использованием библиотеки Open Computer Vision.

На первом этапе производится детектирование лица на фотографии и копирование выделенной части для ее последующей обработки.

На втором шаге осуществляется выделение ключевых точек лица, таких как центры зрачков, нос, рот, подбородок. Для случая неточного определения программой расположения ключевых точек предусмотрена возможность ручной корректировки. Ошибки программы, как правило, возникают при недостаточной контрастности изображений или в случае перекрытия части лица другими объектами. Далее, используя координаты зрачков, производится масштабирование изображения и поворот относительно центра для выравнивания по горизонтали. Затем вычисляются расстояния между выделенными ключевыми точками.

На третьем этапе изображение подвергается предварительной обработке, которая включает в себя перевод изображения в градации серого, приведение среднего значения яркости к единому значению и расширение гистограммы для покрытия всего диапазона от 0 до 255. Общая структура системы распознавания лиц представлена на рис. 1.

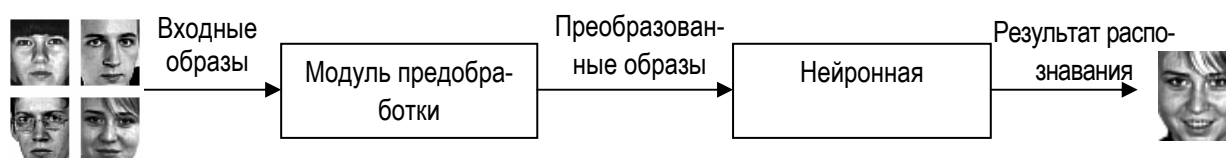


Рисунок 1 – Общая структура системы распознавания образов

Предобработка необходима для устранения влияния условий съемки на результаты работы нейронной сети. В случае непроведения обработки изображений, сеть в большей степени осуществляет сравнение не по биологическим параметрам, а по яркости изображений, углу поворота и пр., что, естественно, приводит к некорректным результатам распознавания. В связи с этим, необходимо отметить, что для достижения наилучших результатов работы системы необходимо использовать фотографии только одного ракурса. Сеть чувствительна к изменениям ракурса и с большой степенью вероятности может выдавать ошибочные результаты при значительном отличии углов съемки используемого в обучающей выборке и подаваемого для распознавания изображений. Для устранения этого ограничения можно помещать в обучающую выборку для каждого человека несколько фотографий с различными ракурсами [2].

Для поиска человека в базе данных применяется нейронная сеть. В силу того, что нейронная сеть может запомнить ограниченное число лиц, поиск человека в базе данных производится в 2 этапа:

1) После проведения предобработки, расстояния между ключевыми точками подаются на многослойный перцептрон, который выделяет ряд наиболее похожих лиц. Сравнение по 5 точкам (9 расстояний) не может дать точный результат, поэтому для определения конкретного человека все отобранные изображения будут подаваться на другую нейронную сеть. Такая методика позволяет расширить возможности системы, т.е. увеличивает максимально возможное число запоминаемых лиц.

2) На втором этапе изображение подается на входы нейронной сети полностью. В качестве основы системы распознавания была выбрана многослойная нейронная сеть,

или многослойный персептрон, содержащий 1 скрытый слой [3]. Количество входных нейронных элементов определяется размерностью подаваемых изображений. Мы использовали изображения размером 40x45 пикселей, что соответствует 1800 входным элементам. Число нейронов скрытого слоя определялось по результатам проведения экспериментов, которые показали, что оптимальным значением является число от 35 до 50. Количество нейронных элементов выходного слоя соответствует числу отобранных первой нейронной сетью человек.

Для уменьшения времени обучения многослойного персептрона был реализован алгоритм выделения главных компонент с использованием рециркуляционной нейронной сети [4], а также, как альтернативный вариант, – алгоритм дискретного косинусного преобразования для получения коэффициентов низкочастотной составляющей, содержащей наибольшее количество информации.

РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ. Эксперименты проводились с использованием изображений лиц из общедоступных баз данных BioID, ORL и YaleFaces в сочетании с собственными фотографиями, полученными с цифрового фотоаппарата. Таким образом, тестирование системы осуществлялось на изображениях, полученных в условиях, максимально приближенных к условиям потенциального применения системы, т.е. использовались фотографии из разных баз данных, полученные с различных камер, в различных условиях освещения. Кроме того, люди фотографировались в различных эмоциональных состояниях, а также с очками или без них.

Результаты экспериментов представлены в таблице 1.

Таблица 1 – Результаты экспериментов

	Время обучения и процент распознавания на незнакомых образах при использовании	
	20 образов	50 образов
Многослойный персептрон	2 мин; 100%	29 мин; 94%
Рециркуляционная сеть + многослойный персептрон	1,5 мин; 100%	22 мин; 90%
Косинусное преобразование + многослойный персептрон	1,5 мин; 100%	8мин; 84%

ЗАКЛЮЧЕНИЕ. В целом, можно сделать вывод, что представленный метод распознавания лиц весьма эффективен. Необходимо лишь правильно подобрать и обработать входные образы. В таком случае будет обеспечена устойчивая и адекватная работа системы. В связи с этим наиболее перспективным направлением в дальнейшей работе по данной теме представляется более тщательная обработка изображений.

Кроме того, корректность работы этой системы обратно пропорциональна количеству лиц в обучающей выборке. Для некоторых задач число запоминаемых системой лиц, возможно, окажется недостаточным. В таких случаях необходимо использовать ансамбль многослойных нейронных сетей, организованный следующим образом: каждая сеть обучается на определенном наборе лиц, распознаваемое изображение подается на входы каждой сети, а результаты сравниваются между собой либо по значениям выходных нейронов, либо с использованием такой же многослойной сети. Описанный метод позволит расширить число запоминаемых системой лиц до любой необходимой величины.

Литература

1. Самаль, Д.И. Подходы и методы распознавания людей по фотопортретам / Д.И. Смаль, В.В. Старовойтов. – Минск: ИТК НАНБ, 1998. – 54 с.
2. Madani, K. Modular and self-organizing connectionist systems: toward higher level intelligent functions / K. Madani // International Scientific Journal of Computing. – 2006. – Vol. 5, Issue 2. – P. 6-17.
3. Головкин, В.А. Нейронные сети: обучение, организация и применение. Кн. 4: учеб. пособие для вузов / Общая ред. А.И. Галушкина. – М.: ИПРЖР, 2001. – 256 с.
4. Oja, E. Principal components, minor components and linear networks / E. Oja // Neural Networks. - 1992. - Vol. 5. - P. 927-935.

УДК 004.93

**ПРИНЦИПЫ РАБОТЫ СИСТЕМЫ ДЕТЕКТИРОВАНИЯ АВТОНОМНОГО
МОБИЛЬНОГО РОБОТА И ПРЕПЯТСТВИЙ ДЛЯ ЕГО ДВИЖЕНИЯ***Дунец И.П.**УО «Брестский государственный технический университет», г. Брест*

В связи с развитием робототехники в автоматизированное производство внедряется все большее количество мобильных роботизированных платформ, выполняющих самые разнообразные действия: от перевозки грузов с одного конвейера на другой до применения в качестве автоматизированного шасси укомплектованного различным оборудованием. Можно предположить, что мобильные роботизированные платформы полностью заменят конвейерные ленты за счет своей высокой мобильности и простоты модификации процессов производства.

Но до сих пор не является полностью решенной одна из наиболее актуальных проблем автоматизации процесса управления мобильными платформами – безопасность передвижения. Эта проблема включает в себя задачу определения положения автономного мобильного робота в пространстве и препятствий для его движения.

На данный момент такой подход как машинное зрение получил наиболее широкое распространение для автоматизации систем управления мобильными роботами. Основная причина его эффективности заключается в высокой информативности получаемых данных.

В данной статье рассматривается система машинного зрения, предназначенная для определения положения и ориентации мобильного робота на кадре из видеопотока, а также детектировании препятствий для его движения с последующей их классификацией на статические и динамические.

Любая система машинного зрения условно делится на аппаратную и программную часть. Аппаратная часть предназначена для получения информации с различных сенсорных устройств, программная часть реализует алгоритмы обработки полученных данных [1].

В данной системе аппаратная часть представлена цифровой видеокамерой, подключенной к персональному компьютеру, выполняющему роль вычислительного центра. Цифровая видеокамера крепится на потолке помещения и направляется на пол, это позволяет получить наиболее полную информацию о положении робота и окружающей его среды.

Программная часть системы состоит из модулей захвата видеоизображения, обработки кадров изображения и связи с мобильным роботом. Для реализации необходимого функционала программной части используется библиотека компьютерного зрения OpenCV v1.0.

Работа системы представляет собой цикл, состоящий из следующих фаз:

- 1) захват кадра из видеопотока, получаемого цифровой видеокамерой;
- 2) выделение объектов на кадре;
- 3) идентификация выделенных объектов;
- 4) расчет параметров движения объектов;
- 5) передача данных на модуль управления мобильным роботом.

Система также имеет стадию предварительной инициализации, состоящей из начального анализа сцены (под сценой понимается область пола, попадающая в поле зрения камеры, включающая все расположенные на ней объекты). Кадр, изображенный на рис. 1, содержит следующие классы объектов, пронумерованные соответствующими цифрами:

- 1) мобильный робот;
- 2) статическое препятствие;
- 3) динамическое препятствие.

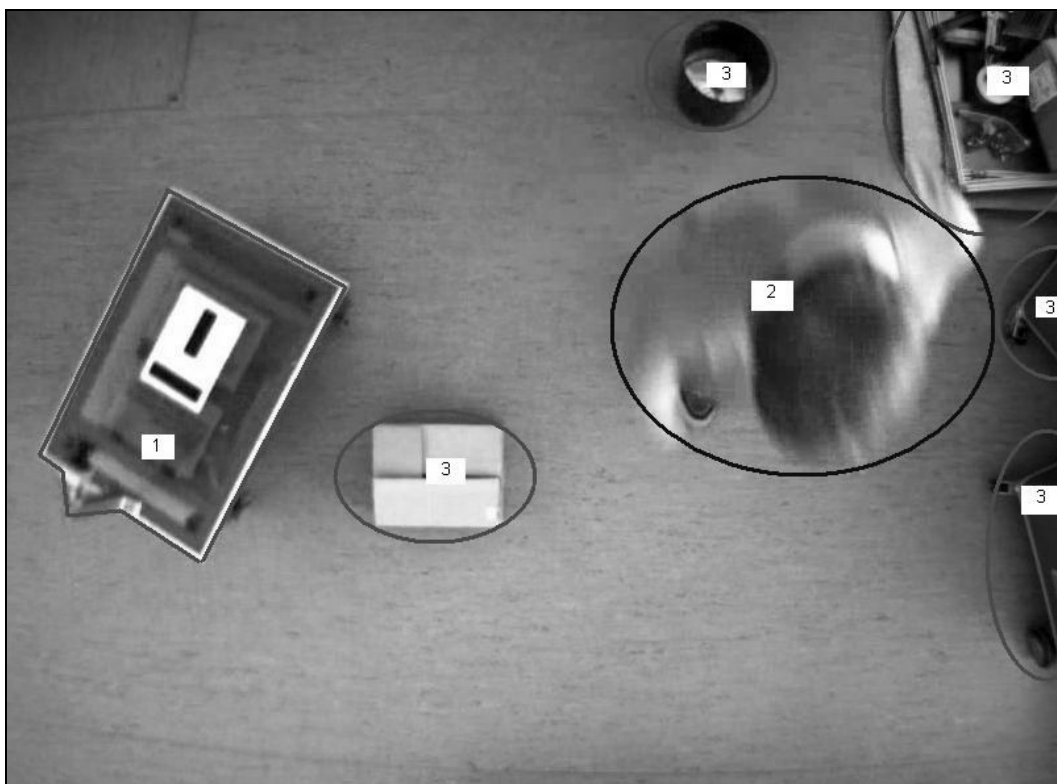


Рисунок 1 – Пример кадра, содержащего изображение сцены

На этапе инициализации происходит сравнение первого полученного кадра с эталонным (изображение сцены без каких-либо объектов) для детектирования объектов, находящихся на сцене. При этом выполняется фильтрация теней, отбрасываемых объектами[2]. Для этого производится конвертация изображений из цветового пространства RGB в YUV. Производится попиксельный подсчет разницы между каналами U и V, отвечающими за цветность (Y отвечает за интенсивность), эталонного и полученного изображений(1). Выделенные объекты идентифицируются как статические препятствия.

$$|U_i^0 - U_i^1| + |V_i^0 - V_i^1| \leq \alpha, \quad (1)$$

где U_i^0 , V_i^0 – значение составляющей U и V, i-го пикселя исходного изображения; U_i^1 , V_i^1 – значение составляющей U и V, i-го пикселя эталонного изображения; α – пороговое значение, выбираемое эмпирически.

На этапе выделения объектов производится сравнение текущего кадра с предыдущим. На основе полученной разности производится выделение контуров с помощью метода библиотеки OpenCV cvFindContours[3], с последующим их сохранением для классификации и анализа в пунктах 3 и 4.

Этап идентификации включает в себя соотнесения объектов соответствующим классам. Классификация объектов проводится по наличию определенных признаков выявленных объектов, таких как способность к перемещению или наличие метки, указывающей на то, что объект принадлежит к классу «мобильный робот». Всего выделяется три класса: статическое препятствие, динамическое препятствие и мобильный робот. К классу «статическое препятствие» относятся все объекты, выделенные на этапе инициализации. В случае обнаружения перемещения эти объекты переходят в класс динамических препятствий. Под классом «динамическое препятствие» понимается любой объект, находящийся на сцене и способный к передвижению (человек; робот, не находящийся под управлением; автопогрузчик и т.д.). На объект класса «робот» наносится метка, используемая для распознавания управляемого мобильного робота, изображенный на рис. 2. Также данная метка позволяет определить пространственную ориентацию управляемого мобильного робота.



Рисунок 2 – Паттерн для распознавания управляемого робота

Поиск и анализ метки происходит на этапе идентификации. Под анализом понимается вычисление ориентации метки, что позволяет определить ориентацию мобильного робота.

Также на этапе идентификации происходит соотнесение имеющихся объектов, с выделенными объектами на предыдущем этапе, добавление новых или удаление объектов, покинувших сцену. Соотнесение происходит путем оценки положения геометрического центра масс объектов в текущий и предыдущий момент времени.

На этапе расчета параметров движения происходит сохранение пути пройденного объектами, расчет таких параметров, как геометрический центр масс, угловая и линейная скорости (не проводится для статических препятствий).

Данные о состоянии сцены, положение и ориентация управляемого мобильного робота передаются на управляющей модуль автономного мобильного робота на этапе 5. Передача данных осуществляется по протоколу TCP/IP.

В данной статье описывается один из подходов решения задачи автоматизации управления автономным мобильным роботом. Данный метод имеет преимущество в виде высокой информативности получаемых данных, но в то же время не позволяет детектировать объекты малого размера (менее 3% от разрешения изображения). По причине многочисленных шумов, возникающих при получении и обработке изображения. Для решения этой проблемы возможно расширение системы путем комплектации цифровой видеокамерой мобильного робота и создании программного обеспечения, способного проводить дополнительный анализ среды, окружающей робота.

На данный момент разрабатывается программный модуль прогнозирования, позволяющий определять риск столкновения управляемого мобильного робота с объектами.

Литература.

1. Воротников, С.А. Информационные устройства робототехнических систем / С.А. Воротников. – М.: Издательство МГТУ имени Н.Э. Баумана, 2005.
2. Гаганов, В. Сегментация движущихся объектов в видео потоке / В. Гаганов, А. Конушин // Научно-популярный online-журнал Графика и Мультимедиа [Электронный ресурс]. – 2004. – Режим доступа к журн.: <http://cgm.computergraphics.ru/content/view/67>, свободный.
3. Gary Bradski and Adrian Kaehler. Learning OpenCV. – Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, P. 234-240, 2008.

УДК 621.372.542, 625.13.08

ПРИМЕНЕНИЕ АЛГОРИТМОВ ЦИФРОВОЙ ОБРАБОТКИ ДИНАМИЧЕСКИХ ИЗОБРАЖЕНИЙ ДЛЯ ДЕТЕКТИРОВАНИЯ ТРАНСПОРТНЫХ СРЕДСТВ И УПРАВЛЕНИЯ ДОРОЖНЫМ ДВИЖЕНИЕМ

Касьяник В.В., Шуть В.Н.

УО «Брестский государственный технический университет», г. Брест

Исследование и разработка проблем безопасности и автоматизации дорожного движения очень актуальна на сегодняшний день в Республике Беларусь.

Автоматизированные системы управления дорожным движением (АСУД) являются сложным комплексом технических средств и решений, обеспечивающим максимально возможные объемы сокращения транспортных задержек, повышение скоростей движения, увеличение пропускной способности дорог и уровня безопасности движения [1].

Перспективным направлением развития дорожного движения является улучшение качества использования проезжей части с помощью увеличения её загрузки и активного реагирования на изменение трафика дорожного движения, распределения потоков движения транспорта, а также мониторинга и сбора статистики для последующего анализа. Существует большое количество вариантов решения дорожной проблемы, однако наиболее эффективный метод решения данной проблемы - применение интеллектуальных систем, позволяющих автоматически оптимизировать дорожное движение. Перекрестки с использованием таких систем называются гибкими и характеризуются возможностью адаптации к изменениям параметров движения.

Гибкие системы, построенные на основе адаптивных алгоритмов, позволяют наиболее эффективно снижать задержки в узлах дорожной сети.

В такой гибкой системе ключевую роль играет получение точных данных о потоке транспортных средств. Существует множество подходов к получению таких данных, одним из современных и наиболее эффективных является применение в качестве датчиков видеокамер.

Применение видеокамеры позволяет определять плотность потока транспортных средств по полосам, скорость и принадлежность к определенному классу (легковые, грузовые, общественный транспорт), а также осуществлять видеобзор с любой выбранной камеры в реальном времени, проводить сбор и обработку накопленных данных, управлять областями детектирования.

Однако в данном случае возникает вопрос создания качественных алгоритмов, позволяющих определять перечисленные существенные параметры потока с учетом различных погодных условий, условий размещения видеокамеры и самих качественных характеристик камеры.

Также алгоритмы должны обеспечивать обработку изображений в режиме реального времени с учетом того, что она накладывает на алгоритмы жесткие ограничения по времени и производительности.

Для реализации алгоритма определения параметров транспортного потока предлагается алгоритм, основанный на распознавании ключевых областей движения. Распознавание ключевых областей производится при помощи исследования характеристик корреляции на последовательностях кадров.

В общем случае система работает циклически. Полученные последовательности кадров в реальном режиме времени анализируются, затем принимается решение адаптивного управления, сохраняется статистика, полученная в процессе анализа.

Анализ изображений делится на следующие этапы [2],[3]:

1. Предварительная обработка изображения

Сигналы, получаемые от видеокамеры, имеют невысокое качество, так как камеры будут установлены на улице в неблагоприятных погодных условиях, снижающих качество изображения. Улучшение качества изображения проводится с применением различных видов фильтрации и сглаживания изображения. Выбор методики производится в зависимости от типа помех.

2. Обнаружение движущихся объектов

Этот этап наиболее важен, поскольку именно он определяет дальнейшее качество системы в целом. В принятой терминологии [4] считается, что движущиеся объекты принадлежат «переднему плану», а стационарные – к «заднему плану», который называют фоном. Процесс обнаружения движущихся объектов называют процессом выделения переднего плана. Для выделения движущихся объектов существует несколько различных методов. К таким методам относят следующие методы:

- методы вычитания фона;
- вероятностные методы;
- методы временного отличия;
- методы оптического потока.

Методы вычитания фона строят модель заднего плана сцены на основе попиксельного сравнения текущего значения и построенного на предыдущих шагах. В вероятностных моделях изменение значений пикселей во времени рассматривается как «пиксельный процесс», т.е. временной ряд, который для каждого пикселя характеризуется своей функцией распределения. Методы временной разности отделяют передний план от фона путем попиксельного вычитания двух или большего числа последовательных кадров. Методы оптического потока основаны на том, что у движущихся объектов можно вычислить направление и величину скорости для каждого пикселя.

3. Трекинг (сопровождение) объектов

Целью трекинга является установление соответствия между объектами или их частями в последовательности кадров, а также определение их траекторий и скорости движения. Отдельного решения требуют сложности при изменении ракурсов объектов во время движения и их частичное или полное перекрытие, когда один объект полностью загромождает другой. Однако решение данной проблемы выходит за рамки данной работы.

4. Принятие решения об адаптивном регулировании

Этот этап характеризует реакцию всей системы управления на дорожную обстановку. После получения и анализа последовательности кадров, вычисления состояния потока, необходимо отреагировать на обстановку и провести воздействие на светофорную систему перекрестка.

Для исследований в указанной области был осуществлен анализ данных этапов и предложен конкретный алгоритм, позволяющий с учетом требований, накладываемых областью применения, снимать необходимые параметры потока.

Данный алгоритм характеризуется простотой и эффективностью, так как анализируется небольшая последовательность кадров на наличие изменений, причем после нахождения изменений область поиска – окно – уменьшается для уменьшения расчетов и нагрузки на вычислительные средства.

Алгоритм позволяет достаточно точно для решения поставленной задачи детектировать объекты, затрачивая минимум вычислительных ресурсов.

Исходными данными для алгоритма является кадр-изображение, полученное с видеокамеры, и порог разницы в изображениях.

Алгоритм поиска движения и детектирования транспортных средств заключается в следующей последовательности шагов:

Шаг 1. Для анализа разницы изображений алгоритм получает текущее системное время, которое применяется для сбора статистики изменений на ряде получаемых изображений. Производится инициализация вспомогательных изображений для маски детектирования, сегмента поиска движения. Сохранение или инициализация предыдущих кадров, которые необходимы для поиска разницы.

Шаг 2. Осуществление преобразования переданного кадра из полноцветной палитры в полутоновую серую. Поиск разницы на заданном количестве последовательных файлов различий методом побитового сравнения. Запись различий во вспомогательное изображение. Обработка вспомогательного изображения пороговым фильтром для устранения малых изменений на изображениях и снижения шумов от погодных условий.

Шаг 3. Сохранение найденных изменений для дальнейшей корректировки движения объекта. Конвертация изображения с выделенным движением в синюю составляющую цвета.

Шаг 4. Расчет движения градиента ориентации и проверка правильности ориентации маски.

Шаг 5. Получение последовательности компонент движения, их количества.

Обработка каждой компоненты по отдельности.

А. Получение для компоненты региона обработки – области, в которой будет происходить уточненная обработка движения.

Б. Получение ориентации найденной компоненты-объекта.

В. Фильтрация наиболее мелких изменений и конечное сглаживание помех.

Определив все компоненты движения, далее вычисляются параметры транспортного потока, которые определяют управляющее воздействие на транспортный поток.

Используя данный алгоритм, можно точно вычислять плотность потока транспортных средств по полосам, скорость, длительности и количество разрывов в потоке и соответственно более точно определять управляющее воздействие.

Дальнейшее развитие данного подхода может идти в нескольких ключевых направлениях. Одно из направлений развития – внедрение различных, практически полезных модулей, например, модуля распознавания номеров. Наиболее важной особенностью при-

менения такого модуля является считывание номеров при пересечении автомобилем перекрестка. Данная информация может использоваться для поиска нарушителей и анализа дорожно-транспортных происшествий.

Другим важным направлением является усовершенствование методик и алгоритма уменьшения влияния погодных факторов и особенностей окружающей природы. Так, снегопад, дождь, вспышки света фар автомобилей при получении изображений ночью существенно ухудшают результаты обработки.

Литература

1. Врубель, Ю.А. Организация дорожного движения / Ю.А. Врубель – Минск, 1996.
2. Вороной, А. Методы и способы поиска событий в видеопотоке / А. Вороной – Дон-НТУ, 2003.
3. Лукьяница, А.А. Программный комплекс для мониторинга транспортных средств по видеоизображению / А.А. Лукьяница, А.Г. Шишкин – Москва: МГУ – 2004.
4. Прэтт, У. Цифровая обработка изображений У. Прэтт; пер.с англ. – М.: Мир, 1982.
5. Форсайт, Д. Компьютерное зрение. Современный подход / Д. Форсайт, Ж. Полс; пер.с англ.- М.: Вильямс, 2004.

УДК 576.8:004.932.72'1

АВТОМАТИЗИРОВАННАЯ СИСТЕМА РАСПОЗНАВАНИЯ ПАЗИТОЛОГИЧЕСКИХ ОБЪЕКТОВ

Корчевская Е.А., Мироненко В.М., Жигалко О.С.

*УО «Витебский государственный университет имени П.М.Машерова»,
УО «Витебская ордена «Знак Почета» государственная академия
ветеринарной медицины», г.Витебск*

Задача распознавания образов заключается в классификации некоторой группы объектов на основе определенных требований. Объекты, относимые к одному классу образов, обладают общими свойствами. Требования, определяющие классификацию, могут быть разными, так как в различных ситуациях возникает необходимость в различных типах классификаций.

Простейший подход к распознаванию образов заключается в сопоставлении с эталонами. В этом случае некоторое множество образов хранятся в памяти машины. Исследуемый образ сравнивается с эталоном каждого класса. Классификация основывается на заранее выбранном критерии близости, однако имеет несколько недостатков. Главный из них заключается в том, что в ряде случаев трудно выбрать подходящий эталон из каждого класса образов и установить необходимый критерий соответствия. Эти трудности особенно существенны, когда образы, принадлежащие одному классу, могут значительно изменяться. Более совершенный подход заключается не в сравнении исследуемого образа с эталонами, а основывается на сравнении некоторым специальным образом проведенных замеров исходного и эталонного образов.

В процессе создания автоматизированной системы распознавания паразитологических объектов необходимо решить следующие основные задачи:

1. Оконтуривание изображений.

2. Вычисление основных параметров объектов, необходимых для идентификации изображений.

3. Анализ изображения паразитологических объектов.

Выделение контура основывается на переводе цветного изображения в черно-белое. Для этого необходимо установить личные пропорции цветов каждого пикселя цветного изображения, соответствующих координатным цветам системы RGB. После чего производится перевод цветного изображения в оттенки серого по формуле:

$$T=0,3\cdot R+0,59\cdot G+0,11\cdot B, \quad (1)$$

где R, G, B- численное отношение личных пропорций цветов Red(красный), Green(зеленый), Blue(синий).

Далее используется методика глобального порогового разделения – метод мод для перевода изображения в оттенках серого в бинарное изображение. После подавления шумов и подчеркивания границ получаем выделенный контур черного цвета на белом фоне.

Вторая задача заключается в определении того, какие измерения на входном объекте могут играть роль признаков. Отметим, что признаки могут иметь различную природу и значимость для задачи классификации, поэтому отбор признаков и их упорядочивание основывается на важности этих признаков для характеристики образов или на влиянии данных признаков на качество распознавания.

Разработанная система распознавания объектов рассчитывает следующие параметры: длина объекта, ширина объекта, радиусы кривизны полюсов объекта, площадь, ограниченная контуром и периметр области. Для распознавания объектов авторами предложены следующие идентификационные показатели: отношение квадрата периметра контура к площади поверхности объекта, ограниченного данным контуром, а также произведение отношений длины к ширине и наименьшего к наибольшему радиусу кривизны полюсов объекта.

Для расчета указанных параметров составляется математическая модель контура объекта. Предварительный анализ разработанного алгоритма с реализацией интерполяционного многочлена различными методами (наименьших квадратов, Ньютона, Эйткена, интерполирование сплайнами различных порядков и др.) показал наибольшую эффективность использования интерполяционного многочлена Лагранжа. Периметр объекта и площадь, ограниченная контуром, вычисляются с помощью квадратурных формул. Для вычисления производных, входящих в формулу, для вычисления кривизны полюсов объекта используются конечноразностная аппроксимация производной.

Третья задача распознавания образов заключается в классификации, т.е. в принятии решения о принадлежности входного образа тому или иному классу.

Для получения численных значений указанных параметров была использована тренировочная коллекция, то есть объекты, для которых заранее известно, к какому виду они относятся. Для каждого вида выполнены расчеты указанных безразмерных параметров и занесены в список шаблонов, при сравнении с которым осуществляется принятие решения о принадлежности объекта к конкретному виду.

Разработанное программное обеспечение позволяет выполнять оконтуривание объектов по введенному в компьютер изображению. Это позволяет рассчитывать безразмерные параметры паразитологического объекта, необходимые для определения принадлежности определенной группе.

УДК 004.932

СОПРОВОЖДЕНИЕ ДВИЖУЩИХСЯ ОБЪЕКТОВ МЕТОДОМ КОРРЕЛЯЦИОННОГО АНАЛИЗА ИЗОБРАЖЕНИЙ В СИСТЕМАХ КОМПЬЮТЕРНОГО ЗРЕНИЯ

Краснобаев Е.А.

УО «Витебский государственный университет им. П. М. Машерова», г. Витебск

Сопровождение движущихся объектов в видеоизображениях – одна из задач компьютерного зрения, широко используемая во многих сферах человеческой деятельности. В соответствии с [1], под сопровождением понимают последовательное получение изображения движущегося объекта в течение всего времени его пребывания в кадре, и подразумевается определение дополнительных характеристик объектов: координат центра объекта, скорости, траектории, время наблюдения и др. При этом к алгоритму решения задачи предъявляются определенные требования:

1. Возможность сопровождать несколько движущихся объектов. Выполнение требования приводит к необходимости идентификации изображений движущихся объектов, с целью определения принадлежности изображения к определенному объекту.

2. Возможность возобновлять сопровождение в случае кратковременной потери объекта в кадре (в случае, если объект скрылся за препятствием в ходе движения, или объекты пересеклись).

3. Возможность сопровождать объекты в случае изменения их ракурса по отношению к камере, удаления/приближения.

Первоочередным этапом решения задачи сопровождения является сегментация движущихся объектов и получение бинарной маски движения [2, 3]. Второй этап – морфологическая обработка изображения с целью выделения в изображении групп пикселей соответствующих движущимся объектам. Третий этап – определение сходства изображений объектов, выделенных на этапе 2.

Одним из методов решения задачи этапа 3 является корреляционный анализ изображений. Он дает возможность установить наличие определенной взаимосвязи между изображениями или меру их сходства. Сходство изображений определяется путем вычисления функции взаимной корреляции, например, в виде:

$$C = \frac{\sum_{i,j} I_1(i,j)I_2(i,j)}{\sqrt{\sum_{i,j} I_1(i,j)^2} \sqrt{\sum_{i,j} I_2(i,j)^2}}$$

где $I_1(i,j)$ и $I_2(i,j)$ – яркость пикселей (i,j) сравниваемых изображений объектов I_1 и I_2 . В случае сходства изображений коэффициент корреляции C будет максимальным.

Определение сходства изображений объектов может осуществляться двумя путями:

1. Сегментация и выделение движущихся объектов осуществляется в каждом кадре видеоизображения. В каждой паре кадров находятся соответствия между фрагментами, которые соответствуют изображениям объектов.

2. Сегментация и выделение движущихся объектов осуществляется в первом кадре видеоизображения. Фрагмент изображения соответствующего движущемуся объекту будет считаться эталонным. В последующих кадрах определяется новое положение объекта путем поиска в кадре фрагмента, максимально схожего с эталонным.

Сравнивая два описанных подхода, можно отметить, что первый случай является наиболее ресурсоемким, так как в каждом кадре необходимо производить сегментацию и выделение объектов. Но при этом во время сопровождения объекты могут значительно менять свою форму, что не повлияет на качество сопровождения, так как за один кадр вид объекта не может существенно измениться. Во втором случае – сравнение происходит с эталонным изображением, которое является неизменным в ходе сопровождения. В этом случае подбирается фрагмент кадра наиболее близкий к эталонному изображению, и предполагается, что за все время нахождения в кадре объект значительно не меняется. Для устранения этого ограничения метода вводится обновление с каждым кадром эталонного изображения объекта.

Рассмотрим подробно этапы алгоритма второго подхода.

1. Получение эталонного изображения методами сегментации в кадре I_n . Сохранение изображения w_{et} . Определение координат центра объекта относительно кадра.

2. В кадре I_{n+1} формируется область $w > w_{et}$, в которой предусматривается поиск нового положения объекта. Размер области определяет максимальное возможное смещение объекта за время одного кадра, т.е. определяет максимальную скорость движения объекта в кадре.

3. Для каждого окна w_{img}^i размера w_{et} из области w вычисляется коэффициент корреляции с изображением w_{et} . Окно w_{img}^k , для которого коэффициент корреляции будет максимальным и большим некоторого порога T , будет соответствовать смещенному объекту w_{et} на изображении I_n .

4. Обновление эталонного изображения объекта $w_{et} = w_{img}^k$.

5. Повторение этапов 2-4 для следующего кадра видеопоследовательности.

Разработанный алгоритм сопровождения движущихся объектов в видеопотоке позволяет выполнять указанные выше требования 1-3. На основе данного алгоритма создан прототип программы для сопровождения движущихся объектов.



Рисунок 1 – Результат сопровождения движущихся объектов методом корреляционного анализа изображений

Проведены экспериментально-тестовые испытания программы на процессоре AMD Sempron 2.1 ГГц, с разрешением видео 352x288 пикселей. Результат сопровождения движущихся объектов приведен на рис 1. Алгоритм удовлетворяет условиям работы в режиме реального времени и может использоваться в системах компьютерного зрения.

Литература

1. Краснобаев, Е. А. Моделирование оптических систем автоматического сопровождения и целеуказания / Е. А. Краснобаев // 4-ая Международная научная конференция по военнотехническим проблемам, проблемам обороны и безопасности, использованию технологий двойного применения, Минск, 20–21 мая 2009 г./ БелИСА – Минск, 2009 – С. 108.

2. Краснобаев, Е. А. Метод оптического потока в задачах сегментации движущихся объектов в видеоизображениях: / Е. А. Краснобаев, А. Ю. Халанский // X(55) региональная научно-практическая конференция преподавателей, научных сотрудников, аспирантов и студентов, Витебск, 28 апр. 2008 г. / ВГУ им. П.М. Машерова – Витебск, 2008 – С. 43.

3. Краснобаев, Е. А. Адаптивная модель фона в задачах сегментации движущихся объектов в видеоизображениях: / Е. А. Краснобаев А. Ю. Халанский // Третья конференция молодых учёных «Молодежь и наука в XXI веке», Витебск, 2008 – С. 11.

УДК 621.397.13:004.932.75'1

ОРГАНИЗАЦИЯ ЭФФЕКТИВНОЙ ОБРАБОТКИ ЦИФРОВЫХ ИЗОБРАЖЕНИЙ СРЕДСТВАМИ .NET-ПЛАТФОРМЫ

Кузьмицкий Н.Н.

УО «Брестский государственный технический университет», г. Брест

Решение задач анализа цифровых изображений предусматривает большую вычислительную работу, построенную на использовании специализированных алгоритмов и многократной обработке обширной базы изображений. Разрабатываемое программное обеспечение, помимо прочной теоретической основы, должно обладать высокими показателями производительности, связанными с его преимущественным применением в условиях реального времени. Поэтому выбор мощной и универсальной среды разработки является одним из ключевых факторов успешного решения поставленных задач, при этом необходимо учитывать как их специфику, так и современные тенденции развития программной индустрии.

Одним из возможных вариантов является направление, основанное на использовании профессиональных математических пакетов (например, Matlab), в которых задачи и решения выражаются в привычном математическом виде. Однако на их основе затруднительно построить универсальное пользовательское приложение, к тому же существенными являются вопросы авторского права и стоимости подобных пакетов.

Перспективным представляется применение платформы .NET: мощной, динамично развивающейся, обладающей библиотекой компонент на "все случаи жизни" (FrameWork), возможностями взаимодействия с унаследованным кодом, удобными методами доступа к базам данных, создания сетевых приложений и т.д. Однако данная платформа, как и др., не уменьшает актуальности вопроса производительности итогового программного обеспечения и ее сравнимости с возможностями профессиональных пакетов.

В частности, с таким вопросом пришлось столкнуться и автору данной работы в связи с необходимостью использования морфологической реконструкции, являющейся важным инструментом анализа полутоновых и бинарных изображений.

Среди имеющихся вариантов описания реконструкции была выбрана "гибридная реконструкция", как наиболее быстрая и при этом являющаяся элементом Matlab Image Processing Toolbox [1]. В результате, на С# с использованием стандартных языковых средств был написан соответствующий алгоритм (version 1.0), скорость работы которого и его Matlab аналога (imreconstruct), отражены в таблице.

Полученные результаты подтолкнули к детальному анализу первоначальной реализации, в ходе которого была обнаружена низкая производительность методов GetPixel() и SetPixel() - членов класса .NET Bitmap, предназначенных для чтения и присвоения значения пикселя изображения. В то же время в данном классе имеются средства, позволяющие значительно ускорить их работу, для этого необходимо придерживаться схемы приведенной на рисунке 1.

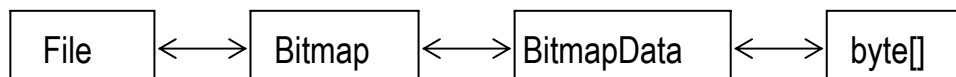


Рисунок 1 – Схема оптимальной работы с изображениями в .NET

После чтения графического файла и создания объекта Bitmap с помощью метода LockBits() осуществляется переход к объекту BitmapData, инкапсулирующему подробную информацию об изображении, в частности, его размещении в системной памяти. С помощью BitmapData можно организовать взаимодействие с объектами типа Bitmap как с массивами байтов, что значительно эффективнее, чем использование методов GetPixel() и SetPixel(). В среднем время попиксельной обработки уменьшилось в 4 раза, что значительно отразилось на скорости работы алгоритма (version 2.0) и сравнимо со скоростью подобных операций в специализированных пакетах.

Следующим шагом в повышении производительности было внедрение в рассматриваемый алгоритм "небезопасного кода". При этом под "небезопасным кодом" в .NET подразумевается код, который не выполняется под полным контролем системы управления .NET CLR [2]. В частности, в качестве основного инструмента работы с матрицами изображений были выбраны указатели. Несмотря на некоторую нестандартность применения технологии указателей в .NET и заметные затраты, связанные с вычислением адреса каждого пикселя, также было получено существенное ускорение работы алгоритма (version 3.0), соответствующие результаты отражены в таблице 1.

Таблица 1 – Сравнение версий алгоритма "гибридной" реконструкции

№	Imreconstruct	Version 1.0	Version 2.0	Version 3.0
Picture1	0.96 с	8.90 с	4.84 с	2.65 с
Picture2	1.16 с	9.97 с	5.41 с	2.96 с
Picture3	1.26 с	10.72 с	6.56 с	3.52 с

Таким образом, на примере отдельного алгоритма математической морфологии было показано, что платформа .NET обладает достаточным потенциалом в решении задач разработки алгоритмов цифровой обработки изображений, хотя иногда оптимальные варианты реализации могут не лежать на поверхности.

При этом особенно перспективным представляется вариант ее совместного использования с Matlab, как средством проведения сложной экспериментальной работы.

Литература

1. Vincent, L. "Morphological Grayscale Reconstruction in Image Analysis: Applications and Efficient Algorithms," / L. Vincent // IEEE Transactions on Image Processing, Vol. 2, No. 2, April, 1993. – p. 176-201
2. Шилдт, Г. Полный справочник по С# / Г. Шилдт. – М.: Издательский дом "Вильямс", 2004. – 752 с.

УДК 681.3

АНАЛИЗ И РАСПОЗНАВАНИЕ ИЗОБРАЖЕНИЙ КОСТЕЙ КИСТИ**Лисица Е. В.**

УО «Белорусский государственный университет», г. Минск

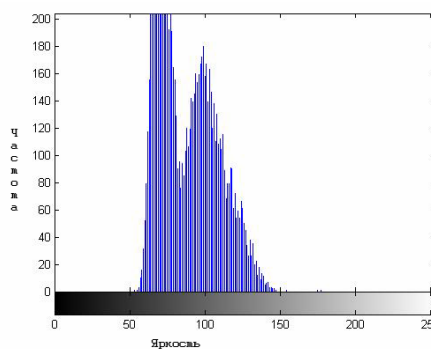
Как известно, центром опоры и двигательной активности, ключевой структурой запястья является полулунная кость. Для оценки взаимоотношений между костями пястья возможно выполнение рентгенограмм в нескольких проекциях. В данной работе рассмотрена прямая стандартная проекция (прямая проекция с предплечьем, находящимся в полной супинации), на которой могут быть заметны изменения с ладьевидно-полулунным промежутком, ладьевидной костью, полулунной костью. Основной целью было классифицировать наличие либо отсутствие расширения ладьевидно-полулунным промежутка, и изменения положения ладьевидной кости. Эти два признака приводят к заметным изменениям на внешних границах лучезапястного сустава. Данная задача может быть полезной для упрощения анализа структуры кисти и улучшения качества диагностики.

Исходная задача может быть поделена на два этапа: выделение внешних границ полулунной, трёхгранной, лучевой и ладьевидной костей; классификация наличия признаков по общей границе этих костей. Для работы с рентгенограммами существует специальный стандарт файлов – DICOM файлы, которые являются исходными несжатыми индексированными изображениями. Однако широкое распространение также получили JPEG-файлы, которые получаются при сжатии исходного DICOM файла. Поэтому выделение границ было рассмотрено для обоих типов файлов.

Операции сжатия, фильтрации, растяжения гистограммы пикселей позволяют привести снимки к более однородному виду. Однако их применение приводит к потерям информации, что в результате только усложняло последующий анализ. Поэтому изображения не подвергались предварительной обработке, а выделение границ проводилось методом глобального порогового разделения. В нашем случае на изображении чётко делимы два объекта: фон и сам объект. Что отражается на гистограмме яркостей пикселей как чётко видимых два пика: первый – фоновый пик, второй – пик для объекта. Используя гистограмму яркостей, можно найти порог для перевода изображения в монохромный вид. К сожалению, применение порогового метода ко всему изображению не является целесообразным, так как в этом случае возникают ошибки: к костным тканям могут приравниваться мышечные ткани, фон изображения и т.д. Поэтому пороговый метод применялся для области возле точки пересечения изображения трапециевидной и первой пястной кости (первая точка), где гистограмма пикселей содержит только пиксели, принадлежащие костной и мышечным тканям, что позволяет устранить отмеченные выше ошибки.



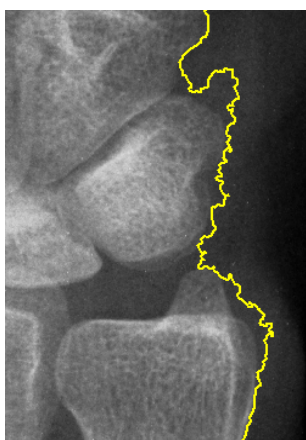
а) исходное изображение



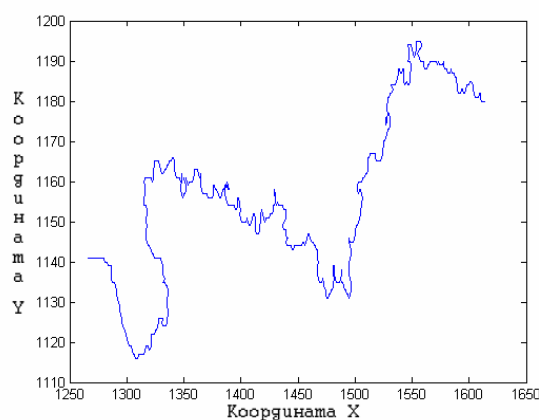
б) гистограмма окрестности точки

Рисунок 1 – Исходное изображение с гистограммой яркостей

Также помимо этой точки использовались ещё две вспомогательные точки: на пересечении 5-ой пястной кости с крючковидной (вторая точка) и пересечение лучевой кости с радиальной (третья точка). Первая и вторая точки применялись для поворота изображения. Граница для анализа находится между первой и третьей точками.



а) область изучения



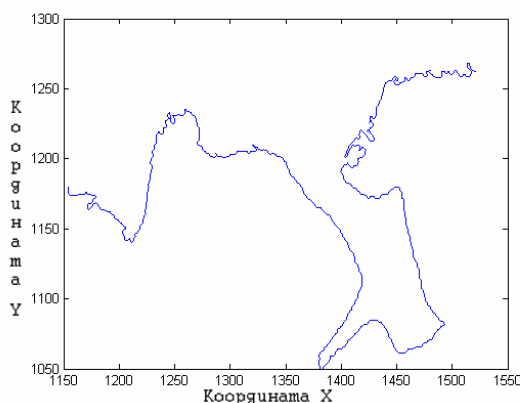
б) внешняя граница костей

Рисунок 2 – Смещение ладьевидной кости

На рисунке 2 показано выделение границ при наличии смещения ладьевидной кости. Для сравнения ниже на рисунке 3 приведён пример для увеличенного промежутка между полулунной и ладьевидной костями.



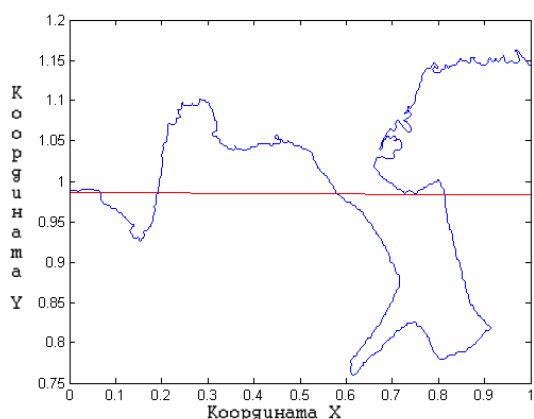
а) область изучения



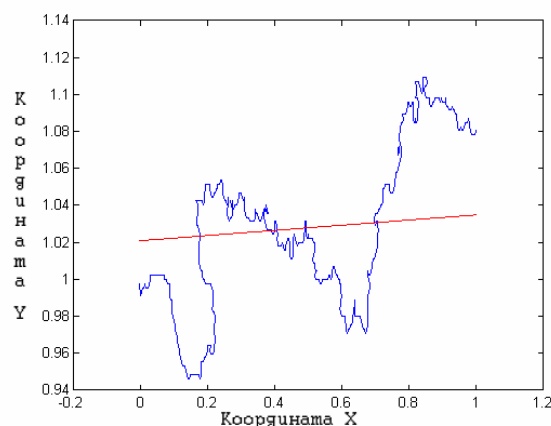
б) внешняя граница костей

Рисунок 3 – Промежуток между ладьевидной и полулунной костями

При анализе найденных границ было замечено, что при смещении ладьевидной кости граница стремится к прямой линии, а при увеличении промежутка между костями выделенная граница становится более вогнутой. Поэтому для классификации этих двух дефектов применялась предварительная аппроксимация найденной границы прямой. Затем на вход первой искусственной нейронной (ИНС) сети подавалась два параметра: квадратичное отклонение всех найденных точек от прямой и квадратичное отклонение точек, лежащих под прямой. Первая нейронная сеть находит смещение ладьевидной кости. Если не было найдено этого дефекта, вторая ИНС проверяет на наличие увеличенного промежутка между полулунной и ладьевидной костями. Предварительно перед подачей на входы ИНС проводилась стандартизация полученных значений, где за единицу длины по координате Y бралось расстояние между первой и второй точками, а по координате X бралось расстояние между третьей и первой точкой.



а) промежуток ладьевидной и полулунной костей



б) смещение ладьевидной кости

Рисунок 4 – Аппроксимация прямой линией

При использовании отмеченных выше параметров задачи сводятся к линейно-разделимым задачам. Поэтому обе ИНС являются персептронами с линейной передаточной функцией с двумя входами и одним выходом.

Рассмотренные методы могут быть полезны для нахождения отклонений в строении кисти и улучшения качества диагностики. В дальнейшем планируется улучшить качество нахождения границ, используя метод активных контуров, а так же обучить нейронную сеть для нахождения других отклонений.

Литература.

1. Гонсалес, Р. Цифровая обработка изображений в среде MATLAB / Р. Гонсалес, Р. Вудс, С. Эддинс – М.: «Техносфера» 2006.
2. Голубев, И. О., Хирургия кисти: карпальная нестабильность/ И.О. Голубев – Ярославль: «Аверс Пресс» Избранные вопросы пластической хирургии. 2001. – Т. 1. – № 8.
3. Волотовский, А.И. Диагностика и лечение повреждений ладьевидно-полулунного сочленения запястья / А.И. Волотовский – Мн.: 2008.
4. Bishop, C. Neural Networks for Pattern Recognition. Oxford. 1997.
5. Ayres, F. Characterization of Architectural Distortion in Mammograms. IEEE Engineering in Medicine and Biology 2005, January. P. 59-67.
6. Абламейко, С.В. Обработка изображений: технология, методы, применение / С.В. Абламейко, Д.М. Лагуновский – Мн.: «Амалфея» 2000.

АНАЛИЗ И МОДЕЛИРОВАНИЕ НЕЛИНЕЙНЫХ ДИНАМИЧЕСКИХ ПРОЦЕССОВ

УДК 004

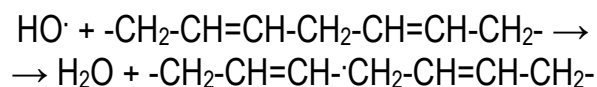
ALGORITHMIC SIMULATION IN STUDYING THE KINETICS OF LIPID PEROXIDATION

Anatole D. Ruslanov, Anton V. Bashylau

State University of New York, Fredonia, New York, USA

The biochemical model we use is a set of chain reactions of lipid peroxidation, thus being a complex system of chemical interactions that includes the following stages:

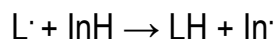
1. *Chain initiation.* During the interaction of a strong oxidizer with an unsaturated fatty acid, specifically with the $-\text{CH}_2$ -group in the α -position relative to the double bond, a hydrogen atom detaches from the bond, forming a free radical (mostly alkyl radicals) of a fatty acid ($\text{L}\cdot$). The free radicals such as hydroxyl radical $\text{HO}\cdot$ are the most frequent reagents capable of the separation of the hydrogen atom:



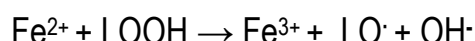
2. *Chain prolongation.* In an environment that contains oxygen, the oxygen molecule rapidly bonds to one of the carbon atoms with unpaired electrons, usually to the outermost carbon atom. Thus, the lipid peroxide radical ($\text{LO}_2\cdot$) is formed. The radical $\text{LO}_2\cdot$ readily takes the hydrogen atom from a neighboring unsaturated fatty acid forming a new lipid alkyl radical and, thereby, forming a chain reaction.

3. *Chain breaking.* The development of free-radical chain reactions is stopped by the reactions of chain breaking. This happens due to the disappearance of free radicals $\text{L}\cdot$ and $\text{LO}_2\cdot$, which directly contribute to the oxidation chain. In lipid-containing systems, the length of these chains can be tens or hundreds of links.

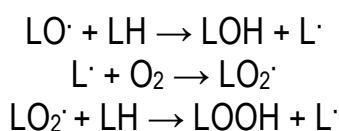
For the cells, the development of the oxidation chains is an undesirable phenomenon both because it leads to useless burning of valuable cell components and because some products of lipid peroxidation are toxic. In this regard, cells have a whole system of protection against peroxidation, in particular the antioxidants, which impede the oxidation chain reaction by breakage of chains:



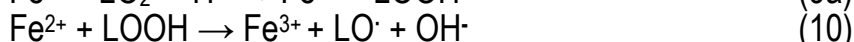
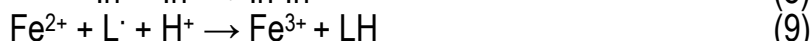
4. *Chain branching.* In most cases, Fe^{2+} cations have prooxidant properties, i.e., they do not suppress but rather reinforce the process of lipid peroxidation. This is because iron cations react with the peroxidation products—lipid hydroperoxides:



As a result, a new free lipid alkoxy radical $\text{LO}\cdot$ appears, which initiates the formation of a new oxidation chain:



The combination of chemical kinetics equations of the chain oxidation of lipids in the system iron (II)-ascorbate-dependent lipid peroxidation is as follows:



The system of differential equations that describes the process of lipid peroxidation is complex and the speed constants of most of these reactions are unknown. The system of chemical and, therefore, differential equations can be significantly simplified if we assume the following:

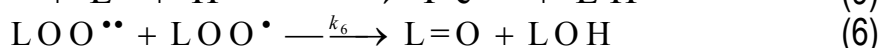
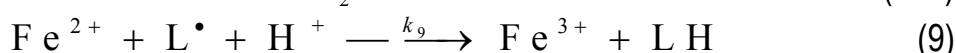
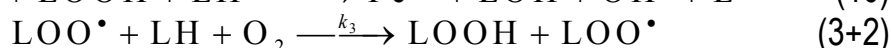
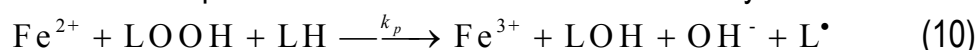
1: The speed of several consecutive reactions is equal to the speed of the slowest one. This allows replacing the sequential reaction with one generalized reaction, taking the speed constant of this reaction to be equal to the speed constant of the slowest of the successive reactions. In our case, the reactions 2 and 3 are sequential reactions. For our purposes, it is not important to consider which of the reactions should be considered the slowest. In an environment that contains oxygen, the rate of reaction 3 is many times lower than the speed of reaction 2, so the equation for the continuation of the chain reaction can be written as follows:



The rate constant of this reaction will be equal to the rate constant of reaction 3, which we denote as k_3 .

2: Among the several parallel reactions, the fastest one of the greatest importance. This permits us to simplify the above system of the eleven events. In the presence of molecular oxygen, the concentrations of radicals $\text{LO}\cdot$ and $\text{L}\cdot$ are much less than that of the radicals $\text{LOO}\cdot$; therefore reactions 4 and 5 can be neglected, leaving only the radical interaction reaction—the reaction 6. In the presence of ions Fe^{2+} in the environment and at low concentrations of other antioxidants, the reaction 7 can also be often ignored, leaving only the reaction 9. In addition, we shall not yet take into account the reaction 9a.

In the absence of other sources of free radicals, the formation of new radicals occurs only in the reaction 10 (the chain branching reaction). Let k_p denote the rate constant for this reaction. The reaction 1 can be neglected, if the source of the radical $\text{HO}\cdot$ does not exit. After all these simplifications, we obtain a system of chemical equations for LPO that contains a total of only four reactions:



The reactions 9 and 10 appear as trimolecular or as third order reactions. In the water environment, however, the concentration of H^+ is constant, and this value can be included in the constant for speed of k_9 . Therefore, we shall consider the reaction 9 as a bimolecular reaction with the second order rate constant k_9 . In addition, the concentration of oxidation substrate for LH within the membrane is sufficiently large and changes little over time; thus, the reaction 10 can also be assumed bimolecular (second order).

After this introduction, we write the kinetic differential equations that describe the speed for the particular lipid peroxidation chain reactions described above as follows:

$$v_3 - k_3[\text{LH}][\text{LOO}\cdot] \quad (11)$$

$$v_6 - k_6[\text{LOO}\cdot]^2 \quad (12)$$

$$v_9 - k_9[\text{Fe}^{3+}][\text{LOO}\cdot] \quad (13)$$

$$v_p - k_p[\text{Fe}^{2+}][\text{LOOH}] \quad (14)$$

The differential equations that describe the changes in the concentrations of the major participants in the above reactions, while taking into account the equations described above, can be written as follows:

$$\frac{d[\text{LOOH}]}{dt} = v_3 - v_p \quad (15)$$

$$\frac{d[\text{LOO}\cdot]}{dt} = v_p - v_6 - v_9 \quad (16)$$

$$\frac{d[\text{Fe}^{2+}]}{dt} = -v_9 - v_p \quad (17)$$

This final system of third order equations cannot be solved analytically, even after the simplifications made above. However, if the velocity constants and the initial concentrations of the reaction reagents are known, the changes in time for the concentrations of all participants in the reactions can be computed numerically using these equations, i.e., modeled algorithmically in a computational environment.

The easiest method for such modeling is of Euler and Cauchy, in which infinitesimal increments are replaced by sufficiently small increments and thus the integration is replaced by summation. The algorithm for computation is as follows:

1. The system of differential equations is reduced to its algebraic form, using infinitely small increments for the reagent concentrations and for time:

$$\Delta[\text{LOOH}] = (v_3 - v_p)\Delta t \quad (18)$$

$$\Delta[\text{LOO}\cdot] = (v_p - v_6 - v_9)\Delta t \quad (19)$$

$$\Delta[\text{Fe}^{2+}] = (-v_9 - v_p)\Delta t \quad (20)$$

2. The initial values for $[\text{LOOH}]$, $[\text{LOO}\cdot]$, and $[\text{Fe}^{2+}]$ are substituted into the right side of the equations. Specifying an infinitely small value for t , we compute the values $D_1[\text{LOOH}]$, $D_1[\text{LOO}\cdot]$, and $D_1[\text{Fe}^{2+}]$.

3. We find the values for the time $t_1 = 0 + t$:

$$[\text{LOOH}]_1 = [\text{LOOH}]_0 + D_1[\text{LOOH}] \quad (21)$$

$$[\text{LOO}\cdot]_1 = [\text{LOO}\cdot]_0 + D_1[\text{LOO}\cdot] \quad (22)$$

$$[\text{Fe}^{2+}]_1 = [\text{Fe}^{2+}]_0 + D_1[\text{Fe}^{2+}] \quad (23)$$

4. Based on the new values of the current concentrations of $[\text{LOOH}]$, $[\text{LOO}\cdot]$ и $[\text{Fe}^{2+}]$, we compute the new values for concentrations increments $D_2[\text{LOOH}]$, $D_2[\text{LOO}\cdot]$, and $D_2[\text{Fe}^{2+}]$.

5. We find values for the time $t_2 = t_1 + D_t$:

$$[\text{LOOH}]_2 = [\text{LOOH}]_1 + k_2[\text{LOOH}] \quad (24)$$

$$[\text{LOO}\cdot]_2 = [\text{LOO}\cdot]_1 + D_2[\text{LOO}\cdot] \quad (25)$$

$$[\text{Fe}^{2+}]_2 = [\text{Fe}^{2+}]_1 + D_2[\text{Fe}^{2+}] \quad (26)$$

6. Repeating the steps 4 and 5, we compute concentrations for the reagents in the lipid peroxidation reaction for any given time.

7. Finally, we compare the kinetic curves that are computed with the experimental data.

With some additional simplification of the system it is possible to reduce the system of differential equations to equations of the first order, which can be solved analytically. These simplifications, however, may deny the opportunity to construct a complete curve for the kinetics of the process, but may allow computation for small segments of the process, and more importantly may allow for better understanding of some features of these reactions.

The first simplification is based on the assumption that the speed of the particular reaction significantly exceeds the rate of the chain oxidation as a whole. The meaning of this assumption can be explained by analyzing the equation that describes the rate of change of the radical LOO^\bullet concentration:

$$\frac{d[\text{LOO}^\bullet]}{dt} = k_p[\text{Fe}^{2+}][\text{LOOH}] - k_9[\text{Fe}^{2+}][\text{LOO}^\bullet] \quad (27)$$

It follows from Equation 27 that the rate of appearance for the new radicals is determined by the difference in the velocities of the two particular reactions: the chain branching and the chain breaking. If we assume that the rate of the radical formation is significantly less than the speed of each of these particular reactions, then the right-hand side of the equation can be assumed to be equal to zero. Then

$$k_p[\text{Fe}^{2+}][\text{LOOH}] = k_9[\text{Fe}^{2+}][\text{LOO}^\bullet] \quad (28)$$

The order of the system of equations is, thus, reduced by one unit. Such simplification, called the Bodenshteyn-Semionov stationary approximation, assumes long oxidation chains. It should be noted, however, that the main conclusion remains without the Bodenshteyn-Semionov approximation but requires a more complex mathematical computation.

Let us rewrite the expanded form of the equation using the Bodenshteyn-Semionov approximation:

$$k_p[\text{Fe}^{2+}][\text{LOOH}] = k_9[\text{Fe}^{2+}][\text{LOO}^\bullet] \quad (29)$$

$$\frac{d[\text{LOOH}]}{dt} = k_3[\text{LH}][\text{LOO}^\bullet] - k_p[\text{Fe}^{2+}][\text{LOOH}] \quad (30)$$

In addition to the concentrations of hydroperoxides and radicals, the equations include another variable, the concentration of Fe^{2+} . Therefore, in order to solve the system of equations, it is necessary to use very small time intervals in which the value of $[\text{Fe}^{2+}]$ does not change much and can be assumed to be constant with a known value in that segment of time. With this additional assumption, we can substitute the values $[\text{LOO}^\bullet]$ from the first equation into the second and divide the variables:

$$\frac{d[\text{LOOH}]}{dt} = k_3[\text{LH}] \frac{k_p[\text{LOOH}]}{k_9} - k_p[\text{Fe}^{2+}][\text{LOOH}]$$

$$\frac{d[\text{LOOH}]}{dt} = \gamma[\text{LOOH}]$$

$$\frac{d[\text{LOOH}]}{[\text{LOOH}]} = \gamma dt$$

$$\text{where } \gamma = k_p \left(\frac{k_3[\text{LH}]}{k_9} - [\text{Fe}^{2+}] \right) \quad (31)$$

Integrating the last equations we get:

$$\int_{[\text{LOOH}]_0}^{[\text{LOOH}]} \frac{d[\text{LOOH}]}{[\text{LOOH}]} = \gamma \int_0^t dt$$

$$\ln \frac{[\text{LOOH}]}{[\text{LOOH}]_0} = \gamma t \quad (33)$$

The equations 31-33 imply that:

$$[\text{LOOH}] = [\text{LOOH}]_0 e^{\gamma t}$$

$$[\text{LOO}^\bullet] = [\text{LOO}^\bullet]_0 e^{\gamma t} \quad (34)$$

The last equation demonstrates that the hydroperoxide concentration in the system and the concentration of oxidation chains (which is equal to the concentration of the radicals that lead the chain) varies in time exponentially. Acceleration or deceleration of the reaction depends on the sign of the degree.

Existence of many antioxidant types with different mechanisms of action complicates determination of the antioxidant activity in biological systems. In this work, we used mathematical modeling for achieving algorithmic simulation to study the kinetics of iron (II)-ascorbate-dependent lipid peroxidation. We selected the minimum number of reactions, the combination of which allowed us to reproduce the phenomenology of the experiments. We then explored the mechanisms and characteristics of these active substances under various conditions, including theoretical predictions of their antioxidant behavior. Pursuing an understanding of antioxidant activity of medicinal substances of plant origin is important as many known diseases are associated with an oxidative stress, i.e., an impairment of antioxidant activity.

УДК 004

ПОДХОД К ЧИСЛЕННОМУ АНАЛИЗУ ЭЛЕМЕНТОВ РЭА

Бондарь С.Г.

УО «Брестский государственный технический университет», г. Брест

В данной работе отражены два подхода к численному анализу теплового режима: метод начальных параметров и метод конечных элементов. Пользуясь методом абстрагирования, мы рассмотрим проводник, на который воздействует тепловое поле, и проведем аналогию с закрепленной балкой с воздействием сосредоточенных сил.

Расчет ведут для того, чтобы рассчитанные температурные характеристики теплового режима не превышали предельные допустимые нормы.

Метод начальных параметров

Необходимо определить прогиб проводника с 3 опорами (рис. 1) с приложенной сосредоточенной силой F и находящимися на расстоянии $-x$ и $+x$ силами $F/2$, эквивалентными тепловому воздействию на участке $[L2-x, L2+x]$.

Определяются функции:

$$\text{mag1}(x, a) = (x - a) \cdot (x \geq a)$$

$$\text{mag2}(x, a) = \frac{1}{2!} (x - a) \cdot (x \geq a)$$

$$\text{mag3}(x, a) = \frac{1}{3!} (x - a) \cdot (x \geq a)$$

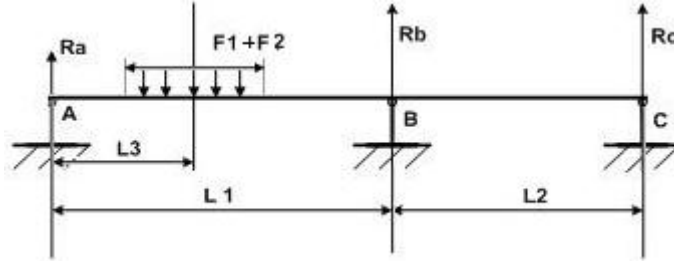


Рисунок 1

Определяются функция моментов, функция угла деформации и функция прогиба, в зависимости от координаты по X, начальных условий по углу деформации и прогибу. Используются функции mag1, mag2, mag3.

Задаются известные граничные условия, и записывается система уравнений для их нахождения. В систему вводятся также соотношения для сумм нагрузок по OY и моментов относительно произвольной точки.

На базе решения системы уравнений определяются функция моментов, функция угла деформации и функция прогиба.

Строятся графики: функция моментов, функция угла деформации и функция прогиба. Метод конечных элементов (одномерный случай).

Определяется функция моментов для данного случая нагрузки, затем выполняются основные этапы метода конечных элементов для разработки документа математической системы. Элементы реализации приведены ниже.

$$x = \begin{pmatrix} 0 \\ L_1 \\ L_1 + L_2 \\ L \end{pmatrix} \quad M_1 = \begin{pmatrix} M(x_0) \\ M(x_1) \\ M(x_2) \\ M(x_3) \end{pmatrix}$$

Зададим начальные условия:

$$y_0 = 0 \quad y_2 = 1 \quad dy_0 = 1$$

$$y_1 = 0 \quad y_3 = 0 \quad dy_3 = 1$$

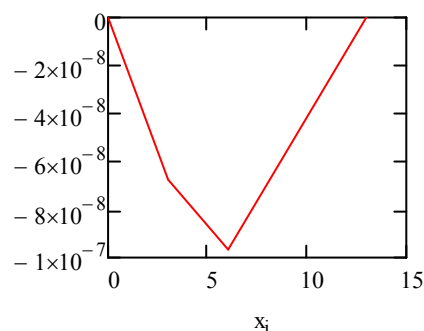
Получим:

$$\begin{pmatrix} \frac{1}{L_1} & -\frac{1}{L_1} & 0 & 0 \\ -\frac{1}{L_1} & \frac{1}{L_1} + \frac{1}{L_2} & -\frac{1}{L_2} & 0 \\ 0 & -\frac{1}{L_2} & \frac{1}{L_2} + \frac{1}{L_3} & -\frac{1}{L_3} \\ 0 & 0 & -\frac{1}{L_3} & \frac{1}{L_3} \end{pmatrix} \cdot \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} dy_0 \\ 0 \\ 0 \\ dy_3 \end{pmatrix} - \begin{pmatrix} \frac{L_1}{3} & \frac{L_1}{6} & 0 & 0 \\ \frac{L_1}{6} & \frac{L_1}{3} + \frac{L_2}{3} & \frac{L_2}{6} & 0 \\ 0 & \frac{L_2}{6} & \frac{L_2}{3} + \frac{L_3}{3} & \frac{L_3}{6} \\ 0 & 0 & \frac{L_3}{6} & \frac{L_3}{3} \end{pmatrix} \cdot \begin{pmatrix} \frac{M_{1_0}}{EI} \\ \frac{M_{1_1}}{EI} \\ \frac{M_{1_2}}{EI} \\ \frac{M_{1_3}}{EI} \end{pmatrix}$$

$$rez = Find(y_1, y_2, dy_0, dy_3)$$

$$rez = \begin{pmatrix} -6,746 \cdot 10^{-8} \\ -9,376 \cdot 10^{-8} \\ 2,491 \cdot 10^{-8} \\ 2,11 \cdot 10^{-8} \end{pmatrix}$$

$$y = \begin{pmatrix} y_0 \\ rez_0 \\ rez_1 \\ y_3 \end{pmatrix} \quad y = \begin{pmatrix} 0 \\ -6,746 \cdot 10^{-8} \\ -9,376 \cdot 10^{-8} \\ 0 \end{pmatrix} \quad y_i$$



Таким образом, в конце расчетов по методу начальных параметров и методу конечных элементов мы свели к построению графиков, которые позволили нам визуальное представить воздействие теплового поля на проводник и тем самым изначально учесть при проектировании слабые участки.

УДК 681.3

ОСОБЕННОСТИ ОРГАНИЗАЦИИ МОДЕЛИРОВАНИЯ СТОХАСТИЧЕСКИХ СЕТЕЙ В МНОГОЗАДАЧНЫХ СРЕДАХ

Воронов В.П., Муравьев Г.Л.

УО «Брестский государственный технический университет», г. Брест

Одной из задач, решаемых при программной организации имитационного моделирования, является реализация квазипараллельного выполнения процессов, когда исполнение процессов параллельно на уровне модели, но последовательно на уровне аппаратного обеспечения.

Для организации параллельных вычислений используют параллельные языки программирования (например, Modula, Ada, Algol-68) либо языки, расширенные соответствующими высокоуровневыми библиотеками и интерфейсами с переносом конструкций параллелизма с языкового уровня на уровень операционной системы (ОС) [1, 2].

Современные многозадачные ОС позволяют представлять выполняемые задачи в виде частей (нитей), выполняемых параллельно и взаимодействующих в едином глобальном пространстве. Пользователю при этом доступны развитые средства управления взаимодействием задач и нитей. Это специализированные библиотеки (например, MPI, PVM; классы объектно-ориентированной библиотеки MFC для управления потоковой многозадачностью системы Visual Studio C++), ориентированные на использование процессоров в режиме MIMD-машины, позволяющие управлять потоками, синхронизацией потоков, семафорами.

Представляется целесообразным реализацию параллельного выполнения процессов модели возложить на многозадачную ОС [3]. Это позволит повысить прозрачность исполнимого кода, упростить реализацию модельных описаний и, с учетом управления ОС переключением процессов на уровне блоков процессора, а также с учетом тенденции производителей процессоров для персональных ЭВМ к развитию многоядерной архитектуры, снизить трудоемкость реализации моделей.

В работе указанная задача рассматривается применительно к организации имитационного моделирования стохастических сетевых моделей (СМ), рассматриваемых как совокупность управляющих и обслуживающих узлов и потоков обслуживаемых заявок [3]. Для распараллеливания вычислений используется параллелизм задач.

Задача сводилась:

- к анализу возможностей операционных систем в части распараллеливания;
- выбору вариантов декомпозиции модели на параллельно-функционирующие компоненты, представимые параллельными компонентами операционной системы;
- разработке алгоритмов и способов отображения имитационных моделей ОС-компонентами (включая связывание компонентов в единую структуру, обеспечение их информационного обмена, расчет характеристик модели и т.п.).

В качестве многозадачных ОС рассматривалось семейство систем UNIX (стандарт POSIX) и Unix-подобные системы, где работа с нитями определена стандартами (POSIX 1003.1c-1995, POSIX 1003.1-1988) и поддерживается соответствующими библиотеками типа Pthreads [4].

В окончательном варианте в качестве программной платформы для реализации моделирования была выбрана Unix-подобная ОС FreeBSD 7.2, отличающаяся полной реализацией системных вызовов, регулируемых стандартом POSIX.

Указанный стандарт регламентирует реализацию ряда объектов. Это мьютекс – простейший двоичный семафор, который может находиться в отмеченном (открытом) или неотмеченном (закрытом) состоянии, связывается и управляется нитью процесса (владельцем мьютекса). Это условная переменная - объект, который может использоваться для оповещения одной приостановленной нити другой нитью о наступлении некоего условия, после чего нить может продолжить работу.

В качестве перспективных подходов к организации моделирования предлагаются алгоритмы: на базе мьютексов и условных переменных; на базе конвейеров ОС. В работе выделены и рассмотрены особенности двух способов декомпозиции модели на нити: по потокам обслуживаемых заявок и по обслуживающим узлам.

В первом случае предполагается транзактное представление параллельностей в модели (в отдельную нить объединялись узлы, участвующие в обслуживании заявок одного класса), что характерно для функционального описания модели. Данный способ оправдывает себя при отсутствии общих узлов, через которые проходят заявки разных классов. При их наличии сложность алгоритма неоправданно растет из-за управления синхронизацией потоков заявок.

Во втором случае предлагается процессное представление параллельностей в модели (в отдельную нить выделяется функционирование каждого узла), что характерно для структурного описания модели. Здесь средства синхронизации нитей выделяются в отдельную сущность – канал, рассматриваемый как общая для нескольких нитей область памяти. Корректность работы с данными в этой области гарантируется использованием встроенных средств синхронизации.

Ниже приведен упрощенный алгоритм работы канала, связывающего нить-источник А и нить-приемник Б, обеспечивающий бесконечную передачу данных только в одну сторону (полудуплекс), что достаточно для моделирования СМ [5]. Пусть канал инициализирован и нити запущены.

Тогда алгоритм передачи данных нитью А включает шаги: 1. Сгенерировать данные. 2. Открыть мьютекс mutex. 3. Загрузить данные в область buf. 4. Установить флаг ready. 5. Послать сигнал через cond. 6. Закрыть мьютекс. 7. Вернуться на пункт 1.

Алгоритм приема данных нитью Б включает шаги: 1. Открыть мьютекс. 2. Ждать, пока не придет сигнал через cond и не будет установлен флаг ready. 3. Обработать данные из buf. 4. Сбросить флаг ready. 5. Закрыть мьютекс. 6. Вернуться на пункт 1.

Здесь канал информационно отображается атрибутами: cond - условная переменная, информирующая нить Б об изменении состояния канала нитью А; mutex – мьютекс для защиты переменной cond; ready – флаг (признак) освобождения канала для защиты от некорректной посылки сигналов; buf – область памяти, общая для А и Б.

Заявкам стохастической сети прикрепляется модельное время, устанавливаемое генератором сети, которое модифицируется при прохождении каждого узла. При этом узел для каждой заявки сохраняет ряд атрибутов, включая модельное время входа и выхода заявки из узла, необходимых для проведения моделирования и для вычисления характеристик модели.

При использовании для процессного представления параллельностей конвейеров ОС предлагается каждый блок ССМ отображать отдельной программой, передающей и принимающей заявки в виде текстовых строк через стандартные потоки ввода-вывода, а узлы связывать с помощью конвейера ОС (например, на языке shell) [5]. Это позволит передавать данные из процесса-источника в процесс-приемник в виде строк, а синхронизацию процессов возложить на ОС. Для отображения маршрутных узлов ССМ предлагается использовать именованные конвейеры, представляемые файлами.

Таким образом, проведено исследование и макетирование алгоритмов и подходов к имитации стохастических сетевых моделей на языках С и shell с использованием библиотек ОС Unix. Установлено, что наибольшая эффективность моделей достигается при использовании процессного подхода, реализуемого на базе нитей библиотеки Pthread. При этом инструменты shell рекомендуется использовать на этапе прототипирования, т.к. здесь модели на порядок медленнее из-за использования операций ввода-вывода.

Литература

1. Воеводин, В.В. Параллельные вычисления / В.В. Воеводин, Вл.В. Воеводин. – СПб.: BHV-Петербург, 2002. – 609 с.
2. Одинцов, И.О. Профессиональное программирование. Системный подход / И.О. Одинцов. - СПб.: BHV-Петербург, 2002. – 512 с.
3. Воронов, В.П. Имитационное моделирование в многозадачных ОС // Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях: материалы XII респ. науч. конф. студентов и аспирантов, Гомель, 16-18 марта 2009 г.: в 2 ч. / ГГУ им. Ф.Скорины; редкол.: О.М. Демиденко [и др.]. – Гомель, 2009. – Ч. 2. – С. 17–18.
4. Многопоточное программирование. – Режим доступа: http://rk6.bmstu.ru/electronic_book/progrws/multithread.html. – Дата доступа: 25.01.2008.
5. Стивенс, У.Р. UNIX: взаимодействие процессов / У.Р. Стивенс, – СПб.: Питер, 2003. – 188 с.

УДК 528.94:551.509.25

МЕТОДИКА ПЕРЕСЧЕТА ПРОГНОЗНЫХ ЗНАЧЕНИЙ СНЕГОВОЙ НАГРУЗКИ ДЛЯ РАЗЛИЧНЫХ ПЕРИОДОВ ПОВТОРЯЕМОСТИ

Дмитриева А.В.

УО «Брестский государственный технический университет», г. Брест

Путем статистической обработки эмпирических данных о снеговой нагрузке на поверхности земли, нами ранее установлено, что на территории Беларуси экстремальные значения этого параметра могут подчиняться трем основным типам вероятностных распределений – Гумбеля (тип I), Вейбулла (тип II) и Фреше (тип III) [1,2]. В этом случае пересчет характеристических значений снеговой нагрузки с обеспеченностью 0.98 (для периода повторяемости 50 лет), на меньшие или большие значения обеспеченности (соответственно, меньшие или большие периоды повторяемости), выполняемый стандартным методом (по зависимости Гумбеля) [3], не может быть корректным для всей территории Беларуси. В связи с этим, была поставлена задача районирования территории по типу (закону) распределения эмпирических распределений снеговой нагрузки.

Для решения данной задачи использование любых методов гритинга применительно к параметру «тип распределения» будет некорректным, так как этот параметр принадлежит к номинальной («слабой») шкале измерений. Однако каждому типу распределения можно поставить в однозначное соответствие числовой параметр «достоверность аппроксимации», принадлежащий к самой сильной, абсолютной шкале. Таким образом, необходимо построить т.н. диаграмму преобладания – тематическую карту, иллюстрирующую превышение значения одной из анализируемых характеристик над другими для конкретных объектов. Тип распределения (I, II или III) для каждой метеостанции выбирался из наибольшей достигнутой достоверности аппроксимации данных (R^2), поэтому для каждой метеостанции имеется набор из трех значений достоверности (R_I^2 , R_{II}^2 , R_{III}^2).

Мы использовали данные наборы значений достоверности для построения трех поверхностей параметра $-\ln(R^2)$, соответствующих трем типам распределения. Предварительное логарифмирование значения достоверности обусловлено необходимостью

некоторой линейаризации параметра в окрестности значения 1.00 (известно, что при хорошей аппроксимации $R^2 \rightarrow 1$). После наложения всех трех поверхностей, тип распределения в любой промежуточной точке может быть найден по тому условию, какая из поверхностей в данной точке расположена выше. Полученная указанным способом карта типов представлена на рисунке 1.

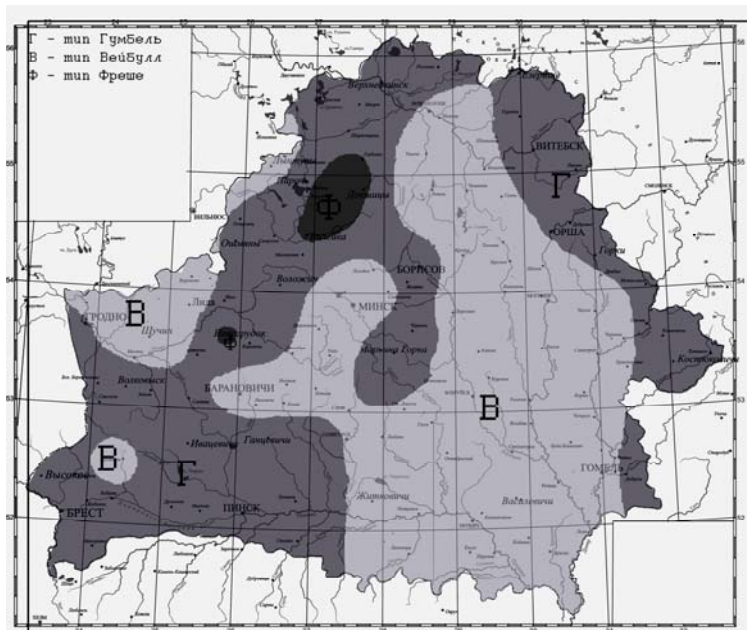
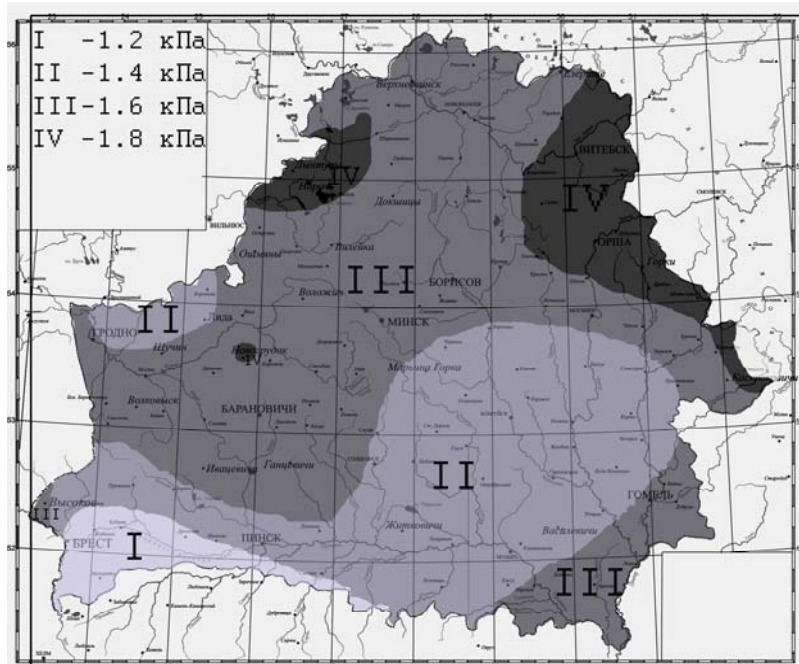


Рисунок 1 – Карта районирования территории Беларуси по типу распределения годовых максимумов снеговой нагрузки

Далее, используя карту районирования территории Республики Беларусь по снеговой нагрузке s_{50} , разработанную для периода повторяемости $T_{ret} = 50$ лет (см. рис. 2), и предыдущую карту (см. рис. 1), выполняем разбиение территории на подрайоны с одинаковой снеговой нагрузкой и типом распределения. В результате такого разбиения получено 9 различных подрайонов (всего было возможно 12 комбинаций «снеговой район – тип распределения», однако некоторые из комбинаций в данном конкретном случае не имели места).

Рисунок 2 – Карта районирования характеристических значений снеговой нагрузки (период повторяемости 50 лет)



Для каждого из подрайонов находилась общая зависимость $s_T = s_T(T_{ret})$, такая, чтобы найденное с ее помощью прогнозное значение имело достаточную обеспеченность расчетных значений снеговой нагрузки для любой из метеостанций, попадающей в данный подрайон. Общий вид зависимости (линейная, экспоненциальная или логарифмическая) при этом определялся типом распределения максимумов нагрузки для данного района (соответственно, Гумбеля, Фреше или Вейбулла). Расчетные значения определялись путем умножения характеристического значения на рекомендуемый нормами проектирования коэффициент запаса 1.5 (2.0 для особых подрайонов 1.6Ф и 1.8Ф). Пример определения искомой зависимости для подрайона 1.4В, в который попадает четыре метеостанции (Василевичи, Житковичи, Пружаны, Лида), представлен на рисунке 3, где оси абсцисс соответствует параметр $x_T = -\ln[-\ln(1 - 1/T_{ret})]$.

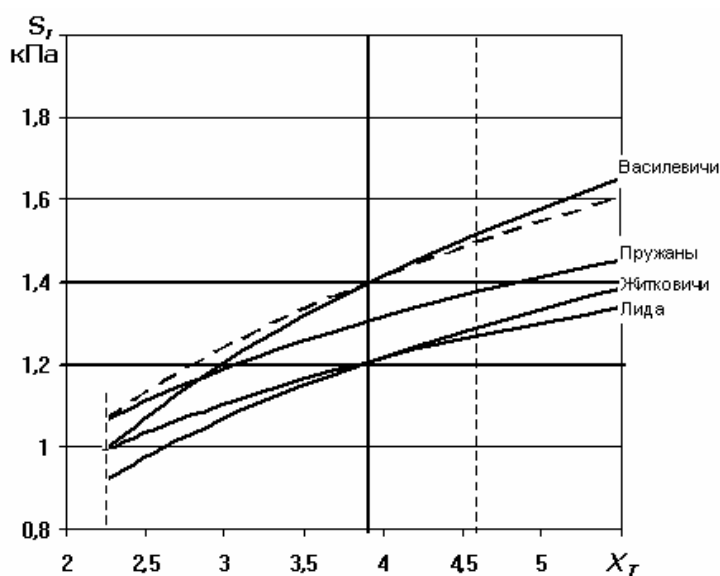


Рисунок 3 – Логарифмическая зависимость $s_T = s_{50} + k \cdot \ln(x_T/x_{50})$ для подрайона 1.4В (характеристическое значение $s_{50} = 1.4$ кПа, распределение Вейбулла). Искомая зависимость представлена пунктирной линией. Границы, соответствующие диапазону $T_{ret} = 10 \dots 100$ лет : $x_{10} = 2.25, x_{100} = 4.60$

В приведенном примере зависимость ищется в виде $y = a + k \cdot \ln x$. Однако, поскольку для района установлено характеристическое значение нагрузки, фиксировано прохождение кривой через точку с координатами (1.4; 3.90), так как $x_{50} = -\ln[-\ln(1-1/50)] = 3.90$. Конкретная зависимость для подрайона, таким образом, будет определяться единственным коэффициентом (в данном случае $k = 0.60$). Указанный коэффициент варьируется таким образом, чтобы достичь наилучшей обеспеченности расчетных значений снеговой нагрузки во всем диапазоне $T_{ret} = 10 \dots 100$ лет. Мы полагали, что соответствующий период повторяемости расчетных значений должен не менее чем в пять раз превышать период повторяемости характеристических значений (составлять, для $T_{ret} = 10$ лет и $T_{ret} = 100$ лет, соответственно, 50 и 500 лет, или, иначе, достигать обеспеченности не менее 0.98 и 0.998).

Найденные для всех подрайонов итоговые зависимости были сведены в таблицу, пригодную для практического использования. Таблица содержит всю необходимую (для пересчета на иные периоды повторяемости) информацию и дополнительных пояснений не требует. Прочерки в трех графах таблицы означают, что на территории Республики Беларусь соответствующие подрайоны отсутствуют.

Таблица 1 – Зависимости и коэффициенты для пересчета характеристических значений снеговой нагрузки на периоды повторяемости T , отличные от 50 лет

Тип вероятностного распределения и вид зависимости для пересчета	Значение коэффициента пересчета k для снеговых районов			
	I ($s_{50} = 1.2 \text{ кПа}$)	II ($s_{50} = 1.4 \text{ кПа}$)	III ($s_{50} = 1.6 \text{ кПа}$)	IV ($s_{50} = 1.8 \text{ кПа}$)
Гумбеля, $s_T = s_{50} + k \cdot (x_T - x_{50})$	0,20	0,20	0,23	0,23
Вейбулла, $s_T = s_{50} + k \cdot \ln(x_T / x_{50})$	-	0,60	0,60	0,60
Фреше, $s_T = s_{50} \cdot \exp[k \cdot (x_T - x_{50})]$	-	-	0,25	0,25
$x_T = -\ln[-\ln(1-1/T)]$, $T = 10 \dots 100$ [лет], $x_{50} = -\ln[-\ln(1-1/50)] \approx 3.902$				

Литература

1. Тур, В.В. Нормирование снеговых нагрузок для территории Республики Беларусь / В.В. Тур, В.Е. Валуев, С.С. Дереченник, О.П. Мешик, И.С. Воскобойников // Строительная наука и техника (Минск). – 2008, №2. – С. 27 – 45.
2. Дмитриева, А.В. Итерационный алгоритм гридинга с покоординатной аппроксимацией для районирования метеорологических данных/ А.В. Дмитриева // Сборник конкурсных научных работ студентов и магистрантов. – Брест, 2009. – Ч. 1. – С. 148 – 152.
3. Снеговые нагрузки на покрытиях зданий в условиях Севера / В.В. Филиппов, А.Т. Копылов, Т.А. Корнилов и др. – М.: Наука, 2000. – 246 с.

УДК 004

ПОСТРОЕНИЕ МАТЕМАТИЧЕСКОЙ МОДЕЛИ АВТОМАТИЗИРОВАННОГО ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА

Емельянченко Н.С.

УО «Гомельский государственный технический университет имени П.О. Сухого», г. Гомель

На любом предприятии или организации используется оборудование, которое находится в постоянной эксплуатации. На производстве выгодно, чтобы было обработано наибольшее количество сырья, но без простоя оборудования и без его перегрузки, в результате которого оборудование может выйти из строя. Поэтому актуальным вопросом является определения наилучшего использования ресурсов автоматизированного технологического процесса с учетом интереса производства.

Предмет моделирования – функционирование автоматизированного участка обработки деталей, состоящего из входного конвейера, транспортного робота и двух станков для обработки деталей. Транспортный робот выполняет весь цикл переноски деталей двух типов на два различных станка. Каждая деталь в зависимости от типа переносится разное количество времени, т.к. обслуживающие их станки находятся на разном расстоянии от входного конвейера. Детали могут образовывать очереди как перед перевозкой на транспортном роботе, так и перед обработкой на станках. После перевозки и обработки детали поступают на выходной конвейер (рисунок 1).

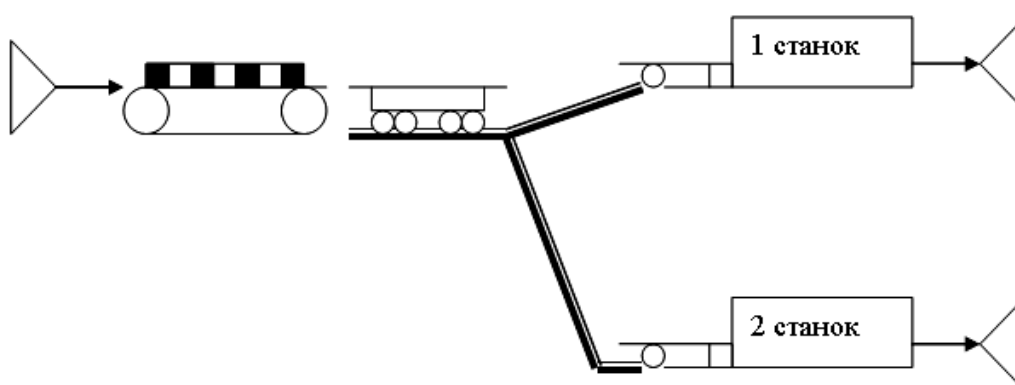


Рисунок 1 – Структурная схема объекта моделирования

Цель моделирования: изучение влияния интенсивности поступления деталей на загрузку транспортного робота и станков обработки деталей позволит:

- определить наиболее благоприятные параметры обработки сырья;
- избежать перегрузки и простоя оборудования;
- использовать ресурсы с учетом интереса производства.

Воспользуемся принципом декомпозиции: отдельными элементами модели будут очереди, транспортный робот, станки (в дальнейшем этапы обработки).

Параметры системы - интенсивность поступления деталей на транспортировку и обработку на станках (λ_i) ($i = 1..3$).

Переменными модели системы являются функции распределения длительности перевозки или обработки детали на i -том этапе $F_i(o)$.

Статистики моделирования: коэффициенты загрузки этапов обработки (z_i); количество обработанных деталей на участке (v); размер очереди к каждому этапу обработки (l_i); общее время обработки деталей на i -ом участке обработки (T_i).

Уточнение критериев эффективности. В качестве критериев эффективности могут выступать коэффициент загруженности этапов обработки (η_i) и среднее значение времён обработки деталей (T_i). В результате моделирования нужно найти следующие функциональные зависимости: $\eta_i = \varphi_i(\lambda_i)$ и $T_i = \Psi_i(\lambda_i)$.

Аппроксимация реальных процессов математическими величинами. При задании функций распределения $F_i(o)$ длительность обработки детали на i -ом этапе обработки достаточна аппроксимация дискретными функциями.

Принимаемые гипотезы: φ_i и Ψ_i имеют вид полиномов, порядок и значение коэффициентов которых необходимо будет определить.

Установление основной структуры моделирования. Необходимо прогнозировать характеристики загрузки этапов обработки (z) и времена обработки деталей на этапе обработки (T_i) в зависимости от изменяющихся характеристик входного потока деталей (λ_i) при заданных $F_i(o)$.

Исследование осуществляется с помощью имитационного моделирования. Имитационная модель представляет собой набор активностей AK_{ij} , каждая из которых после выполнения на ЭВМ возвращает управление управляющей программе моделирования. Формализация осуществляется поэтапно:

1. Осуществляется поиск места для поступающей детали: это либо новая строка, либо место ушедшей детали. Также происходит генерация детали, которая зависит от параметров табличного распределения и определяется тип детали, зависящий от значения параметра равномерного распределения.

2. Производится процесс перевозки детали роботом к различным станкам в зависимости от типа деталей, обрабатываемых ими, и осуществляется пополнение очередей деталей к станкам и отправка робота к входному конвейеру.

3. Осуществляется перевод робота из состояния «активен» в состояние «свободен» и подготовка его к перевозке следующей детали, обслуживание деталей на станках и завершения обслуживания. Также определяется среднее время пребывания детали в системе. Происходит завершение процесса обслуживания деталей и остановка всей системы.

Согласно принятым предложениям было разработано соответствующее ПО и проведена его верификация, в ходе которой определили, что достоверные результаты моделирования можно получить при модельном времени более 200 минут.

Для построения имитационной модели процесса, проверки ее адекватности и для оценки влияния на процесс каждого учитываемого технологического фактора использовался регрессионный анализ. Для получения регрессионных зависимостей был реализован В-план второго порядка [1]. Уровни и интервалы варьирования факторов представлены на рисунке 1. Контролируемыми факторами эксперимента были выбраны коэффициент загрузки робота КЗР, коэффициент загрузки 1 станка и коэффициент загрузки 2 станка.

Наименование фактора	Обозначения		Интервал варьирования фактора	Уровень варьирования фактора		
	Натур.	Норм.		Нижний (-)	Основной (0)	Верхний (+)
Количество деталей, K	K	X_1	5	295	300	305
Время в системе, T	T	X_2	2,4	37,62	40,02	42,42
Длина очереди, L	L	X_3	1,8	18	19,8	21,6

Рисунок 2 – Основные факторы и уровни их варьирования

В результате реализации схемы регрессионного анализа и оценки степени влияния каждого входного технологического фактора на выходную величину была получена следующая регрессионная модель:

$$K3P = 0,9658 + 0,0001K + 0,0003T + 0,0004L + 0,0000024K^2 + 0,0007T^2 + L^2 + 0,00045K_T + 0,034T_L + 0,00045K_L \quad (1)$$

Проанализируем по полученной модели влияние на отклик факторов К и Т и взаимодействия КТ. Зафиксировав фактор L с ростом значений количества деталей, величина времени в системе также увеличивается. При этом с увеличением значений фактора Т – времени в системе увеличивается практически линейно. Отсюда следует, что при больших значениях времени продолжительность обработки будет оказывать более сильное влияние на коэффициент загрузки.

Аналогично рассмотрим влияние факторов К и L и их взаимодействия. Влияние фактора К – количества деталей – сохраняет свой вид, но чем больше значения фактора L – длины очереди, тем более возрастает влияние количества деталей на значения коэффициента загрузки. Также очевидно, что и при больших значениях количества деталей влияние как времени в системе, так и длины очереди на коэффициент загрузки возрастает.

Проанализируем и взаимодействие факторов Т и L. Очевидно, что как с увеличением значений фактора Т влияние длины (L) на коэффициент загрузки усиливается, так и при увеличении значений фактора L усиливается влияние времени в системе на коэффициент загрузки.

На заключительном этапе была решена задача о поиске времени перевозки деталей роботом, при котором средняя длина очереди перед станками будет больше нуля и меньше наперед заданной конкретной величины (рис. 3 - 4). Решение данной задачи даст возможность избежать перегрузки оборудования.

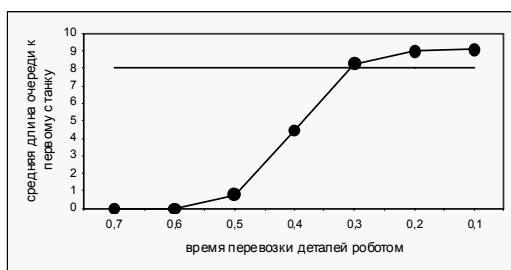
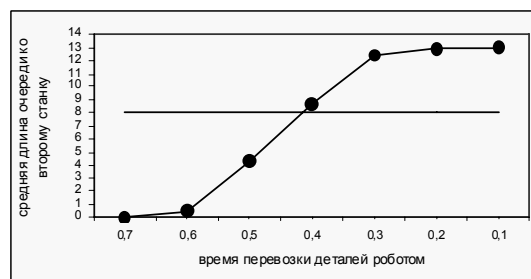


Рисунок 3 – Зависимость изменения средней длины очереди к первому станку от изменения времени перевозок деталей роботом

Рисунок 4 – Зависимость изменения средней длины очереди ко второму станку от изменения времени перевозок деталей роботом



Из рисунка 3 видно, что критическое значение длины очереди к первому станку достигается при времени перевозки детали первого типа роботом равном 0.315, а нулевое – при 0.6. Рисунок 4 показывает, что критическое значение длины очереди ко второму станку достигается при времени перевозки детали второго типа роботом равном 0.43, а нулевое – при 0.7.

Выводы:

1. При увеличении количества прогонов величина доверительного интервала постепенно уменьшается и стремится к нулю.

2. При оценке чувствительности имитационной модели выявили, что отклик чувствителен к изменениям переменной на интервале ее варьирования, т.к. коэффициент корреляции R^2 между значениями коэффициента загрузки робота и временного интервала перевозки им деталей составил 0,9337, что очень близко к единице.

3. В результате постановки имитационного эксперимента по сравнению альтернатив можно сформулировать выводы о наилучшем использовании ресурсов автоматизированного участка обработки деталей с учетом интересов производства. Анализ полученной математической модели позволит определить параметры оборудования для обработки деталей с целью наименьшего износа оборудования.

Литература

1. Лоу, А.М. Имитационное моделирование / А.М. Лоу, В.Д. Кельтон. СПб.: Издательский дом ПИТЕР, 2004. - 848 с.

УДК: 004.652.5

ОБЪЕКТНО-РЕЛЯЦИОННАЯ МОДЕЛЬ ДАННЫХ

Лысюк А.Н

УО «Брестский государственный технический университет», г. Брест

Разработка любой информационной системы основана на построении модели данных. Базовыми понятиями в модели данных являются объекты и отношения между ними. Спроектированная модель ложится в основу базы данных (включается в концептуальный и внешний уровни ее архитектуры). Качество проектирования модели данных непосредственно влияет на качество работы с конкретной базой данных.

В настоящее время широко распространена реляционная модель данных. Основной структурой данных в реляционной модели является отношение, как подмножество декартова произведения, представимое в виде таблицы. Каждая строка этой таблицы представляет запись (кортеж), а каждый столбец соответствует атрибуту отношения из заданного домена. В таблице не может быть двух одинаковых записей. Таким образом, реляционная модель данных представляется множеством таблиц-отношений.

Иерархические связи между отношениями поддерживаются неявным образом. В каждой связи одно отношение - основное, а другое – подчиненное. Таким образом, один кортеж основного отношения может быть связан с несколькими кортежами подчиненного отношения. Для поддержки этих связей оба отношения должны содержать набор атрибутов, по которым они связаны. В основном отношении это первичный ключ (primary key), в подчиненном отношении это внешний ключ (foreign key). Над множеством отношений определен ряд операций. В алгебре Кодда определены 8 операций, которые делятся на две группы: теоретико-множественные и специализированные операции.

Основными достоинствами реляционной модели данных является: полная независимость данных, строгий математический аппарат. Недостатками этой модели являются трудность понимания структуры данных, низкая скорость доступа, большой объем внешней памяти. Кроме того, не всегда предметную область можно представить в виде совокупности таблиц.

Рассмотрим следующий пример. При реализации, средствами языка С++, информационной системы, которая должна выполнять расчет численности профессорско-преподавательского состава кафедр университета, возникла необходимость в представлении объекта «Академическая группа». Для данной сущности был определен набор существенных в рамках рассматриваемой модели атрибутов. Кроме того, при проведении лабораторных занятий группа делится на подгруппы, а механизм деления зависит от нескольких факторов. Целесообразно реализовать деление на подгруппы в виде отдельной функции-метода.

Способ реализации рассматриваемой функции заключается в следующем: создается класс Group, в этом классе инкапсулируются атрибуты, а также функции доступа к данным атрибутам (интерфейс). Затем в этот класс добавляется метод деления группы на подгруппы (DivideBySubGroup). Схема реализации показана на рисунке 1.

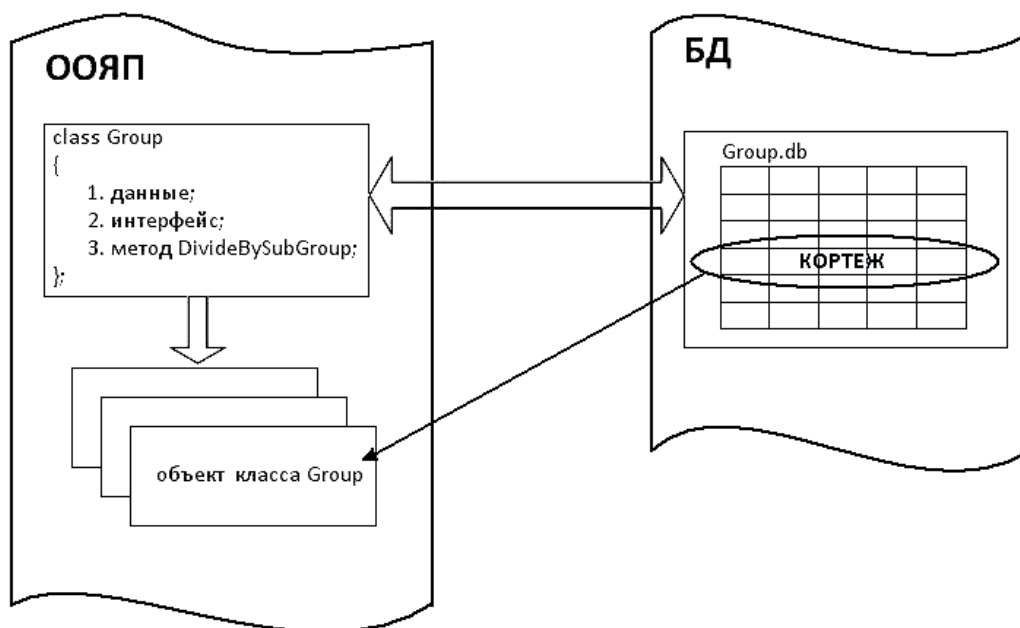


Рисунок 1 – Взаимодействие объектно-ориентированного языка программирования с реляционной базой данных

При таком подходе обнаруживается несоответствие, связанное с тем, что модели данных, используемые в программировании, отличаются от моделей данных СУБД, что влечет за собой необходимость поддерживать соотношения и взаимосоответствия между кортежами базы данных и объектами в программировании. Кроме того, при таком подходе нарушается семантическая целостность данных. В данной ситуации решением является использование объектно-ориентированной модели данных.

Объектно-ориентированные базы данных явились результатом совмещения принципов объектно-ориентированного программирования и принципов управления базами данных. В одной системе объединены понятия инкапсуляции, полиморфизма, наследования из объектно-ориентированного программирования и атомарности, целостности, изоляции из баз данных. В результате мы имеем возможность управлять большими объемами информации при помощи объектно-ориентированного подхода.

Объектно-ориентированная база данных (ООБД) способна хранить объекты в том же виде, в котором они будут доступны для языка программирования. Это обеспечивается за счет того, что объекты в ООБД принадлежат классу, имеющему в своем составе на-

бор атрибутов, выражаемых простыми типами данных или другими классами. К классам применяются правила наследования, несущие в себе преимущества ООП: полиморфизм, переопределение наследованных методов и возможность динамической привязки.

Преимущества:

1. Объекты могут хранить произвольное количество простых типов и других объектов.
2. Объект является наилучшей моделью отображения реального мира, нежели реляционные картежи.
3. Для доступа к данным из систем управления ООБД не обязателен отдельный язык запросов, поскольку доступ происходит непосредственно к объектам.

Недостатки:

1. Изменение схемы данных в результате создания, изменения или удаления таблиц обычно зависит от приложения.
2. В объектно-ориентированном программировании отсутствуют общие средства манипулирования данными, такие как реляционная алгебра или реляционное счисление. Работа с данными ведется с помощью одного из объектно-ориентированных языков программирования общего назначения (Python, Java, C#, Visual Basic .NET, C++, Objective-C).

Очевидно, что оба указанных недостатка связаны с отсутствием развитых средств манипулирования данными. Эта задача решается двумя способами – расширение объектно-ориентированных языков в сторону управления данными (стандарт ODMG) либо добавление объектных свойств в реляционные СУБД (SQL-3, а также так называемых объектно-реляционных СУБД).

Основой предлагаемого подхода является объединение объектного кода с соответствующим отношением реляционной модели, при этом файл базы должен состоять из четырех блоков (рисунок 2).

Заголовок файла
Блок метаданных
Отношение - таблица
Объектный код

Рисунок 2 – Структура файла базы данных

В блоке метаданных необходимо описать реализуемый класс, указать свойства и прототипы методов. Это упростит дальнейшую интеграцию базы данных при разработке приложения. Кроме того, необходимо, чтобы объектный код был платформонезависимым. Лучшим вариантом является использование промежуточного языка платформы .NET.

Развитие данного подхода в дальнейшем позволит устранить недостатки, связанные с использованием только реляционной либо только объектно-ориентированной базы данных.

Литература

1. Кузнецов, С. Д. Основы баз данных / С.Д. Кузнецов – 1-е изд. – М.: Интернет-университет информационных технологий - ИНТУИТ.ру, 2005. – С. 38-58.
2. Когаловский, М.Р. Энциклопедия технологий баз данных / М.Р. Когаловский – М.: Финансы и статистика, 2002. – С. 800.
3. Тяпкин, С.В. Преимущества и недостатки объектно-ориентированных баз данных / С.В. Тяпкин // Журнал научных публикаций аспирантов и докторантов. – 2007.

УДК 004.9(075)

ТЕСТИРОВАНИЕ ДАННЫХ ДЛЯ АНАЛИЗА ФАКТОРОВ, ВЛИЯЮЩИХ НА ДИНАМИКУ ВВП

Миколенко Е. А.

УО «Брестский государственный технический университет», г. Брест

В настоящее время имеются развитые инструментальные средства обработки статистических данных. Вместе с тем их использование достаточно ограничено из-за необходимости освоения специального математического аппарата. Поэтому предлагается технология освоения такого рода инструментальных средств.

Так, на сегодняшний день в Республике Беларусь, в условиях принятия активных мер по привлечению иностранных инвестиций в экономику со стороны государства, достаточно важным и актуальным является вопрос о том, в какой степени валовой внутренний продукт (ВВП) увеличивается за счёт привлечения иностранных инвестиций, а также какая доля его роста приходится на собственные ресурсы государства. Ответ на этот вопрос позволит определить целесообразность принимаемых шагов в обозначенном направлении. Т. е. на макроэкономическом уровне появляется необходимость вывести математическую зависимость между объёмами привлечения в экономику государства иностранных инвестиций и динамикой ВВП.

Нами анализировалось влияние следующих факторов на уровень ВВП:

- общей величины иностранных инвестиций;
- прямых иностранных инвестиций (ПИИ);
- портфельных иностранных инвестиций;
- прочих иностранных инвестиций;
- инвестиций в основной капитал, как за счёт внутренних, так и внешних (иностран-ных) источников;
- среднегодовой численности населения, занятого в экономике.

Для решения поставленной задачи использовались возможности математической статистики в системе MathCAD.

Для анализа имеющихся данных и для построения полиномиальной регрессии использовалась встроенная функция MathCAD **regress**, которая позволяет на базе экспериментальных данных получить коэффициенты полиномиальной регрессионной функции. У функции **regress** есть три параметра. Первые два – это экспериментальные данные, для которых строится аналитическая приближённая зависимость (они должны быть записаны в виде вектор-столбцов). Третий параметр задаёт степень полиномов, которые будут использоваться в регрессии.

Выбор полиномиальной регрессионной функции обусловлен возможностью описания практически любой зависимости с её помощью.

Кроме того, рассчитывался коэффициент детерминации R^2 , и коэффициент Дарбина-Уотсона (DW), характеризующий качество исходных данных. Этот коэффициент должен попадать в диапазон от 1,5 до 2,5 ($1,5 < DW < 2,5$).

Статистические данные по анализируемым факторам, которые в своём первоначальном виде не были пригодны для проведения расчётов и осуществления исследования, были обработаны соответствующим образом для получения удовлетворительного значения коэффициента DW.

Так, для установления зависимости между динамикой ВВП и объёмами поступающих иностранных инвестиций уровень ВВП, заданный в млрд. бел. Рублей, был переведён в млн. долл. США с учётом средневзвешенного годового курса доллара к белорусскому рублю.

В результате была выведена полиномиальная регрессионная функция, отражающая зависимость динамики ВВП от общей величины иностранных инвестиций и позволяющая при заданном значении общей величины иностранных инвестиций прогнозировать соответствующий им уровень ВВП:

$$y = -1,566 \cdot 10^3 + 24,064x - 5,597 \cdot 10^{-3} x^2 + 5,032 \cdot 10^{-7} x^3,$$

где x - общая величина иностранных инвестиций;

y – уровень ВВП, соответствующий данной величине иностранных инвестиций.

Аналогичным образом были получены регрессионные функции зависимости ВВП от темпов роста поступления в страну ПИИ и портфельных иностранных инвестиций.

В силу того, что данные о динамике поступления иностранных инвестиций в экономику Республики Беларусь приведены лишь за шестилетний период, при проведении расчётов на основании абсолютных значений в последних двух случаях были получены неудовлетворительные значения коэффициентов детерминации и Дарбина-Уотсона. В связи с этим мы в своих расчётах перешли к темпам роста.

Зависимость динамики ВВП от темпов роста поступления в экономику страны ПИИ (анализ за 2003-2007 гг.):

$$y = 0,494 + 2,766x - 2,884x^2 + 1,158x^3 - 0,16x^4$$

Зависимость динамики ВВП от темпов роста поступления в экономику страны портфельных инвестиций (анализ за 2003-2007 гг.):

$$y = -25,636 + 115,373x - 163,549x^2 + 79,94x^3 - 5,198x^4$$

Далее проводились исследования в области определения влияния динамики инвестиций в основной капитал на уровень ВВП и установления регрессионной зависимости между этими показателями. С этой целью имеющиеся исходные данные, приведенные в текущих ценах, переводились в доллары США с учётом средневзвешенного годового курса долл. США (бел. руб. к доллару).

Полученные функции полиномиальной регрессии, отражающие взаимосвязь между динамикой ВВП и объёмами инвестиций в основной капитал, объёмами инвестиций в основной капитал за счёт внутренних источников и объёмами инвестиций в основной капитал за счёт внутренних источников имеют соответственно следующий вид:

$$- y = 3,9 + 4,072 \cdot x - 0,058x^2,$$

где x – объём инвестиций в основной капитал;

y – уровень ВВП, соответствующий данной величине инвестиций в основной капитал;

$$- y = -26,633 + 34,785 \cdot x - 11,149 \cdot x^2 + 1,87x^3 - 0,148x^4 + 4,403 \cdot 10^{-3} x^5,$$

где x - объём инвестиций в основной капитал за счёт внутренних источников;

y – уровень ВВП, соответствующий данной величине инвестиций в основной капитал за счёт внутренних источников;

$$- y = 29,305 - 465,813x + 3,356 \cdot 10^3 x^2 - 5,452 \cdot 10^3 \cdot x^3,$$

где x - объём инвестиций в основной капитал за счёт внешних источников;

y – уровень ВВП, соответствующий данной величине инвестиций в основной капитал за счёт внешних источников.

В таблице представлены значения коэффициента детерминации R^2 , коэффициента Дарбина-Уотсона и параметра k , задающего степень полиномов, полученные в процессе тестирования данных на наличие связи между рассматриваемыми факторами и динамикой ВВП с использованием функции MathCAD regress.

Практически во всех рассмотренных случаях коэффициент Дарбина-Уотсона имеет удовлетворительное значение, что подтверждает достаточно высокое качество данных. Использование инструментальных компьютерных математических пакетов для обработки статистических данных позволяют сделать следующие выводы в области экономики.

Имеется достаточно тесная взаимосвязь (R^2 приближен к 1) между уровнем ВВП и такими факторами, как общий объём иностранных инвестиций, поступающих в экономику Республики Беларусь, темпами роста объёмов ПИИ, темпами роста объёмов портфельных иностранных инвестиций, объёмами инвестиций в основной капитал, инвестиций в основной капитал как за счёт внутренних, так и внешних (иностраных) источников. Рост этих факторов приводит к соответствующему увеличению уровня ВВП.

При этом изменение среднегодовой численности населения, занятого в экономике, не оказывает первостепенного влияния на увеличение уровня ВВП. Это подтверждает малое значение коэффициента детерминации R^2 , которое свидетельствует о наличии незначительной связи между динамикой ВВП и изменением среднегодовой численности населения, занятого в экономике.

Были получены функции полиномиальной регрессии, на основании которых при заданных значениях независимых переменных можно осуществлять прогнозы динамики ВВП от факторов, на него влияющих. На макроэкономическом уровне, т. е. уровне государственного управления экономикой, это позволит достигать желаемого уровня ВВП, в результате чего экономический рост в государстве станет величиной более управляемой.

Таблица – Значения коэффициента детерминации, коэффициента Дарбина-Уотсона и параметра k в зависимости от фактора, влияющего на динамику ВВП.

Факторы, влияющие на динамику ВВП (независимые переменные)	N	k	R^2 наличие связи	DW качество данных
Иностранные инвестиции, всего	6	3	0,956	2,409
ПИИ (абсолютные значения)	6	5	0,607	1,244
ПИИ (темпы роста)	5	4	1	1,604
Портфельные инвестиции (темпы роста)	5	4	1	2,202
Прочие инвестиции (темпы роста)	5	2	0,05	1,54
Инвестиции в основной капитал	10	2	0,992	1,663
Инвестиции в основной капитал за счёт внутренних источников	8	5	0,998	1,885
Инвестиции в основной капитал за счёт иностраных источников	8	3	0,934	2,08
Численность занятых в экономике	16	15	0,091	2,347

В результате проведения исследования проверена пригодность данных для анализа факторов, предложена технология освоения инструментальных статистических средств для решения данной задачи.

Литература

1. MathCad 6.0 Руководство пользователя. – М.: Мир, 1996.
2. Бородич, С.С. Эконометрика. – Мн.: Новое знание, 2001.

УДК 621.396.96

РАЗРАБОТКА АДАПТИВНОГО МНОГОГИПОТЕЗНОГО АЛГОРИТМА КОМПЛЕКСНОЙ ВТОРИЧНОЙ ОБРАБОТКИ ИНФОРМАЦИИ В ИНЕРЦИАЛЬНО-СПУТНИКОВОЙ НАВИГАЦИОННОЙ СИСТЕМЕ

Мороз А.Н.

УО «Военная академия Республики Беларусь», г. Минск

Постановка задачи

В результате первичной обработки спутниковых сигналов на вход комплексного измерителя поступают радионавигационные параметры (псевдодальность и псевдоскорость) для четырех навигационных спутников [2], а также наблюдения блока ориентации и блока акселерометров бесплатформенной инерциальной навигационной системы.

Необходимо получить оценку составляющих вектора состояния потребителя

$$\lambda_k = \left[\mathbf{X}_k^T \quad D'_k \quad \mathbf{V}_k^T \quad V'_k \quad \mathbf{A}_k^T \quad \boldsymbol{\varepsilon}_k^T \quad \mathbf{V}_{\varepsilon,k}^T \right]^T,$$

где \mathbf{X}_k – вектор координат летательного аппарата (ЛА); D'_k – ошибка измерения дальности; \mathbf{V}_k – вектор скоростей ЛА; V'_k – ошибка измерения скорости; \mathbf{A}_k – вектор ускорений ЛА; $\boldsymbol{\varepsilon}_k$ – вектор ошибок оценивания углов Эйлера; $\mathbf{V}_{\varepsilon,k}^T$ – вектор скоростей изменения ошибок углов Эйлера.

Синтез алгоритма комплексирования на основе адаптивного многогипотезного измерителя (МГИ) с перекрестными связями

Принятые гипотезы модели движения потребителя: 1) неподвижен; 2) полет с постоянной скоростью; 3) маневрирует с ускорением.

Алгоритм адаптивной МГИ фильтрации для k -го шага измерения [3]:

1. Вычисляется априорная вероятность i -й ($i, j = \overline{1, M}$) гипотезы:

$$p_{k|k-1}^{(i)} = \sum_{j=1}^M p_{ji} p_{k-1}^{(j)},$$

где p_{ji} – априорные вероятности перехода; $p_{k-1}^{(j)}$ – апостериорная вероятность j -й гипотезы.

2. Определяются условные вероятности перехода p_{k-1}^{ji} из j -й модели движения потребителя в i -ю:

$$p_{k-1}^{ji} = p_{ji} p_{k-1}^{(j)} / p_{k|k-1}^{(i)}.$$

3. Определяются взвешенные значения апостериорных условных оценок $\bar{\lambda}_{k-1|k-1}^{(i)}$:

$$\bar{\lambda}_{k-1|k-1}^{(i)} = \sum_{j=1}^M \hat{\lambda}_{k-1|k-1}^{(j)} p_{k-1}^{ji},$$

а также условные взвешенные значения матрицы ошибок фильтрации

$$\bar{D}_{\lambda, k-1|k-1}^{(i)} = \sum_j [D_{\lambda, k-1|k-1}^{(j)} + (\bar{\lambda}_{k-1|k-1}^{(i)} - \bar{\lambda}_{k-1|k-1}^{(j)}) (\bar{\lambda}_{k-1|k-1}^{(i)} - \hat{\lambda}_{k-1|k-1}^{(j)})^T] p_{k-1}^{ji}.$$

4. Определяются экстраполированные условные оценки и матрицы ошибок фильтрации:

$$\hat{\mathbf{F}}_{k|k-1}^{(i)} = \mathbf{F}_{k-1}^{(i)} \bar{\lambda}_{k-1|k-1}^{(i)} + \mathbf{G}_{k-1}^{(i)} \bar{\boldsymbol{\mu}}_{k-1}^{(i)}; \quad \mathbf{D}_{\lambda, k|k-1}^{(i)} = \mathbf{F}_{k-1}^{(i)} \bar{D}_{\lambda, k-1|k-1}^{(i)} (\mathbf{F}_{k-1}^{(i)})^T + \mathbf{G}_{k-1}^{(i)} \mathbf{D}_{\boldsymbol{\mu}, k-1}^{(i)} (\mathbf{G}_{k-1}^{(i)})^T,$$

где $\bar{\boldsymbol{\mu}}_{k-1}^{(i)} = E[\boldsymbol{\mu}_{k-1}^{(i)}]$; $\mathbf{F}_{k-1}^{(i)}$, $\mathbf{D}_{\boldsymbol{\mu}, k-1}^{(i)}$ – условные динамические матрицы пересчета и корреляционные матрицы случайного маневра.

5. По результатам новых наблюдений y_k определяется невязка измерений

$$\tilde{y}_k^{(i)} = y_k - S_k(\hat{\mathbf{x}}_{k|k-1}^{(i)}) - \bar{\mathbf{n}}_k^{(i)},$$

где $\bar{\mathbf{n}}_k^{(i)} = E[\mathbf{n}_k^{(i)}]$.

6. Определяются условные результирующие оценки и ошибки фильтрации вектора состояния:

$$\mathbf{D}_{\lambda,k|k}^{(i)} = \left[(\mathbf{D}_{\lambda,k|k-1}^{(i)})^{-1} + \left(\frac{\partial \mathbf{S}_k(\mathbf{c}^{(i)} \hat{\mathbf{x}}_{k|k-1}^{(i)})}{\partial \boldsymbol{\lambda}^{(i)}} \right)^T (\mathbf{D}_n^{(i)})^{-1} \left(\frac{\partial \mathbf{S}_k(\mathbf{c}^{(i)} \hat{\mathbf{x}}_{k|k-1}^{(i)})}{\partial \boldsymbol{\lambda}^{(i)}} \right) \right]^{-1};$$

$$\hat{\mathbf{x}}_{k|k}^{(i)} = \hat{\mathbf{x}}_{k|k-1}^{(i)} + \mathbf{D}_{\lambda,k|k}^{(i)} (\mathbf{c}^{(i)})^T \left(\frac{\partial \mathbf{S}_k(\mathbf{c}^{(i)} \hat{\mathbf{x}}_{k|k-1}^{(i)})}{\partial \mathbf{H}^{(i)}} \right)^T (\mathbf{D}_n^{(i)})^{-1} \tilde{y}_k^{(i)}.$$

7. Определяются коэффициенты правдоподобия для каждой гипотезы и апостериорные вероятности гипотез:

$$L_k^{(i)} = \frac{\exp \left[-(1/2) (\tilde{y}_k^{(i)})^T \left[\left(\frac{\partial \mathbf{S}_k(\mathbf{c}^{(i)} \hat{\lambda}_{k|k-1}^{(i)})}{\partial \boldsymbol{\lambda}^{(i)}} \right) \mathbf{D}_{\lambda,k|k-1}^{(i)} \left(\frac{\partial \mathbf{S}_k(\mathbf{c}^{(i)} \hat{\lambda}_{k|k-1}^{(i)})}{\partial \boldsymbol{\lambda}^{(i)}} \right)^T + \mathbf{D}_n^{(i)} \right]^{-1} \tilde{y}_k^{(i)} \right]}{\left| 2\pi \left[\left(\frac{\partial \mathbf{S}_k(\mathbf{c}^{(i)} \hat{\lambda}_{k|k-1}^{(i)})}{\partial \boldsymbol{\lambda}^{(i)}} \right) \mathbf{D}_{\lambda,k|k-1}^{(i)} \left(\frac{\partial \mathbf{S}_k(\mathbf{c}^{(i)} \hat{\lambda}_{k|k-1}^{(i)})}{\partial \boldsymbol{\lambda}^{(i)}} \right)^T + \mathbf{D}_n^{(i)} \right] \right|^{1/2}},$$

$$P_k^{(i)} = \frac{P_{k|k-1}^{(i)} L_k^{(i)}}{\sum_j P_{k|k-1}^{(j)} L_k^{(j)}}.$$

8. Вычисляются безусловные оценки вектора состояния и значения матрицы ошибок фильтрации:

$$\hat{\mathbf{x}}_{k|k} = \sum_{i=1}^M \hat{\mathbf{x}}_{k|k}^{(i)} P_k^{(i)}; \quad \mathbf{D}_{\lambda,k|k} = \sum_{i=1}^M [\mathbf{D}_{\lambda,k|k}^{(i)} + (\hat{\mathbf{x}}_{k|k} - \hat{\mathbf{x}}_{k|k}^{(i)})(\hat{\mathbf{x}}_{k|k} - \hat{\mathbf{x}}_{k|k}^{(i)})^T] P_k^{(i)}.$$

Результаты моделирования

Моделировалась следующая траектория движения потребителя: с 1-го по 1000-й шаг потребитель неподвижен, с 1000-го по 2000-й шаг скорость потребителя увеличивается на каждом шаге на 0,3 м/с, на 2000-м шаге ускорение скачком достигает значения 30 м/с² и далее до 3000-го шага осуществляется полет по модели Зингера [1].

Для оценки точности использовались сферические ошибки (СФО) координат и скорости $e_{сф}$. На рис. 1,2 приведены результаты моделирования двух алгоритмов: неадаптивного и адаптивного МГИ.

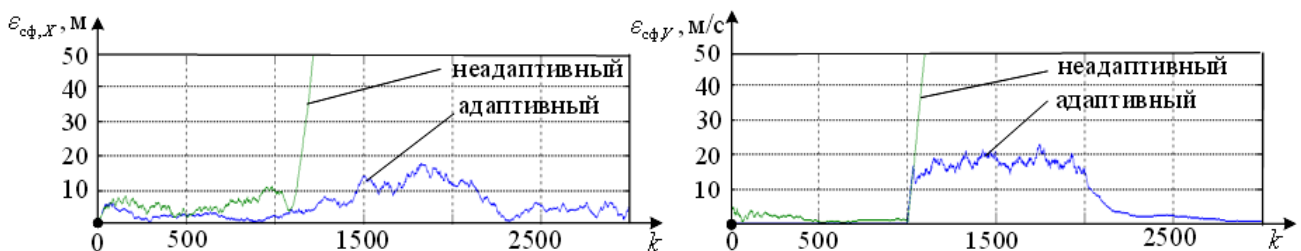


Рисунок 1 – СФО оценки координат и скорости потребителя

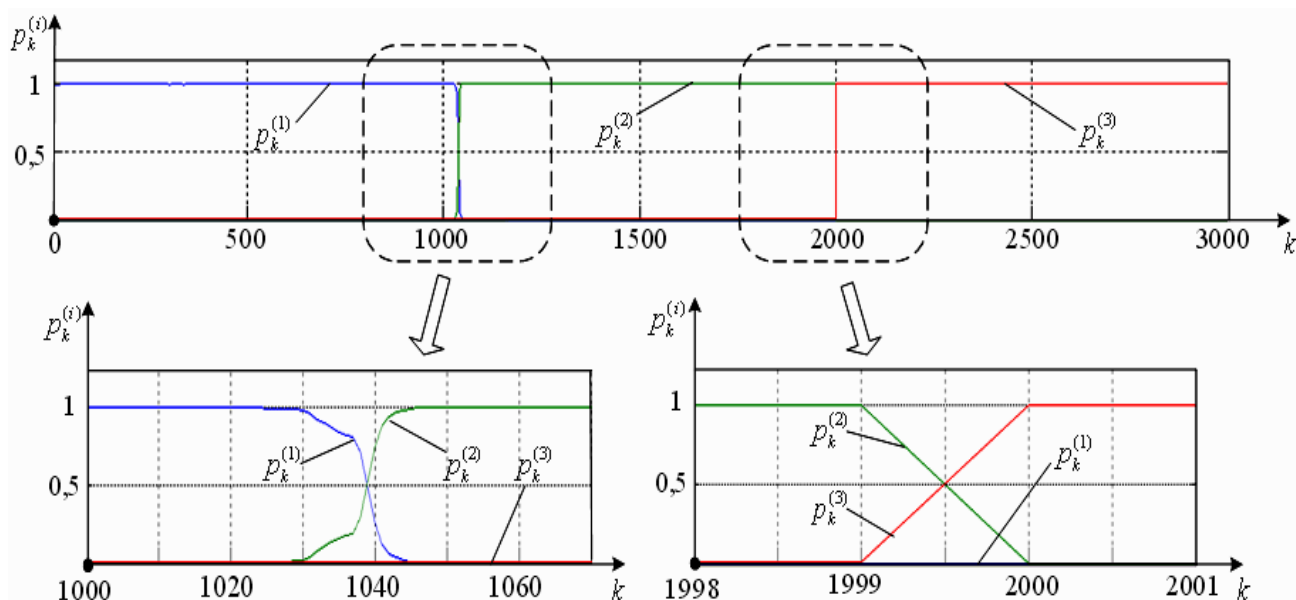


Рисунок 2 – Изменение апостериорных вероятностей гипотез

Заключение

В процессе синтеза неадаптивного измерителя априорно задавалась модель движения потребителя по Зингеру, которая закладывалась в его структуру. Для синтеза адаптивного измерителя принимались три модели возможного движения потребителя, при этом один из каналов являлся идентичным каналу неадаптивного измерителя. При соответствии априорных данных, заложенных в структуру неадаптивного измерителя, модели задающего воздействия неадаптивный измеритель давал выигрыш по СФО по сравнению с адаптивным МГИ не более чем на 10 % при оценивании координат потребителя. Это объясняется влиянием ошибок, вносимых параллельными каналами адаптивного измерителя, вес которых достаточно близок, но не равен нулю. При оценивании других составляющих вектора состояния результаты работы измерителей достаточно близки.

В процессе маневрирования структура неадаптивного измерителя становилась неоптимальной и возникали ошибки навигационно-временных определений. Большое структурное или параметрическое различие априорной информации и задающего воздействия неизбежно вызывало ошибки, которые приводили к срыву сопровождения (через 0,5...1 с после начала маневра). В рассматриваемой тактической ситуации адаптивный МГИ успешно адаптировался к маневру (время адаптации составило 0,01...0,45 с), обеспечивая устойчивое оценивание составляющих вектора состояния.

Литература

1. Singer, A.R. Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets / A.R. Singer // IEEE Trans. on Aerospace and Electronic Systems. – July, 1970, Vol. AES-6, No. 4. – P. 473-483.
2. ГЛОНАСС. Принципы построения и функционирования / Под ред. А.И. Перова, В.Н. Харисова. – Изд. 3-е, перераб. – М.: Радиотехника, 2005. – 688 с.: ил.
3. Li X.R., Jilkov V.P. A Survey of Maneuvering Target Tracking – Part V: Multiple-Model Methods / X.R. Li, V.P. Jilkov // IEEE Trans. Aerospace and Electronic Systems. – Nov 26, 2003. – 58 p.

УДК 32.973-018.2

ИСПОЛЬЗОВАНИЕ СИСТЕМЫ РАСПРЕДЕЛЕНИЯ ПАМЯТИ НА ПРИМЕРЕ ВЫДЕЛЕНИЯ ПАМЯТИ ПОД МАССИВ ПСЕВДОСЛУЧАЙНЫХ ЧИСЕЛ СРЕДСТВАМИ ООП

Пашкевич А.А.

УО «Барановичский государственный университет», г. Барановичи

ВВЕДЕНИЕ. Алгоритмы отведения и распределения памяти – это особый интересный раздел теории и практики программирования [1]. На сегодняшний день мы имеем в своем распоряжении очень мощные компьютеры, которые за долю секунды обрабатывают огромное количество информации, начиная от простейших математических расчетов до симуляции сложных явлений [2]. Но какой бы мы производительный компьютер не имели, все утратит смысл без должной оптимизации программного обеспечения. Одним из способов увеличения быстродействия программы является внедрение в нее системы распределения динамической памяти.

Такая система позволит программам с использованием статических элементов либо динамических структур увеличить их эффективность и скорость обработки данных, при этом уменьшить количество выделяемой памяти и количество вызовов сложных системных механизмов распределения памяти [3].

Автором создана программа, которая наглядно демонстрирует различные алгоритмы распределения памяти, сравнивая их между собой по быстродействию и затраченной памяти на примере выделения памяти под массив псевдослучайных чисел. Для этого была разработана система распределения памяти блоками постоянной длины.

Объектом и предметом исследования является быстрое и эффективное выделение памяти под динамические структуры и ее освобождение, а также сравнение данной системы со статическим методом и стандартными функциями языка C++.

ОСНОВНАЯ ЧАСТЬ. Автором создана программа для наглядной работы алгоритмов распределения памяти:

- Создание входных данных происходит автоматически;
- Организация представления данных в таблице;
- Промежуточный вывод результатов для каждого из алгоритмов;
- Изменение размерности буфера и отображение его наполненности в системе распределения памяти;
- Вывод информации о системе, на которой запущено приложение;
- Графическое представление результатов в виде диаграмм.

Пользователю предлагается определить на начальном этапе входные данные. Для этого через меню Данные → Создать либо сочетанием клавиш Ctrl+N вызывается дополнительное окно «Создание данных», в котором можно задать тип данных (целочисленный или вещественный), размерность массива, диапазон данных и делитель для вещественного типа данных. После внесения необходимых параметров и нажатия кнопки «Создать данные» получаем готовые исходные данные (рисунок 1) и загружаем их на главное окно нажатием клавиши «Загрузить данные».



Рисунок 1 – Создание исходных данных

После создания исходных данных загружаем их в оперативную память любым из трех алгоритмов распределения памяти, получая промежуточные данные. Увеличивая размерность буфера системы распределения памяти, повышаем скорость выделения памяти под исходные данные (рисунок 2).

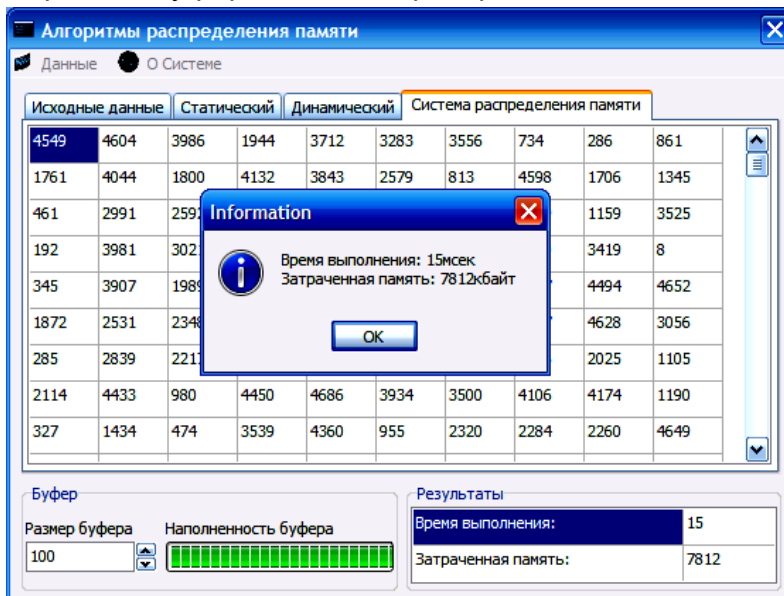
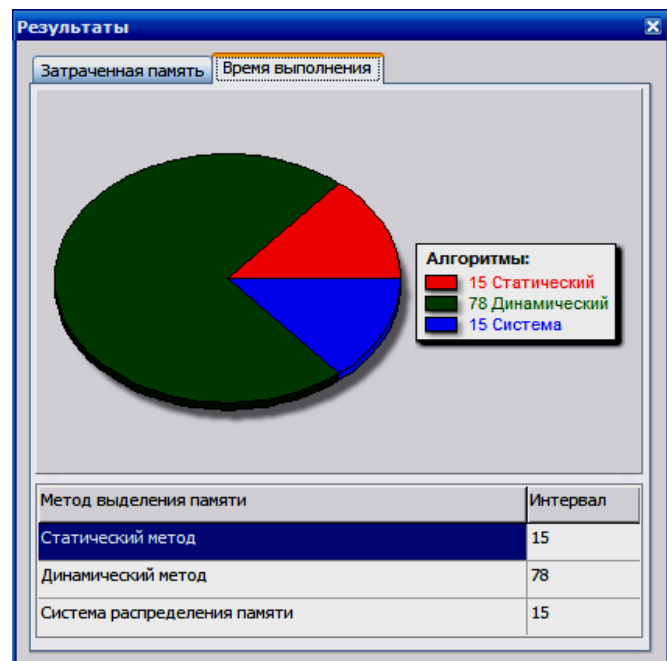


Рисунок 2 – Использование системы распределения памяти

Для наглядного представления выходных характеристик используются диаграммы. Программа демонстрирует пользователю время выполнения алгоритма и затраченную память на создание данных из исходных (рисунок 3).

Рисунок 3 – Результаты использования алгоритмов распределения памяти

Также для полной характеристики алгоритмов можно использовать информацию (касательно алгоритмов распределения памяти) о системе, на которых производилось исследование. Для этого достаточно открыть окно «О системе» из меню О системе (рисунок 4).



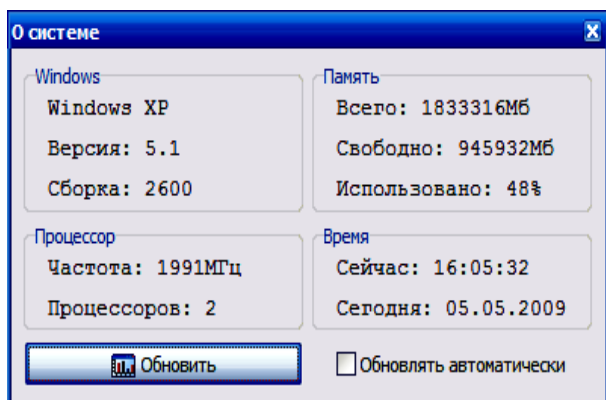


Рисунок 4 – Информация о системе

Данные исследования показали, что эффективно использовать систему распределения памяти возможно при размерах буфера, кратных размерности исходных данных, не малых, но и не слишком больших, и тем более не превышающих размерность исходных данных.

ЗАКЛЮЧЕНИЕ. В ходе исследования была разработана система распределения памяти блоками постоянной длины на примере выделения памяти под массив псевдослучайных чисел. Программа наглядно показывает работу алгоритмов распределения памяти, сравнивая их по быстродействию и затраченной памяти. Среда разработки CodeGear C++Builder 2009.

Системные требования:

- Дисковое пространство - 1.39Мб;
- Занимаемая оперативная память – от 5Кб (варьируется от размера массива и использованного для этого алгоритма);
- Операционная система –Windows 2000, XP, VISTA или 7 (необходима поддержка Unicode).

Данная работа рекомендуется также в качестве методического обеспечения при изучении раздела «Указатели» дисциплины «Конструирование программ и языка программирования» для студентов технических специальностей инженерного факультета БарГУ, может использоваться в системах с большим количеством входных данных, быстрым выделением памяти под них, например, в имитации ядерных реакций по распаду атомов урана, в видеомонтаже и обработки 3D-фигур.

Основные достоинства:

- Создание массива большой размерности и автоматическое его заполнение псевдослучайными числами с выбором типа и диапазона;
- Изменение размерности буфера;
- Информация о системе;
- Графическое представление результатов исследования.

К недостаткам можно отнести:

- Добавление/Удаление элементов из массива не реализована;
- Буфер имеет постоянную длину.

Литература

1. Кубенский, А.А. Структуры и алгоритмы обработки данных: объектно-ориентированный подход и реализация на С++ / А.А. Кубенский – СПб.: БХВ-Петербург, 2004.
2. Вирт, Н. Алгоритмы и структуры данных / Н. Вирт; пер. с англ. – М.: Мир, 1989.
3. Романов, Е.Л. "Практикум по программированию на С++" / Е.Л. Романов – Санкт-Петербург, "БХВ-Петербург", 2004.

УДК 338.24

ГИБРИДНЫЙ ПОДХОД К МОДЕЛИРОВАНИЮ УПРАВЛЕНИЯ СОЦИАЛЬНО-ЭКОНОМИЧЕСКИМ РАЗВИТИЕМ РЕГИОНА

Печко Е.В.

УО «Белорусский государственный экономический университет», г. Минск

Решение задач тактического и стратегического управления является частью повседневной работы органов государственного управления любого уровня. Однако процесс принятия решений характеризуется сложностью реальных проблем, необходимостью согласовывать интересы различных экономических субъектов, неполнотой и неопределенностью доступной информации о состоянии объекта управления. Для обеспечения информационной поддержки принятия решения на уровне региона разработан прототип системы поддержки принятия решений [1], который базируется на интегрированном комплексе экономико-математических моделей.

Рассматриваемый подход к моделированию является гибридным и сочетает в себе идеи импульсного моделирования на функциональных графах [2], агентно-ориентированного подхода к моделированию [3], согласования экономических интересов путем формирования системы динамических нормативов [4] с последующим решением задач распространения ограничений [5] и построения сети напряженных вариантов [6].

В данной работе система моделирования управления социально-экономическим развитием региона рассматривается как мультиагентная. В ней каждый экономический субъект описывается как агент, преследующий в своей деятельности определенные цели и обладающий определенным уровнем интеллекта, достаточным для того, чтобы самостоятельно принимать решения. Взаимодействие агентов происходит путем обмена импульсами.

Математическая модель агента X имеет вид:

$$X = \langle G_x, S_x, R_x, ACT_x \rangle \quad (1)$$

Экономические агенты являются агентами с состоянием $S_x = \langle V_x, C_x \rangle$, которое изменяется в зависимости от восприятия окружающей среды и других агентов.

Здесь $V_x = \langle V_x^{in}, V_x^{ext} \rangle$ – множество параметров, характеризующих состояние агента;

V_x^{in} – множество внутренних параметров, недоступных другим агентам;

V_x^{ext} – множество параметров состояния, информация о которых доступна другим агентам;

C_x – множество целей агента.

Множество целей агента C_x представляется в виде множества пар $\langle c_{x,i}, w_{x,i} \rangle, i = \overline{1, k}$, являющихся частью его внутреннего состояния:

$c_{x,i}$ – i -ое ограничение, представляющее собой логическое выражение;

$w_{x,i} \in [0, 1]$ – приоритетность данной цели для агента (необходимая, желательная, возможная).

Формирование целей агентов происходит на основе построения динамических нормативов (ДН), характеризующих эталонный режим функционирования, с последующим анализом степени близости к эталону и формированием целевых установок корректирующего ДН. При этом целевые установки дополняются ограничениями, позволяющими минимизировать диспропорции в системе «цели – ресурсы – структура».

Изменение внутреннего состояния происходит в зависимости от восприятия окружающей среды. Для формализации этого процесса вводится функция реакции R_x :

$$R_x : P_{y \neq x} \times S_x \rightarrow S_x, \quad (2)$$

где $P_{y \neq x}$ - возмущение (импульс), воздействующее на агента в текущий момент времени;
 S_x - состояние агента.

Для оценки соответствия внутреннего состояния поставленным целям после его изменения вводится функция цели G_x :

$$G_x : C_x \times S_x \times V_{y \neq x}^o \rightarrow [0,1], \quad (3)$$

где C_x – множество целей агента;

S_x – текущее состояние агента;

$V_{y \neq x}^o$ – состояние других агентов в текущий момент времени.

Так, например, один из подходов к вычислению функции цели включает расчет для каждого ограничения/логического условия показателя совпадения $g_{x,i}, i = \overline{1, k}$:

$$g_{x,i} = \begin{cases} 1, & \text{если } c_{x,i} = true \\ 0, & \text{если } c_{x,i} = false \end{cases} \quad (4)$$

Тогда функция цели G_x может быть представлена в виде (5):

$$G_x = \frac{\sum_{i=1}^k w_{x,i} g_{x,i}}{k}, \quad (5)$$

где $w_{x,i} \in [0,1]$ – приоритетность i -ой цели для агента X ;

$g_{x,i}$ – показатель совпадения для цели i ;

k – число целей агента X .

Важной частью математической модели агента является прогнозирующая функция, которая реализует поведение, управляемое целями. В предлагаемом подходе данная функция реализована путем совместного решения задачи удовлетворения ограничений и задачи построения сети напряженных вариантов с целью выбора оптимальных по Парето решений из найденных.

Существует несколько подходов к решению задач удовлетворения ограничений [5], но только часть из них позволяет учитывать приоритетность ограничений. К таким алгоритмам относятся IHCS [7] и Indigo [8]. В СППР имеется возможность перед имитацией выбирать один из алгоритмов.

Действия агента представляются в виде импульсов, которыми он обменивается с другими агентами. Функция генерации импульсов ACT_x задается следующим образом:

$$ACT_x : S_x \times G_x \rightarrow P^{ext}, \quad (6)$$

где S_x – текущее состояние агента;

G_x – текущее значение функции цели;

P^{ext} - исходящее возмущение (импульс).

Расчеты, проводимые с помощью СППР для Брестской области, позволяют сделать вывод, что интеграция вышеперечисленных подходов в единую систему позволяет минимизировать недостатки, присущие каждому подходу в отдельности. Так, например, импульсное моделирование позволяет получать в основном качественные результаты и не учитывает интеллектуального поведения экономических агентов, которое не всегда может быть выражено только функциями реакции или генерации импульсов. В свою очередь, в мультиагентных системах агенты строят свое поведение с учетом тактических и стратегических целей, но действия планируются в основном на основе накопленного опыта. Решение задачи распространения ограничений позволяет уточнить возможные значения параметров состояния в зависимости от состояния окружающей среды и других агентов, но не дает ответа на вопрос, какой из вариантов следует предпочесть. В то же время набор допустимых альтернатив может быть получен путем построения сети напряженных вариантов с последующим отбором вариантов, удовлетворяющих выбранному критерию, например, минимизации издержек.

Таким образом, предлагаемый гибридный подход к построению подсистемы моделирования в региональной СППР позволяет учитывать интересы всех экономических агентов, моделировать их интеллектуальное поведение в соответствии с поставленными целями и выбранными приоритетами развития, что дает возможность своевременного принятия управленческих решений и анализа их последствий.

Литература

1. Печко, Е.В. Разработка региональных систем поддержки принятия решений / Е.В. Печко // Современные проблемы математики и вычислительной техники: материалы V республиканской научной конференции молодых ученых и студентов, Брест, 28-30 ноября 2007. – Брест: БрГТУ, 2007. – С. 113-115.
2. Ковалевский, С.С. Создание систем мониторинга реализации федеральных целевых программ/ С.С. Ковалевский, В.В. Кульба. – М.: Синтег, 2006. – С. 31-77.
3. Бугайченко, Д.Ю. Абстрактная архитектура интеллектуального агента и методы ее реализации / Д.Ю. Бугайченко, И.П. Соловьев // Системное программирование. Вып. 1: Сб. статей; под ред. А.Н.Терехова, Д.Ю.Булычева. – СПб.: Изд-во СПбГУ, 2005. – С. 36-67.
4. Фридман, Ю.А. Оценка уровня согласованности экономических интересов субъектов региональной промышленной политики/ Ю.А. Фридман, Г.Н. Речко, О.А. Бияков, Ю.Ш. Блам // Регион: экономика и социология. – 2008. – № 3. С. 78-96.
5. Семенов, А.Л. Методы распространения ограничений: основные концепции / А.Л. Семенов // Международное совещание по интервальной математике и методам распространения ограничений: труды совещания, Новосибирск, 8–9 июля, 2003 г. – Новосибирск, 2003.– С. 6–20.
6. Андронникова, Н.Г. Комплексное оценивание в задачах регионального управления / Н.Г. Андронникова, В.Н. Бурков, С.В. Леонтьев. – М.: ИПУ РАН, 2002. – 58 с.
7. Menezes, F. Incremental Hierarchical Constraint Solver (IHCS) / F. Menezes, P. Barahona, P. Codognet // An Incremental Hierarchical Constraint Solver, in: Proceedings of PPCP93, – Newport, 1993. – P. 190-199.
8. Borning, A. The Indigo Algorithm / A. Borning, R. Anderson, B. Freeman-Benson// TR 96-05-01, Department of Computer Science and Engineering, University of Washington. – July 1996.

УДК 004.9(075)

ТЕСТИРОВАНИЕ ДАННЫХ БИРЖЕВЫХ СВОДОК ИТ КОМПАНИЙ

Сазончик А.А

УО «Брестский государственный технический университет», г. Брест

В настоящее время для регулирования и прогнозирования мировой экономики применяются различные инструменты, и занимаются этим различные государственные, коммерческие и некоммерческие организации, а также органы власти. Особенным показателем состояния финансового рынка являются фондовые биржи, в частности их основные составляющие ценные бумаги. Данными ценными бумагами выступают акции. По динамике изменений цен акций можно отследить влияние некоторых факторов на капитализацию компании. Будь-то изменение цен на ресурсы, сырьё или некоторые преобразования в самой компании и много других. Задача состоит в том, что, возможно, кто-то влияет на колебания уровня цены акции на фондовой бирже. Так как разнообразие и количество компаний, торгующих своими акциями на биржах, в мире очень велико, то фондовые биржи формируют индексы, которые включают в себя некоторую группу компаний для прослеживания динамики состояния определенной области экономики, занимаемой компаниями, входящих в индекс.

Весьма актуальным является получение экономических выводов на базе математической обработки статистических показателей. При этом возникает задача проверки пригодности этих данных для анализа.

Для обработки данных по основным факторам и показателям использовалась множественная полиномиальная регрессия и коэффициенты (как выходные данные анализа): коэффициент детерминации и коэффициент Дарбина-Уотсона. Все вычисления проводились в системе Mathcad. Рассматривался период от 7 лет у компании Siemens до 47 лет у компании IBM. Цена акций этих компаний сравнивалась с такими факторами как:

- биржевые индексы NYSE и NASDAQ в периодах от 3 до 12 лет;

NASDAQ-100 (SYMBOL IXNDX)

The NASDAQ-100 Equal Weighted (SYMBOL NDXE)

NASDAQ Global Select Market (SYMBOL NQGS)

NYSE COMPOSITE INDEX (SYMBOL NYA)

NYSE ENERGY INDEX (SYMBOL NYE.ID)

NYSE INTERNATIONAL 100 INDEX (SYMBOL NYI.ID)

- цена акций с учётом инфляции;

- общий объём акций;

- мировые цены на золото (в долларах США за унцию);

- цены на нефть с учётом инфляции (в долларах США за баррель).

При обработке данных были получены следующие показатели коэффициентов:

Таблица 1

	R ²	DW
Apple	1	2.703
Siemens	1	1.5
Intel	0.999	1.439
Nokia	0.964	2.522
Microsoft	0.487	2.625
IBM	0.587	2.551

Где R^2 – коэффициент детерминации, который показывает, какая доля дисперсии резуль- тативного признака объясняется влиянием независимых переменных.

DW – коэффициент Дарбина-Уотсона

По полученным данным можно отметить, что у компании Apple сила связи факторов с ценой акции очень сильна, но в то же время коэффициент Дарбина-Уотсона не удалось сделать приемлемым, что позволяет предположить невысокое качество данных. Компа- ния Siemens также имеет высокую силу связи, однако качество данных плохое. Практи- чески такая же ситуация и у компании Intel. Это можно объяснить влиянием на коэффи- циент Дарбина-Уотсона индексов NYSE ENERGY INDEX и, возможно, цен на нефть.

Очень благоприятной ситуацией обладает компания Nokia. Качество данных этой компании приемлемое, и коэффициент детерминации показывает отличную силу связи. В меньшей степени это относится к компаниям Microsoft и IBM. У данных компаний, не- смотря на удовлетворительное качество данных, сила связи низкая.

В общей сложности, получены удовлетворительные показатели по анализируемым данным, что свидетельствует о неплохом качестве этих данных и (у большинства ком- паний) сильной связи с ценой акций. В итоге хочется отметить, что полученные резуль- таты могут быть использованы для развития критериев, предъявляемых к исходным данным статистической модели и её компьютерной реализации. Также необходимо дальнейшее развитие экономической модели.

Реализация в математической системе

N := 25

i := 0.. N - 1

apple := READ ("F:\Science\Apple\basa_price_share_1984_2008.txt")

index_nasdaq_gsm_i := READ ("F:\Science\Apple\basa_index_nasdaq_GSM_1998_2008.txt")

index_nasdaq_10Q := READ ("F:\Science \Apple\basa_index_nasdaq_100_1998_2008.txt")

index_nasdaq_EW_i := READ ("F:\ScienceApple\basa_index_nasdaq_100_EW_2005_2008.txt")

index_nyse_int_i := READ ("F:\Science\ Apple\basa_index_nyse_int_100.txt")

index_nyse_comp_i := READ ("F:\Science\ Apple\basa_index_nyse_composite_1984_2008.txt")

index_nyse_energy_i := READ ("F:\Science\Apple\basa_index_nyse_energy_2003_2008.txt")

volume_i := READ ("F:\Science\Apple\basa_volume_share_1984_2008.txt")

gold_i := READ ("F:\Science\Apple\basa_price_gold_1984_2008.txt")

oil_i := READ ("F:\Science\Apple\basa_oil_price_1984_2008.txt")

inflation_i := READ ("F:\Science Apple\basa_price_share_inflation_1984_2008.txt")

date_i := N - i

N = 25

data <0>	:= apple	Цены акций компании Apple
data <1>	:= date	количество значений
data <2>	:= index_nasdaq_gsm	индекс NASDAQ
data <3>	:= volume	объём котируемых акций
data <4>	:= gold	цены на золото
data <5>	:= index_nasdaq_100	индекс NASDAQ-100
data <6>	:= index_nasdaq_EW	индекс NASDAQ-100 Equal Weighted
data <7>	:= index_nyse_int	индекс NYSE INTERNATIONAL 100
data <8>	:= index_nyse_comp	индекс NYSE Composite
data <9>	:= index_nyse_energy	индекс NYSE ENERGY
data <10>	:= oil	цены на нефть
data <11>	:= inflation	доллар США с учетом инфляции

Зависимая переменная Y находится в векторе $data^0$, а независимая переменная $data^1 - data^{11}$

$N := \text{rows}(\text{data})$

$n := \text{cols}(\text{data})$

Степень полинома:

$K:=1$

$Y := \text{data} \langle 0 \rangle$

$X := \text{submatrix}(\text{data}, 0, N - 1, 1, n - 1)$

Число точек данных:

$N = 25$

Число переменных:

$n = 12$

$z := \text{regress}(X, Y, k)$

$i := 0.. N - 1$

Полином, соответствующий функции:

$\text{fit}(x) := \text{interp}(z, X, Y, x)$

$\text{pred}Y_i := \text{fit} \left[\begin{matrix} (X^T) \langle i \rangle \end{matrix} \right]$

Коэффициенты уравнения регрессии $y = a_0 + a_1x_1 + \dots + a_nx_n$

$\text{coeffs} := \text{submatrix}(z, 3, \text{length}(z) - 1, 0, 0)$

$\text{coeffs}^T =$		0	1	2	3	4	5
	0	-0.513	$-4.94 \cdot 10^{-5}$	$-1.667 \cdot 10^{-7}$	$-7.515 \cdot 10^{-3}$	$9.421 \cdot 10^{-4}$	$6.363 \cdot 10^{-3}$

Отклонение:

$\text{resid} := \text{pred}Y - Y$

$R^2:$

$$\frac{\sum (\text{pred}Y - \text{mean}(Y))^2}{\sum (Y - \text{mean}(Y))^2} = 1$$

$U := \text{pred}Y - Y$

$$DW := \frac{\sum_{i=1}^{N-1} (U_i - U_{i-1})^2}{\sum_{i=1}^{N-1} (U_i)^2}$$

$$DW = 2.703$$

Экспериментальные данные (Y):

Предсказанные на базе регрессии значения(predY):

	0		0
0	85.35	0	86.283
1	199.83	1	199.879
2	84.84	2	84.852
3	71.89	3	71.184
4	32.2	4	30.712
5	10.69	5	12.153
6	7.16	6	5.98
Y = 7	10.95	predY = 7	11.638
8	7.44	8	7.695
9	25.7	9	25.122
10	10.2	10	11.177
11	3.28	11	3.588
12	5.22	12	4.68
13	8	13	7.606
14	9.64	14	9.761
15	7.13	15	8.038

Литература

1. <http://inflationdata.com/Inflation/Inflation/Inflation.asp>
2. <http://www.nasdaq.com/asp/flashquotes.aspx?symbol=IXNDX&selected=IXNDX>
3. <http://www.nyse.com/about/listed/lcddata.html?ticker=NYA>

УДК 656.13.05

ИССЛЕДОВАНИЕ ВЛИЯНИЯ СДВИГА РАБОЧЕГО ГРАФИКА ПРЕДПРИЯТИЙ ГОРОДА НА ОСНОВНЫЕ ХАРАКТЕРИСТИКИ АВТОТРАНСПОРТНЫХ ПОТОКОВ

Сашко А.Н., Теленкевич Р.С.

УО «Брестский государственный технический университет», г. Брест

В современном мире проблема большой загруженности автотранспортных потоков, приводящая к пробкам, становится всё значительнее. Пробки приводят к огромным экономическим потерям, а также временным потерям. По прогнозам специалистов, в ближайшие годы движение в крупных городах может прекратиться из-за перегруженности городской автотранспортной сети. В связи с этим, поиск методов решения проблемы высокой загруженности автотранспортных потоков является актуальной проблемой.

Очевидно, что универсального решения данной проблемы не существует. Уменьшение нагрузки на городскую транспортную сеть является нетривиальной задачей и требует одновременного применения различных методов для её решения. В данной статье предлагается один из методов решения данной проблемы. Суть метода сдвига рабочего графика крупных городских предприятий заключается в поиске оптимального графика работы основных предприятий города, при котором нагрузки на транспортную сеть будут минимальны.

В рамках данного исследования используется абстрактная городская транспортная сеть, приближённая к реальной. Данная сеть имеет ряд особенностей, связанных с планировкой города (она не имеет ярко выраженных индустриальных зон и спальных районов) и населением (количество ежедневно присутствующего в городе приезжего населения незначительно в сравнении с популяцией города). Источниками потоков данной транспортной сети являются жилые зоны, узлами – дорожные регулируемые перекрёстки, а точками назначения потоков – предприятия города. Маршрутный граф данной сети является неориентированным, взвешенным и имеет следующие элементы:

1. Дорожные регулируемые перекрёстки N_1, N_2, \dots, N_n , где n – количество таких перекрёстков в сети. Каждый узел имеет четыре параметра $N_{i1}, N_{i2}, N_{i3}, N_{i4}$, $i=[1, n]$, каждый из которых содержит значение количества полос движения по одному из четырёх возможных направлений движения к перекрёстку.

2. Жилые районы S_1, S_2, \dots, S_m , где m – количество таких районов в сети, выступающие в качестве источников потоков.

3. Предприятия города E_1, E_2, \dots, E_l , где l – количество таких предприятий в сети, выступающие в качестве конечных точек потоков.

4. Дуги графа взвешенные. Веса дуг (w_1, w_2, \dots, w_k) имеют значение времени, необходимого для преодоления пути от узла к узлу, вычисленное статистически как среднее значение в интервале заданного времени.

В рамках исследования данного метода мы построили простой маршрутный граф (рисунок 1).

Входными параметрами модели служат значения весов дуг графа, количество рабочих мест на рассматриваемых в модели предприятия, среднее количество личных транспортных средств на тысячу населения. Распределение транспортных потоков по маршрутному графу основывается на маршрутной матрице, которая представляет собой набор наиболее вероятных маршрутов движения транспортных средств от жилых районов к городским предприятиям.

С точки зрения моделирования, регулируемый дорожный перекрёсток представляет собой сеть массового обслуживания. В качестве транзактов сети выступают транспортные средства. Транзакты поступают на многоканальное (роль каналов играют полосы движения) обслуживающее устройство (дорожный перекрёсток) группами через различные интервалы времени. Время обслуживания равно сумме времени на преодоление перекрёстка и времени, необходимого, чтобы начать движение, в случае, если до момента начала обслуживания машина находилась в очереди либо прибыла на перекрёсток в момент красной фазы светофора. Время обслуживания различное для различных транзактов и зависит от положения в очереди в момент включения зелёной фазы светофора. Обслуживание ведётся только во время активной зелёной фазы светофора. В момент активной красной фазы на обслуживающем перекрёстке нарастает очередь. Роль источника заявок играют жилые районы, роль приёмников – городские предприятия.

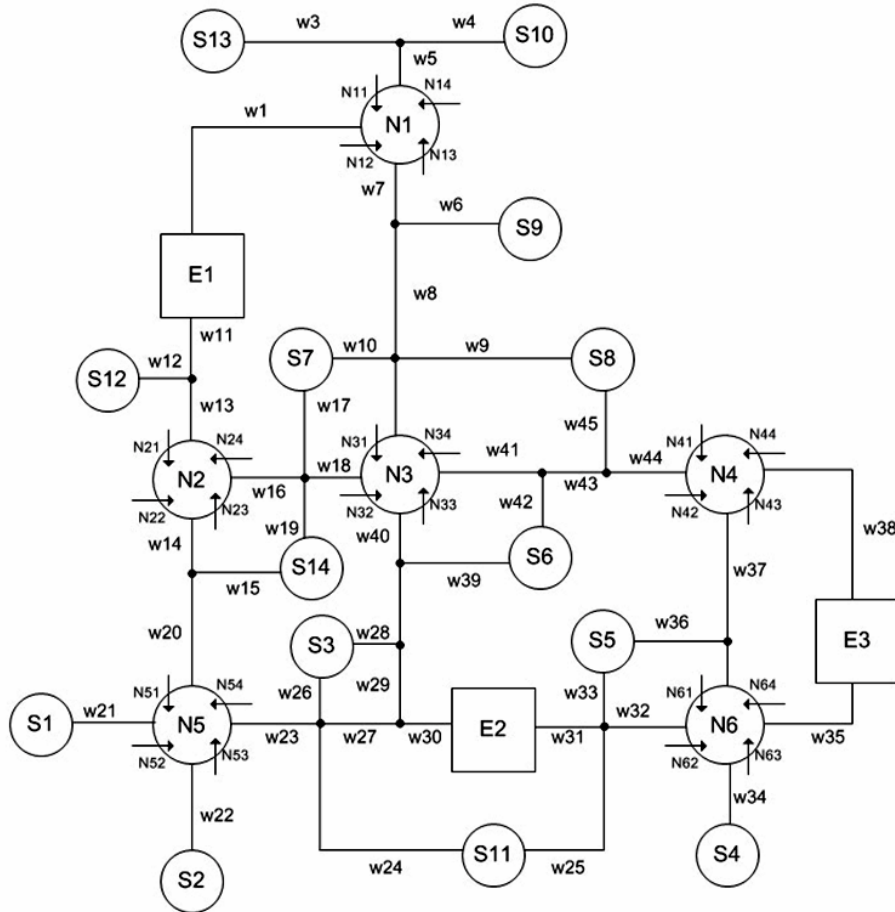


Рисунок 1 – Маршрутный граф

Целью эксперимента является определение наилучшего графика работы предприятий города, при котором средние показатели количества машин в очереди по каждому направлению перекрёстка и времени ожидания в очереди будут наименьшими, а также нагрузка на сеть будет распределена наиболее равномерно. Если обозначить среднее количество машин в очереди по каждому направлению перекрёстка N_{ij} (где i – индекс перекрёстка, а j – индекс направления) как V_{ij} и времени ожидания в очереди как T_{ij} , то условие оптимальности можно записать в виде формулы 1:

$$\left\{ \begin{array}{l} U = \sum_{i=1}^n \sum_{j=1}^4 \frac{V_{ij}}{4n} \rightarrow \min \\ U_S = \sum_{i=1}^n \left(\sum_{j=1}^4 \frac{V_{ij}}{4} - \sum_{i=1}^n \sum_{j=1}^4 \frac{V_{ij}}{4n} \right) \rightarrow 0 \\ H = \sum_{i=1}^n \sum_{j=1}^4 \frac{T_{ij}}{4n} \rightarrow \min \\ H_S = \sum_{i=1}^n \left(\sum_{j=1}^4 \frac{T_{ij}}{4} - \sum_{i=1}^n \sum_{j=1}^4 \frac{T_{ij}}{4n} \right) \rightarrow 0 \end{array} \right. \quad (1)$$

Значения U и H представляют собой среднее количество машин в очереди и время пребывания машины в очереди по всей сети соответственно. Значения U_S и H_S показывают общую среднюю разность времени пребывания в очереди и количества машин в очереди по всей сети и позволяют судить о равномерности распределения нагрузки на транспортную сеть.

Перед проведением эксперимента на модели необходимо определить шаг изменения времени графика работ предприятий. Известно, что максимальным возможным сдвигом в графике работы предприятия может быть один час. Это крайнее и наиболее нежелательное значение. В то же время сдвиг менее чем на полчаса может оказаться неэффективным. Таким образом, для нашей модели выберем шаг в полчаса.

Для заданной модели были проведены необходимые эксперименты, результаты которых доказывают эффективность предложенного метода. Результаты приведены в таблице 1. В таблице представлены данные экспериментов, при которых были получены наиболее оптимальные и неоптимальные данные.

Таблица 1 – Результаты экспериментов

№ эксперимента	Начальное время работы первого предприятия	Начальное время работы второго предприятия	Начальное время работы третьего предприятия	U	H	U _s	H _s
1	8:00	8:00	8:00	31,819	238,499	53,348	1672,151
4	8:00	7:30	8:00	0,740	32,698	1,130	232,852
6	8:00	7:30	8:30	0,391	32,360	0,599	231,671
14	7:30	7:30	7:30	43,659	238,473	73,369	1673,467
18	7:30	8:30	8:30	0,725	73,879	0,026	461,690
19	8:30	8:00	8:00	5,688	129,323	12,138	978,642
21	8:30	8:00	8:30	0,688	32,305	1,096	231,761
22	8:30	7:30	8:00	0,391	32,518	0,600	231,413
27	8:30	8:30	8:30	25,023	221,848	41,49	1525,160

Как видно из результатов моделирования, наиболее неудачными, с точки зрения нагрузки на транспортную сеть, являются те случаи, когда предприятия начинают работу в одно и то же время. Самыми эффективными, как с точки зрения минимизации средней нагрузки на сеть, так и равномерности распределения нагрузки по сети, являются обратные случаи. Как видно из полученных результатов, сдвиг в графике работы предприятий может существенно снизить нагрузку на транспортную сеть.

Данные, полученные на экспериментальной модели, подтвердили эффективность метода сдвига рабочих графиков предприятий города, на основе чего можно утверждать, что данный метод даст положительные результаты и на реальной модели. Метод не позволяет полностью разгрузить городскую транспортную сеть, но нагрузка на сеть снижается значительно.

Литература

1. Буслаев, А.П. Вероятностные и имитационные подходы к оптимизации автомобильного движения / А.П. Буслаев, А.В. Новиков, В.М. Приходько, А.Г. Таташев, М.В. Яшина – М.: Мир, 2003.
2. Брайловский, Н.О. Моделирование транспортных систем / Н.О. Брайловский, Б.И. Грайновский - М.: Транспорт, 1978.
3. Капитанов, В.Т. Управление транспортными потоками в городах / В.П. Капитанов, Е.Б. Хилажев – М.: Транспорт, 1985.

УДК 53.0871.088

МОДЕЛЬ КОРРЕЛЯЦИОННОГО ДЕТЕКТОРА НЕСТАЦИОНАРНЫХ СОСТОЯНИЙ ДИНАМИЧЕСКОЙ СИСТЕМЫ

Семёнов Георгий Александрович

Алтайский государственный университет, г.Барнаул, Россия

При изучении свойств различных открытых динамических систем принято пользоваться критериями и способами оценки хаотичности (показатели Ляпунова, энтропия, фрактальные размерности), основанными на исследовании всего временного ряда какой-либо физической величины (сигнала, порождаемого системой). При этом существуют задачи, для решения которых является необходимым разделение массива сигнала на некоторое количество подмассивов. Это связано с необходимостью изучения локальных изменений рассматриваемой физической величины.

Одной из таких задач является задача поиска режима перемежаемости динамической системы, который относят к надёжному способу детектирования детерминированного хаоса [1]. Режим перемежаемости – один из сценариев перехода динамической системы из упорядоченного состояния в хаотическое и обратно. В графическом отображении режим перемежаемости представляет собой чередование периодического и хаотического сигнала. Пример графического отображения режима перемежаемости представлен на рисунке 1.

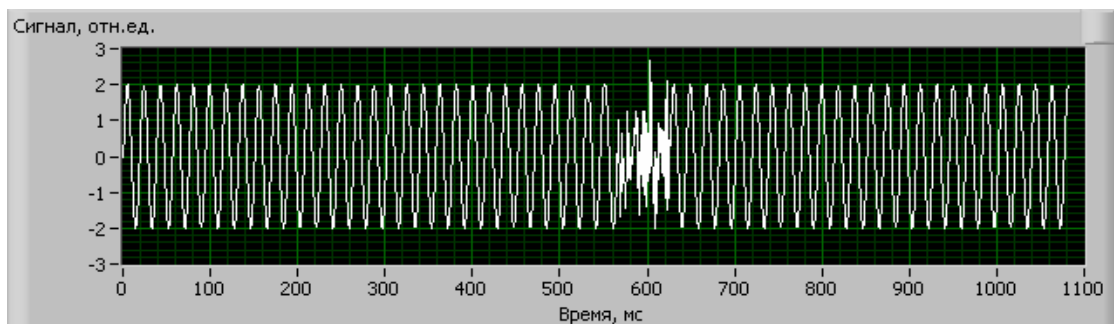


Рисунок 1 – Режим перемежаемости открытой динамической системы

Детектировать начало режима перемежаемости можно, решив задачу обнаружения ограниченного во времени хаотического фрагмента в упорядоченном сигнале. В случаях, когда хаотические фрагменты кратковременны, частота их появления мала, а сигнал является зашумлённым, визуальный поиск данных нестационарных участков по графическому отображению сигнала является весьма затруднительным.

Для решения этой проблемы необходимо создание детектора нестационарных состояний, позволяющего проводить автоматический поиск режима перемежаемости динамической системы.

Регистрация нестационарных состояний в рамках данной модели осуществляется по следующей схеме:

1. Формирование временного окна d – размера подмассивов сигнала, для которых будет вычислен контрольный параметр (d выбирается на основе гипотезы о длительности нестационарного фрагмента по отношению ко всему исследуемому сигналу);
2. Вычисление контрольного параметра для $(n-1)$ сдвигов временного окна с шагом $\Delta = d/k$ (значение величины k определяется требованиями точности);
3. Анализ полученного массива значений контрольных параметров для n подмассивов исследуемого сигнала;

4. Вывод о наличии нестационарной составляющей в рассмотренных участках исследуемого сигнала на основе результата проверки условия, определяемого критерием алгоритма поиска.

Контрольным параметром здесь является дисперсия автокорреляционных функций подмассивов исследуемого сигнала: в случае наличия хаотической составляющей дисперсия будет существенно ниже дисперсии массива сигнала, соответствующего периодическому сигналу. Поиск проводится следующим образом:

После подсчёта дисперсий D_i автокорреляционных функций подмассивов сигнала производится вычисление их наиболее вероятного значения D_{HB} , затем находится значение дисперсии $D(D_i)$. Заключительный шаг – проверка условия:

$$D_i < D_{HB} - 1,5\sqrt{D(D_i)}, \quad (1)$$

в случае положительного результата для некоторого D_k делается вывод о локализации в k -том подмассиве сигнала режима перемежаемости.

Данная модель корреляционного детектора скрытых нестационарных состояний динамической системы была создана в среде визуального программирования LabVIEW. Вычисление автокорреляционных функций производилось при помощи встроенного виртуального прибора “AutoCorrelation”, создание массива D_i и проверка условия – в окружениях циклов “For Loop” и “While Loop”. Также использовались встроенные функции “Mode”, “Standard Deviation and Variance”, “Greater?”, “Multiply”, “Divide”.

Для визуального наблюдения за изменением сигнала, корреляционных функций, массивом значений максимумов корреляционных функций использовался виртуальный прибор “Waveform Graph”.

Для апробации модели на основе встроенных виртуальных приборов “Gaussian White Noise” и “Sine Wave” при помощи функции “Insert Into Array” был создан генератор зашумлённого периодического сигнала с вкрапленными в него короткими гауссовскими шумовыми фрагментами.

При тестировании созданного детектора на данном генераторе был получен положительный результат, вследствие чего был сделан вывод о его применимости к реальным физическим лазерным системам.

Кроме того, корреляционный детектор был успешно применён к генератору решений системы Лоренца-Хакена [2], созданного на основе встроенного виртуального прибора “Lorenz Attractor” пакета LabVIEW.

$$\begin{cases} \frac{dE}{d\tau} = \kappa \cdot (P - E) \\ \frac{dP}{d\tau} = \gamma_{\perp} (nE - P) \\ \frac{dn}{d\tau} = \gamma_{\parallel} (A - n - EP) \end{cases}, \quad (2)$$

Здесь E – амплитуда электрического поля волны, P – амплитуда поляризации активной среды, n – разность населённостей рабочих уровней (инверсия), τ – время, γ_{\parallel} – скорость релаксации разностей населённостей, γ_{\perp} – скорость релаксации поляризации, A – параметр накачки (отношение ненасыщенной инверсии к её пороговому значению), $\kappa = \frac{1}{2T_c}$, T_c – время жизни фотона в резонаторе.

Модель, описываемая системой уравнений (2), соответствует одномодовому лазеру бегущей волны с двухуровневой активной средой, в которой реализуется однородное уширение спектральных линий излучения. Выбор математической модели динамической системы обусловлен тем, что она удовлетворительно описывает большое количество различных физических процессов, например, процесс образования ячеек Бенара [3]. Это позволило выявить класс задач, для которых описанный корреляционный детектор является применимым.

Таким образом, была создана и протестирована модель корреляционного детектора скрытых нестационарных состояний открытой динамической системы. На основе полученных результатов был сделан вывод о возможности детектирования в рамках данного метода детерминированного хаоса с помощью обнаружения режима перемежаемости.

Литература.

1. Климонтович, Ю.Л. Введение в физику открытых систем / Ю.Л. Климонтович – М.: Янус-К. 2002 – 284 с.
2. Ханин, Я.И. Основы динамики лазеров / Я. И. Ханин. – М.: Наука. Физматлит, 1999. – 368 с.
3. Хакен Г. Лазерная светодинамика / Г. Хакен – М.: Мир.– 1988. 350 с.

УДК 621.396.96

УПРОЩЕННЫЙ МЕТОД СТАТИСТИЧЕСКОГО АНАЛИЗА СИСТЕМ С ПЕРЕМЕННОЙ СТРУКТУРОЙ

Сидорович О.В.

УО «Военная академия Республики Беларусь», г. Минск

Введение. Для анализа системы стабилизации ракеты с регулятором переменной структуры (РПС), функционирующей в условиях случайных возмущений, можно воспользоваться двумя известными способами статистического анализа: методом статистической линеаризации (СЛ) [1] и методом, основанным на теории систем со случайной структурой (ТССС) [2].

Суть статистической линеаризации состоит в том, что нелинейное преобразование аппроксимируется линеаризованной зависимостью [1]. В результате для анализа нелинейного звена, которым является регулятор переменной структуры, можно применить аппарат линейной теории.

При анализе методом, основанным на ТССС [2], рассматривают систему с переменной структурой, действующую в условиях возмущений, как своеобразную модель марковского случайного процесса с поглощением и восстановлением реализаций на границе плоскости скольжения.

При этом для системы n -го порядка методом СЛ необходимо решить n уравнений для математических ожиданий и $n(n+1)/2$ для дисперсий, то есть всего $n^2 + 3n/2$ дифференциальных уравнений. Одна из проблем ТССС связана с тем, что закон изменения структуры предполагается пуассоновским, и это является главной сложностью применения ТССС для стохастического анализа ССР с РПС, так как в этом случае закон изменения структуры, очевидно, носит близкое к нормальному распределение. Учет произвольного закона изменения структуры динамической системы случайной структуры осуществляется с помощью метода псевдосостояний. «Однако одним из основных недостатков данного метода являются необходимость интегрирования существенно большого числа дифференциальных уравнений для вероятностных моментов и вероятностей

псевдосостояний структур (например, даже при однократном переключении структуры из первого состояния во второе по нормальному закону это увеличение составляет $k = 7...10$ раз)”[3]. Также необходимо отметить, что все дифференциальные уравнения для обоих методов необходимо решать совместно. В этом отношении метод СЛ выглядит значительно менее громоздким. Результаты математического моделирования показывают, что погрешности метода СЛ 10% – 15% [4]. Поэтому необходимо искать новые подходы в направлении обеспечения возможности эффективного анализа таких систем без увеличения числа интегрируемых уравнений для вероятностных моментов.

Основная часть. Предлагается анализ системы стабилизации ракеты с регулятором переменной структуры в канале тангажа в условиях действия случайных возмущений вести следующим образом: составить уравнения вероятностных моментов для каждой из структур отдельно, а затем интегрировать эти уравнения с учетом начальных условий на каждом шаге по знаку математического ожидания сигнала переключения.

Рассмотрим анализ системы стабилизации ракеты с РПС в условиях действия случайных возмущений, представленной на рисунке 1.

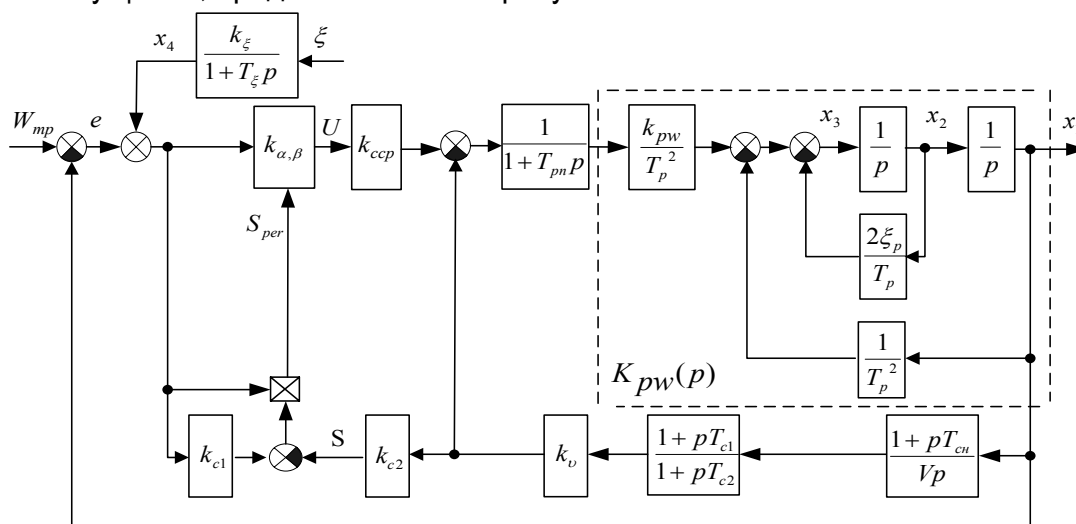


Рисунок 1 – Система стабилизации ракеты с РПС

На рисунке 1 введены обозначения:

W_{mp} – требуемое нормальное ускорение ракеты;

U – управляющее воздействие;

e – сигнал ошибки;

$K_{pw}(p) = \frac{k_{pw}}{1 + 2\sigma_p T_p p + T_p^2 p^2}$ – передаточная функция ракеты по нормальному ускорению (представлена в форме удобной для составления уравнений в нормальной форме);

T_p, ξ_p – постоянная времени и коэффициент демпфирования ракеты, соответственно;

$\frac{1}{1 + T_{pn} p}$ – передаточная функция рулевого привода;

T_{cn}, V – аэродинамическая постоянная времени снаряда и скорость полета ракеты,

соответственно;

S – сигнал плоскости скольжения;

x_1, x_2, x_3 – фазовые координаты системы.

Переключение структур осуществляется с помощью пси-ячейки, описываемой соотношениями

$$\mathbf{k}_{\alpha,\beta} = \begin{cases} \alpha & \text{при } (\mathbf{W}_{\text{тр}} - \mathbf{x}_1 + \mathbf{x}_4) \mathbf{S} \geq \mathbf{0}; \\ \beta & \text{при } (\mathbf{W}_{\text{тр}} - \mathbf{x}_1 + \mathbf{x}_4) \mathbf{S} < \mathbf{0}, \end{cases}$$

где

$$S = (W_{mp} - x_1 + x_4)k_{c1} - \frac{k_v k_{c2}}{V} x_2 - \frac{k_v k_{c2} T_{c1}}{V} x_3.$$

Случайное экспоненциально-коррелированное стационарное воздействие $x_4(t)$ со спектральной плотностью $S_x(\omega) = \alpha\sigma^2 / (\alpha^2 + \omega^2)$ является результатом преобразования белого шума ξ с единичной спектральной плотностью $N_\xi = 1$ формирующим фильтром $\frac{k_o}{I + T_o p}$.

По этим соотношениям можно получить уравнения вероятностных моментов [2]. На основании структурной схемы системы стабилизации ракеты, представленной на рисунке 1, запишем выражение для сигнала плоскости скольжения: $S = (W_{mp} - x_1 + x_4)k_{c1} - \frac{k_v k_{c2}}{V} x_2 - \frac{k_v k_{c2} T_{c1}}{V} x_3$.

Тогда сигнал переключения будет равен:

$$S_{per} = (W_{mp} - x_1 + x_4)S = (W_{mp} - x_1 + x_4) \left((W_{mp} - x_1 + x_4)k_{c1} - \frac{k_v k_{c2}}{V} x_2 - \frac{k_v k_{c2} T_{c1}}{V} x_3 \right)$$

Выражение для математического ожидания сигнала переключения структур

$$\begin{aligned} M_{per} = & k_{c1}W_{mp}^2 - k_{c1}W_{mp}M_1 + k_{c1}W_{mp}M_4 - \frac{k_v k_{c2}}{V}W_{mp}M_2 - \frac{k_v k_{c2} T_{c1}}{V}W_{mp}M_3 + \\ & + k_{c1}(M_1M_1 + D_{11}) - 2k_{c1}(M_1M_4 + D_{14}) + \frac{k_v k_{c2}}{V}(M_1M_2 + D_{12}) + \\ & + k_{c1}W_{mp}M_4 - k_{c1}(M_4M_4 + D_{44}) - \frac{k_v k_{c2}}{V}(M_2M_4 + D_{24}) - \frac{k_v k_{c2} T_{c1}}{V}(M_3M_4 + D_{34}) - \\ & - k_{c1}W_{mp}M_1 + \frac{k_v k_{c2} T_{c1}}{V}(M_1M_3 + D_{13}). \end{aligned}$$

где $M_i(j)$ и $D_{ik}(j)$ – математическое ожидание и дисперсия фазовых координат соответственно, находятся из решения уравнений вероятностных моментов.

На каждом j -м шаге, на основании значений $M_i(j)$ и $D_{ik}(j)$, определяется знак математического ожидания сигнала переключения M_{per} . Устанавливается значение коэффициента $k_{\alpha,\beta}$ в соответствии с неравенством

$$\mathbf{k}_{\alpha,\beta} = \begin{cases} \alpha & \text{при } M_{per} \geq 0; \\ \beta & \text{при } M_{per} < 0. \end{cases}$$

Затем осуществляется интегрирование уравнений вероятностных моментов до тех пор, пока математическое ожидание сигнала переключения не сменит свой знак. В этом случае в уравнениях моментов необходимо изменить значение коэффициента $k_{\alpha,\beta}$ и далее продолжать интегрирование. При этом на j -ом шаге начальными условиями для интегрирования уравнений моментов будут значения моментов на $j - 1$ шаге в момент смены знака, т.е. в момент переключения. Таким образом, последовательно осуществляется интегрирование уравнений моментов до необходимого момента времени

Заключение. В данной статье предложен метод статистического анализа систем стабилизации с регулятором переменной структуры работающего в скользящем режиме, суть которого заключается в интегрирование уравнений вероятностных моментов для каждой из структур отдельно, по математическому ожиданию сигнала переключения. Этот метод позволил:

- по сравнению с методом, основанном на теории систем со случайной структурой, исключить необходимость интегрирования существенно большего числа дифференциальных уравнений для вероятностных моментов и вероятностей псевдосостояний структур;
- по сравнению методом СЛ упростить методику составления уравнений моментов, т.к. коэффициенты статистической линеаризации являются громоздкими выражениями, содержащими нелинейную функцию Лапласа;
- получить необходимую точность, достаточную для анализа системы стабилизации ракеты с регулятором переменной структуры, что подтверждается результатами математического моделирования.

Литература

1. Жильцов, К.К.. Приближенные методы расчета систем с переменной структурой / К.К. Жильцов –М.: Энергия, 1974. – 224 с.: ил
2. Казаков, И.Е. Анализ систем случайной структуры / И.Е. Казаков, В.М. Артемьев, В.А. Бухалев –М.: Наука, 1993. –272 с.
3. Косачев, И.М. Аналитическое моделирование стохастических систем / И.М. Косачев, М.Г. Ерошенко –М.: Наука и техника, 1993. – 264 с.
4. Шабан, С.А. Особенности анализа системы стабилизации ракеты с регулятором переменной структуры методом статистической линеаризации / С.А. Шабан, О.В. Сидорович, И.Г. Ильев // Вестник Военной академии РБ №4(21) от 29.12.2008. – С.38-44.

УДК 681.3

ПОДХОД К ОРГАНИЗАЦИИ СИСТЕМЫ МОДЕЛИРОВАНИЯ

Согоян А.Л., Кочурко В.А.

УО «Брестский государственный технический университет», г. Брест

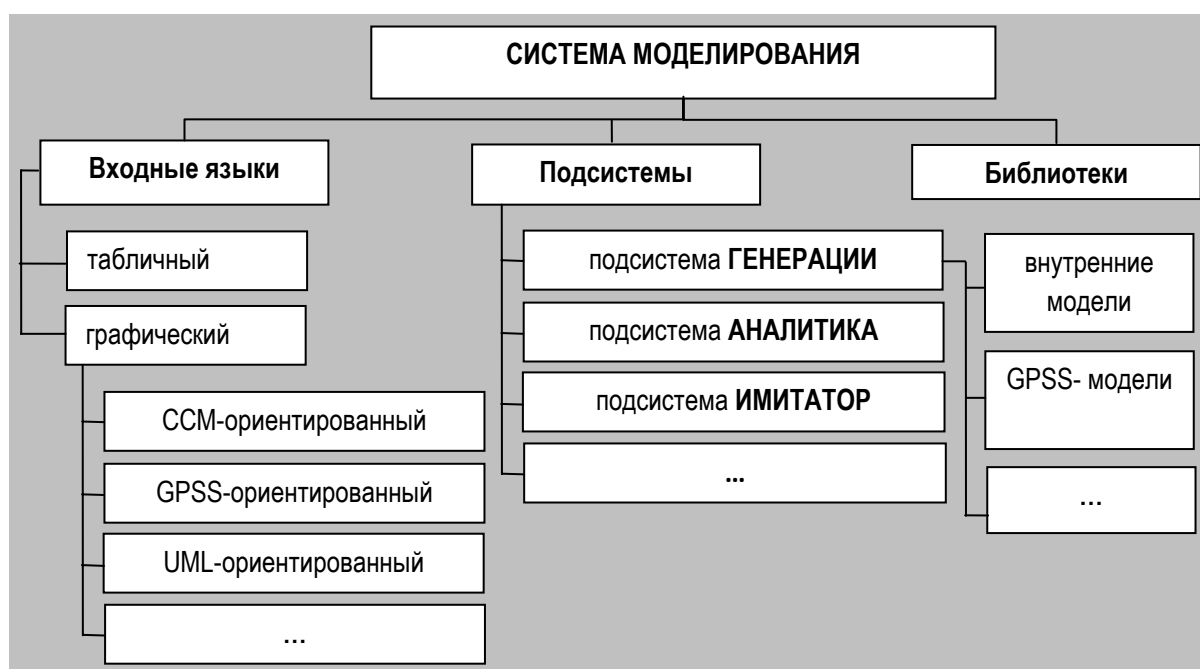
Существует много систем моделирования, ориентированных на реализацию конкретного метода, подхода к проведению моделирования либо ориентированных на предметную область [1-3].

Здесь решается задача выявления требований к архитектуре системы, обладающей следующими отличительными особенностями. Это, в том числе: ориентация на широко используемое в технических и иных приложениях подмножество Q-схем - стохастические сетевые модели (ССМ) и сети массового обслуживания (сети МО) [1, 2], описание которых обеспечивается входными языками системы; использование в качестве входных

языков системы графических ССМ-, GPSS-, UML-подобных языков, что обеспечивает визуальность системы и позволяет пользователю работать в привычных терминах предметной области, ССМ, а не конкретного языка моделирования; - акцент на роли системы в качестве надстройки, обеспечивающей использование как оригинальных средств моделирования так и инструментов существующих систем (например, GPSS [4]) через единый языковой интерфейс за счет обеспечения автоматической генерации результативных моделей; - акцент на многорежимном использовании системы, в т.ч. в режиме обучения, например, за счет предоставления обучаемому возможности обследовать и моделировать имитируемые системой объекты и др. Такая система моделирования (СМ) позволит создавать модели на интерфейсном уровне, не углубляясь в уровень реализации. Примерная структура системы приведена на рисунке ниже.

Система должна включать графическую оболочку и ядро, роль которого выполняет подсистема ГЕНЕРАЦИИ, служащая для автоматического порождения результативных моделей. Это внутренние модели СМ, настраиваемые на параметры объекта, или модели на языках существующих систем моделирования. В основу принципа работы подсистемы заложена идея распараллеливания потоков заявок моделируемого объекта: после трансформации входных данных в табличную модель с помощью механизма рекурсивного погружения поочередно для каждого генератора заявок из описания объекта исследуется вероятный маршрут их движения и обслуживания. Как результат, генерируется соответствующий текст модели на входном языке системы моделирования. Здесь для апробации системы в качестве такого языка взят входной язык системы GPSS World [4]. Кроме этого, СМ должна обеспечивать возможность обратной генерации графических описаний объекта на входных языках, например, по результатам анализа GPSS-текстов моделей объектов; проведение аналитического расчета упрощенных моделей.

Выходные отчеты могут содержать таблицы числовых данных, графики, гистограммы и т.п., а также исходные и сгенерированные тексты моделей, например, на языке GPSS. Отчеты хранятся в собственном формате данных СМ и экспортируются в общеизвестные форматы данных офисных пакетов для дальнейшей обработки и анализа.



Таким образом, в работе сформулированы подходы к организации системы моделирования, ориентированной на работу с Q-схемами и реализующей имитационные расчеты по табличным и графическим описаниям объектов с использованием как внутренних моделей, так и инструментов существующих систем моделирования. Рассмотрена примерная структура программного и лингвистического обеспечения, состав подсистем и их взаимодействие в ходе моделирования. Применительно к языку GPSS выполнено макетирование принципов работы программного ядра системы для случая использования описаний уровня сетей массового обслуживания. Дальнейшая работа предполагает макетирование подсистем, добавление оригинального модуля компиляции GPSS, разработку иерархии классов для поддержки внутренних моделей системы, расширение возможностей графического описания Q-схем и т.д.

Литература

1. Советов, Б.Я. Моделирование систем / Б.Я. Советов, С.А. Яковлев. - М.: Высшая школа, 2001. – 430 с.
2. Максимей, И.В. Имитационное моделирование на ЭВМ / И.В. Максимей. - М.: Радио и связь, 1988. – 270 с.
3. Кельтон, В. Имитационное моделирование. Классика CS / В. Кельтон, А. Лоу. – СПб.: Питер, 2004. – 630 с.
4. Рыжиков, Ю.И. Имитационное моделирование. Теория и технологии / Ю.И. Рыжиков. – СПб.: КОРОНА, 2004. – 320 с.

УДК 621.74

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ СВОЙСТВ СЫРОЙ ПЕСЧАНО-ГЛИНИСТОЙ ФОРМОВОЧНОЙ СМЕСИ

Филипенко Е.В.

УО «Гомельский государственный технический университет им. П. О. Сухого», г. Гомель

Литьё металла в одноразовую песчано-глинистую форму – один из самых популярных способов литья. Качество литейной продукции, изготавливаемой в сырых песчаных формах, в значительной степени находится под влиянием свойств формовочного материала, таких, как прочность на сжатие, уплотняемость, твердость и других.

Данная работа нацелена на определение отклика, а именно, формуемости, уплотняемости, прочности на сжатие и насыпной плотности, как функции различных независимых переменных, таких как влажность формовочной смеси, процентное содержание бентонита, процентное содержание крахмалита в смеси. При проведении исследования использовались такие статистические инструменты, как планирование эксперимента, регрессионный анализ и методология поверхности отклика.

Планирование эксперимента позволяет провести минимально необходимое число опытов для оценки отклика с заданной точностью. Планирование эксперимента в сочетании с методологией поверхности отклика – сильный статистический инструмент, позволяющий получить зависимости. Методология поверхности отклика позволяет создать приближенную модель используя полином, как аппроксимацию истинных зависимостей входа-выхода. Для разработки математической модели использовался центральный композиционный план. Он наиболее широко применяется при подборе моделей второго порядка.

В качестве зависимых переменных рассматривались формуемость, уплотняемость, прочность на сжатие, объемная плотность, а в качестве факторов – содержание бентонита, крахмалита и влажность. Бентонит увеличивает прочность по сырому, индекс формуемости, текучесть по Орлову, насыпной вес в гильзе, мгновенный модуль упругости, вязкопластические свойства. Вода является определяющим компонентом смеси, который в очень значительной мере, можно сказать, определяющим образом влияет на свойства смеси. Повышенное содержание влаги в формовочной смеси снижает прочностные характеристики смесей, увеличивает их газотворность, приводит к оолитизации зерен песчаной основы, вызывает пригар и повышение шероховатости отливок. Заниженное содержание влаги в формовочной смеси увеличивает ее осыпаемость и снижает прочностные свойства. Для стабилизации влажности форм, снижения хрупкости и осыпаемости, повышения прочности смеси в зоне конденсации влаги и уменьшения склонности к образованию ужимин в песчано-бентонитовые смеси вводят крахмалсодержащие добавки.

Были выбраны следующие обозначения: X_1 — содержание активного бентонита в смеси, фактор X_2 — влажность формовочной смеси, фактор X_3 – содержание крахмалита. Отклик эксперимента – Y_1, Y_2, Y_3, Y_4 , соответственно формуемость, уплотняемость, прочность и насыпная плотность смеси.

Далее был построен план эксперимента, и, в соответствии с построенным планом, собраны экспериментальные данные.

В процессе проведения исследования для приготовления смеси применялся лабораторный смеситель (модель LM-1), весы лабораторные квадратные (модель ВЛК-500т-м ГОСТ 24104-88), уплотнение образцов из песчано-глинистой смеси для испытаний проводится тремя ударами копра лабораторного (модель 5033А). Определение предела прочности образцов смеси при сжатии производилось на машине для испытания формовочных и стержневых смесей (модель LRu-1). Анализ влажности проводили с помощью прибора для определения влажности (модель 062M).

В качестве экспериментальных данных использовались результаты трех реплик построенного центрального композиционного плана. Для оценки результатов эксперимента применялись статистические методы, а также использовался программный продукт STATISTICA.

Формуемость формовочной смеси была выражена как нелинейная функция входных параметров следующим образом:

$$Y_1 = 202,045 - 2,145 X_1 - 63,416 X_2 - 1132,439 X_3 - 0,029 X_1^2 + 2,898 X_2^2 + 5960,668 X_3^2 + 1,650 X_1 X_2 + 16,281 X_1 X_3 + 204,793 X_2 X_3$$

Далее была проведена проверка значимости коэффициентов регрессии. Для этого выдвигается нулевая гипотеза о том, что коэффициенты статистически незначимо отличны от нуля. Для проверки гипотезы используется t-критерий. Далее эмпирическое значение t-критерия должно быть сопоставлено с табличным. Если $t_{эмп} > t_{крит}$, то нулевая гипотеза должна быть отвергнута. Следовательно, коэффициент значимо отличается от нуля и его следует сохранить в регрессионной модели. Если же $t_{эмп} \leq t_{крит}$, нулевая гипотеза принимается, соответствующий коэффициент регрессии полагается незначимым и исключается из регрессионной модели. После того, как были отброшены незначимые факторы из регрессионной модели, модель пересчитывается.

Так, последовательно осуществляя проверку значимости коэффициентов регрессии и производя соответствующую коррекцию, в конечном итоге получили модель, содержащую лишь значимые коэффициенты регрессии:

$$Y_1 = 124,720 - 30,515 X_2 + 0,915 X_1 X_2$$

Для полученного уравнения регрессии проводилась проверка его адекватности. Данная проверка осуществляется с помощью F-критерия Фишера, численное значение которого сравнивается с табличным критическим значением. Если $F_{эмт} \leq F_{крит}$, то нет оснований отклонять нулевую гипотезу. Если же $F_{эмт} > F_{крит}$, то гипотеза об отсутствии линейной связи отвергается. Так как $F_{эмт} = 99,593$, а $F_{крит} = 3,204$, значит, построенная регрессионная модель для формуемости является адекватной.

Возможность использования уравнения регрессии для предсказания значения отклика оценивалась при помощи анализа работоспособности модели. Модель может считаться работоспособной, если у этой модели коэффициент детерминации $R^2 \geq 0,75$.

Так как для построенной регрессионной модели $R^2 = 0,816$, то модель работоспособна.

Уплотняемость формовочной смеси была выражена как логарифмическая зависимость входных параметров:

$$Y_2 = 32,918 - 16,556 \ln X_1 + 41,301 \ln X_2 + 1,167 \ln X_3$$

Далее была проведена проверка значимости оценок коэффициентов в уравнении регрессии. Оказалось, что значимые эффекты имеют факторы $\ln X_1$ и $\ln X_2$. После отбрасывания незначимых факторов из регрессионной модели, коэффициенты были пересчитаны.

В итоге, получили следующую регрессионную модель:

$$Y_2 = 28,900 - 16,480 \ln X_1 + 41,001 \ln X_2$$

Из уравнения следует, что фактор $\ln X_2$ имеет больший вклад в значение параметра y_2 , чем фактор $\ln X_1$. Данная модель является адекватной, так как $F_{эмт} = 117,956$, а $F_{крит} = 3,220$. Так как $R^2 = 0,849$, модель считается работоспособной.

Прочность формовочной смеси была выражена как гиперболическая зависимость входных параметров:

$$Y_3 = -0,004 - 1,098 \frac{1}{X_1} + 1,222 \frac{1}{X_2} - 0,0002 \frac{1}{X_3} - 0,673 \frac{1}{X_1^2} - 1,984 \frac{1}{X_2^2} + 7,174 \frac{1}{X_3^2} + 2,579 \frac{1}{X_1 X_2} - 0,002 \frac{1}{X_1 X_3} - 0,002 \frac{1}{X_2 X_3}$$

Далее была проведена проверка значимости оценок коэффициентов в уравнении регрессии. Оказалось, что значимые эффекты имеют факторы $\frac{1}{X_1}$, $\frac{1}{X_2}$, $\frac{1}{X_2^2}$, $\frac{1}{X_1 X_2}$. В

итоге, получили следующую регрессионную модель:

$$Y_3 = 0,086 - 1,155 \frac{1}{X_1} + 0,651 \frac{1}{X_2} - 1,094 \frac{1}{X_2^2} + 1,803 \frac{1}{X_1 X_2}$$

Данная модель является адекватной, так как $F_{эмт} = 88,392$, а $F_{крит} = 3,606$. Так как $R^2 = 0,898$, модель считается работоспособной.

Насыпная плотность формовочной смеси была выражена как логарифмическая зависимость входных параметров:

$$Y_4 = 1,044 + 0,206 \ln X_1 - 0,588 \ln X_2 + 0,012 \ln X_3$$

Проверка значимости оценок коэффициентов в уравнении регрессии показала, что значимые эффекты имеют факторы $\ln X_1$, $\ln X_2$. После отбрасывания незначимых факторов из регрессионной модели и пересчета коэффициентов, получили следующую регрессионную модель:

$$Y_4 = 1,806 + 0,205 \ln X_1 - 0,585 \ln X_2$$

Данная модель является адекватной, так как $F_{эмп} = 139,679$, а $F_{крит} = 3,220$. Модель является работоспособной, $R^2 = 0,869$.

Полученные эмпирические зависимости показывают связь технологических свойств формовочной смеси с основными компонентами состава:

- влажность формовочной смеси оказывает в 2 раза большее влияние на ее формуемость, чем содержание активного бентонита;
- влажность формовочной смеси оказывает в 3 раза большее влияние на ее уплотняемость, чем содержание активного бентонита;
- содержание активного бентонита оказывает в 2 раза большее влияние на прочность формовочной смеси, чем влажность;
- содержание активного бентонита оказывает в 3 раза меньшее влияние на насыпную плотность формовочной смеси, чем влажность.

Полученные модели могут быть непосредственно использованы для формирования корректирующего воздействия в процессе смесеприготовления, так как они отражают связь свойств формовочной смеси с ее компонентами. На базе полученных эмпирических зависимостей возможно разработать алгоритм корректировки состава формовочной смеси, что позволит стабилизировать свойства формовочной смеси и повысить качество изготавливаемых отливок.

УДК 004

ПРИМЕНЕНИЕ ГРАФО-МАТЕМАТИЧЕСКОГО МЕТОДА «ДЕРЕВО ПРИНЯТИЯ РЕШЕНИЙ»

Фурманова А.В.

УО «Белорусский государственный экономический университет», г.Минск

Одной из популярных систем поддержки принятия решений является метод «дерево решений». С помощью этого метода можно принимать решения как по социальным и макроэкономическим вопросам, так и по вопросам развития предприятия или банковской сферы. «Дерево решений» также используются для решения диагностических задач в медицине, биологии, химии и генетике.

Однако построение «дерева решений» чаще всего используется для анализа проектных рисков – при оценке проектов с ограниченным числом возможных вариантов и известными значениями вероятностей того или иного события.

Таким образом, метод «дерево решений» - это один из методов построения систем поддержки принятия решений на основе правил вывода. Такие системы еще называются системами прямого логического вывода, так как анализ начинается с фактов, а результатом является тот или иной вывод.

В основе метода положена оценка альтернатив в случае, когда действие, предпринимаемое на одной стадии, зависит от действия, предпринятого на предыдущей стадии.

В качестве показателей метода «дерево решений» выступают вероятность альтернативы развития, её стоимость (NPV) или ожидаемая стоимостная оценка (EMV) — максимальное значение из сумм оценок выигрышей, умноженных на вероятность реализации выигрышей, для всех возможных вариантов, а также ряд других принципиально важных показателей.

«Дерево решений» — это графическое изображение процесса принятия решений, в котором отражены альтернативные решения, возможные факторы окружающей среды, соответствующие вероятности и выигрыши для любых комбинаций альтернатив и состояний среды.

Основное отличие «деревьев решений» от методов распознавания образов и моделирования состоит в том, что проводимое исследование основывается на логических рассуждениях, а не на вычислениях.

Алгоритм метода можно представить следующим образом:

- определение альтернатив развития событий;
- формулировка всех возможных решений, которые могут быть приняты в результате наступления каждого события;
- определение вероятности принятия каждого решения;
- определение стоимости каждого этапа.

На основании полученных данных строится «дерево решений», структура которого содержит узлы, представляющие собой ключевые события (точки принятия решений), и ветви, соединяющие узлы, - работы по реализации проекта.

Дерево изображается слева направо со следующей системой обозначений:

- (квадрат) - места принятия решений,
- (круг) - места появления исходов,
- (пунктир) – возможные решения,
- (сплошная) – возможные исходы.

Рассмотрим применение метода «дерева решений».

Экономическая проблемная ситуация: Управленческий аппарат банка должен принять решение об участии в инвестиционном проекте по финансированию капитального ремонта предприятия. Так как проект нацелен на долгосрочную перспективу, при условии, что экономическая ситуация в стране сложится удачно (налоговая политика, процентная ставка, политическая обстановка и т.д.), компания получит прибыль от реализации проекта в размере 500 млн. рублей. В противном случае, компания может потерять 250 млн. рублей. Согласно мнению экспертов, вероятность того, что стечение факторов окажется неблагоприятным - 70%. С другой стороны, существует альтернатива вложения только части запланированных средств на реконструкцию лишь основной производственной линии, что позволит впоследствии решать вопрос о дальнейшем финансировании проекта. Стоимость данной альтернативы - 40 млн. рублей. По расчетам экспертов существует 70% шансов, что реконструкция производственной линии будет эффективна, а следовательно 80% шансов, что весь проект окажется прибыльным. Если же реконструкция производственной линии себя не оправдает, то лишь с 10% вероятностью весь проект капитального ремонта может оказаться выгодным для инвесторов. Следует

ли участвовать в проекте по капитальному ремонту предприятия в целом? Следует ли первоначально проводить реконструкцию производственной линии? Какова ожидаемая стоимостная оценка наилучшего решения?

Решение.

Принимаемые решения:

- 1) первоначальная реконструкция основной производственной линии предприятия;
- 2) участие в капитальном ремонте предприятия;
- 3) отказ от участия в капитальном ремонте предприятия;
- 4) отказ от первоначальной реконструкции основной производственной линии предприятия.

Альтернативы развития:

- A. Проведение первоначальной реконструкции производственной линии;
- B. Участие в проекте капитального ремонта при успешной реконструкции основной производственной линии предприятия
- C. Отказ от участия в проекте при успешной реконструкции основной производственной линии предприятия;
- D. Участие в проекте капитального ремонта неудачной реконструкции основной производственной линии предприятия;
- E. Отказ от участия в проекте при неудачной реконструкции основной производственной линии предприятия;
- F. Участие в проекте капитального ремонта без первоначальной реконструкции производственной линии предприятия;
- G. Отказ от участия в проекте без первоначальной реконструкции производственной линии предприятия.

(+) – положительное решение (исход)

(-) – отрицательное решение (исход)

Схема метода «дерева решений» изображена на рис. 1.

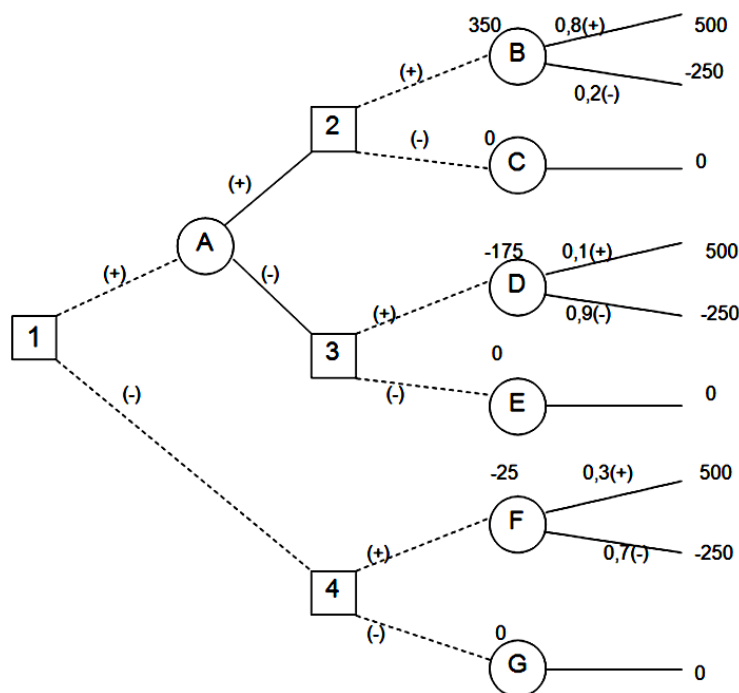


Рисунок 1 – Схема метода «дерево решений»

В узле F возможна эффективность всего проекта с вероятностью 0,3 (прибыль 500) и неудача его осуществления с вероятностью 0,7 (убыток -250) => рассчитывается оценка узла F.

$$EMV(F) = 0,3 \times 500 + 0,7 \times (-250) = -25 - \text{записываем над узлом F.}$$

$$EMV(G) = 0.$$

В узле 4 происходит выбор между участием в проекте ($EMV(F) = -25$) и неучастием в нем ($EMV(G) = 0$): $EMV(4) = \max \{EMV(F), EMV(G)\} = \max \{-25, 0\} = 0 = EMV(G)$ – записываем оценку над узлом 4, а решение об участии в проекте отбрасываем.

Аналогично:

$$EMV(B) = 0,8 \times 500 + 0,2 \times (-250) = 400 - 50 = 350.$$

$$EMV(C) = 0.$$

$EMV(2) = \max \{EMV(B), EMV(C)\} = \max \{350, 0\} = 350 = EMV(5)$. Поэтому в узле 2 отбрасываем возможное решение об игнорировании проекта.

$$EMV(D) = 0,1 \times 500 + 0,9 \times (-250) = 50 - 225 = -175.$$

$$EMV(E) = 0.$$

$EMV(3) = \max \{EMV(D), EMV(E)\} = \max \{-175, 0\} = 0 = EMV(E)$. Поэтому в узле 3 отбрасываем возможное решение об участии в проекте.

$$EMV(A) = 0,7 \times 350 + 0,3 \times 0 - 40 = 205.$$

$EMV(1) = \max \{EMV(A), EMV(4)\} = \max \{205; 0\} = 205 = EMV(A)$. Поэтому в узле 1 отбрасываем возможное решение об отказе от первоначальной реконструкции основной производственной линии.

Ожидаемая стоимостная оценка наилучшего решения равна 205 млн. рублей.

Таким образом, наиболее оптимальным решением является первоначальная реконструкция основной производственной линии предприятия. После чего при эффективности произведенной реконструкции производится дальнейший капитальный ремонт предприятия, в противном случае – участие в проекте прекращается.

Как известно, решение не всегда принимается при наличии полной информации, т.е. в условиях неопределенности. Поэтому метод «дерева решений» позволяет свести влияние неопределенности на принятие оптимального решения к минимуму. Чаще всего это происходит при решении новых, нетипичных проблем, когда требующие учёта факторы настолько новы и сложны, что о них невозможно получить достаточно информации. В такой ситуации можно попытаться получить дополнительную информацию и ещё раз проанализировать проблему с целью уменьшить её новизну и сложность либо полагаться на прошлый опыт и интуицию.

УДК 621.3.049.73.75:001.2(024)

УЧЁТ КОНФИГУРАЦИИ ПРИНЦИПАЛЬНЫХ ЭЛЕКТРИЧЕСКИХ СХЕМ В ПРОЦЕССЕ ПОСТРОЕНИЯ ИХ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ

Шандриков Анатолий Сергеевич

УО «Витебский государственный политехнический колледж», г. Витебск

Проектирование современных радиоэлектронных средств (РЭС) с многоуровневой конструктивной иерархией осуществляется с использованием систем автоматизированного проектирования (САПР), что предполагает применение математических моделей принципиальных электрических схем. На этапе компоновки в качестве таких моделей используется граф $G = (X, U)$, в котором множество вершин X заменяет множество радиоэлектронных компонентов (РЭК), а множество рёбер U – электрические связи между ними в соответствии с принципиальной электрической схемой [1, 2].

В процессе построения графа принципиальной электрической схемы полные подграфы, интерпретирующие электрические узлы, заменяются связывающими деревьями путём удаления «лишних» рёбер, количество которых определяется цикломатическим числом. Далее решение задачи компоновки сводится к разрезанию полученного графа на куски с заданным количеством вершин в кусках.

Одним из основных критериев качества результатов компоновки РЭС является минимум внешних связей между сформированными кусками или коэффициент разрезания графа, определяемый по формуле:

$$D(G) = Y_{L_{ij}}/K,$$

где $Y_{L_{ij}}$ – суммарное количество связей между вершинами внутри сформированных кусков графа (внутренних связей); K – количество связей между сформированными кусками графа (внешних связей).

Повысить в некоторой степени оптимальность результата можно ещё на этапе построения математической модели проектируемого РЭС, точнее, на этапе построения связывающих деревьев полных подграфов электрических узлов принципиальной электрической схемы. Каждый полный подграф заменяется любым из $d = n^n - 2$ возможных вариантов связывающих деревьев. Любой из числа d вариантов связывающих деревьев будет эквивалентным с точки зрения требований дискретной математики к связности подграфа и отсутствию в нём циклов. По этой причине выбор конкретного варианта связывающего дерева чаще всего осуществляется произвольно, без какой-либо предварительной оценки последующих результатов разрезания графа независимо от наличия или отсутствия накладываемых технологических ограничений. Это объясняется неочевидностью влияния выбранного варианта связывающего дерева на получение оптимальных результатов разрезания графа, отсутствием пояснений по данному вопросу в литературных источниках [1, 2] и в их библиографических описаниях.

В большинстве случаев принципиальная электрическая схема РЭС содержит некоторое количество групп РЭК, имеющих между собой более одной связи, например, в случае параллельного соединения и т.п. Очевидно, что оптимальной компоновкой РЭС будет та, в результате которой параллельно соединённые РЭК окажутся в одном модуле, например, на одной плате. Для оптимизации результатов решения задач разрезания и размещения графа необходимо анализировать возможные варианты связывающих деревьев и выбирать наиболее подходящий вариант в каждой конкретной ситуации [3].

Рассмотрим зависимость результатов разрезания графа от выбранного варианта конфигурации связывающих деревьев в узлах параллельного соединения РЭК.

В качестве примера на рис. 1 представлен упрощённый вариант принципиальной электрической схемы звукового сигнализатора [4]. Предположим, что звуковой сигнализатор необходимо скомпоновать в три блока по 5, 6 и 7 РЭК при отсутствии каких-либо технологических ограничений, и для решения данной задачи построим математическую модель в виде графа.

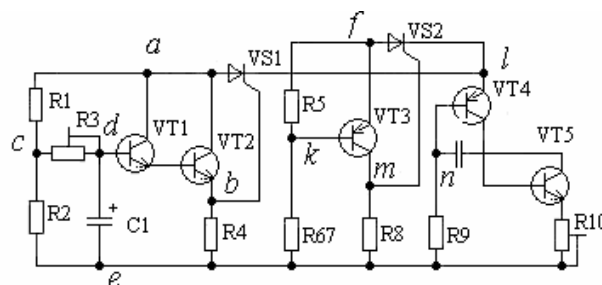


Рисунок 1

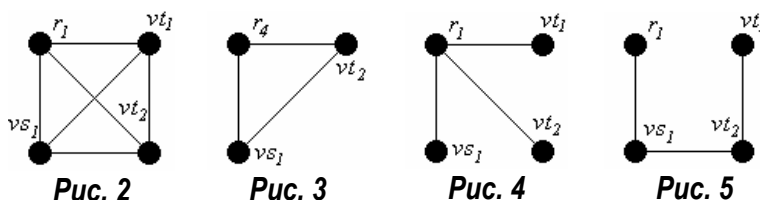


Рис. 2

Рис. 3

Рис. 4

Рис. 5

В данной принципиальной электрической схеме транзисторы VT1, VT2 и тиристор VS1 имеют по два соединения: транзистор VT2 и тиристор VS1 соединены в узлах *a* и *b*. Коллекторы транзисторов VT1 и VT2 соединены в узле *a*, а эмиттер транзистора VT1 связан с базой транзистора VT2 отдельной ветвью. Полные подграфы узлов *a* и *b* представлены на рис. 2 и 3 соответственно.

Для узла *a* из 16 возможных могут быть выбраны любые варианты связывающих деревьев, но мы рассмотрим только два из них: первый вариант – случайный, вершины *vt1*, *vt2* и *vs1*, интерпретирующие транзисторы VT1 и VT2 и тиристор VS1, не соединены маршрутом и второй вариант – проанализированный, когда эти вершины соединены маршрутом. Первый и второй варианты связывающих деревьев для подграфа G_a узла *a* представлены на рис. 4 и 5 соответственно.

Аналогично, для узла *b* в одном из вариантов связывающего дерева может отсутствовать связь между вершинами *vt2* и *vs1* (первый вариант, рис. 6), а может быть, и сохранена (второй вариант, рис. 7).

Сходная ситуация характерна и для попарно соединённых в узлах *f*, *k* и *m* резистора R5, транзистора VT3 и тиристора VS2. Резистор R5 и тиристор VS2 одной связью соединены в узле *k*. Первый вариант связывающего дерева для подграфа G_k при отсутствии ребра $u(r_5, vt_3)$, представлен на рис. 8, а второй, проанализированный, вариант, т.е. с сохранённым ребром $u(r_5, vt_3)$ – на рис. 9.

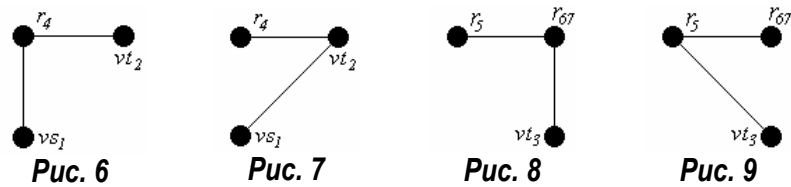


Рис. 6

Рис. 7

Рис. 8

Рис. 9

Соответственно, первый (случайный) и второй (проанализированный) варианты связывающего дерева для подграфа G_m узла *m* представлен на рис. 10 и 11.

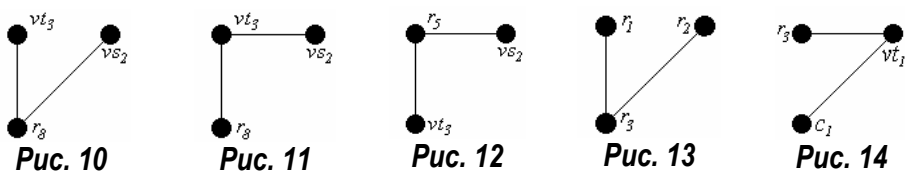


Рис. 10

Рис. 11

Рис. 12

Рис. 13

Рис. 14

Узел *f* отличается от всех рассмотренных узлов тем, что в нём соединены только те РЭК, которые имеют между собой параллельное соединение, поэтому все рёбра в полном подграфе G_f узла *f* равноценны, т.е. нет вариантов связывающих деревьев, среди которых в результате анализа можно было бы выделить лучший. В этом случае можно удалить любое ребро и рассматривать только один вариант связывающего дерева подграфа G_f , например, представленный на рис. 12.

Для подграфов G_c и G_d анализ возможных вариантов связывающих деревьев особого значения не имеет. При сохранении в подграфе G_c ребра $u(r_1, r_3)$ и выбранном «лучшем» варианте связывающего дерева для узла *a* (рис. 5) возрастает вероятность распределения резистора R1 в один блок с резистором R3, транзисторами VT1 и VT2 и тиристором VS1. В противном случае, в одном блоке с транзисторами VT1 и VT2, резистором R3 и конденсатором C1 может оказаться резистор R2. Количество внешних связей между

сформированными блоками при этом не изменяется, поэтому достаточно выбрать любой возможный вариант. Аналогичная ситуация складывается и в узле d . Варианты связывающих деревьев для подграфов G_c и G_d представлены на рис. 13 и 14.

Подграф G_e узла e содержит наибольшее количество вершин. Один из возможных «случайных» вариантов связывающего дерева этого подграфа представлен на рис. 15.

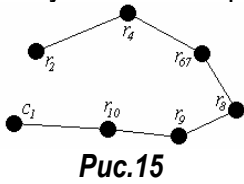


Рис. 15

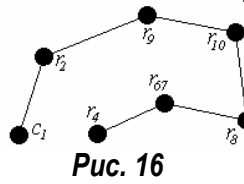


Рис. 16

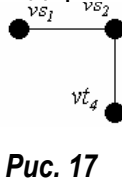


Рис. 17

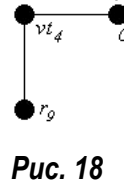


Рис. 18

Учитывая заданное условие компоновки и конкретную конфигурацию принципиальной электрической схемы звукового сигнализатора «разрыв» в

дереве целесообразно оставить между вершинами c_1 и r_4 (рис. 16) или между вершинами r_8 и r_9 .

Для узлов l и n любые варианты связывающих деревьев будут равнозначными, как, например, представленные на рис. 17 и 18.

По результатам построения «случайных» вариантов связывающих деревьев для каждого электрического узла схемы был сформирован граф $G^{(1)}$, представленный на рис. 19. На рис. 20 представлен граф $G^{(2)}$, построенный с учётом конфигурации соединений принципиальной электрической схемы.

Как видно из рис. 20, учёт связей между параллельно соединёнными РЭК в каждом электрическом узле схемы приводит к построению мультиграфа $G^{(2)}$.

Результаты разрезания сформированных графов $G^{(1)}$ и $G^{(2)}$ на куски G_1, G_2 и G_3 , содержащие $n_1 = 5, n_2 = 6$ и $n_3 = 7$ вершин, представлены на рис. 21 и 22 соответственно.

Коэффициенты разрезания графов: $D(G^{(1)}) = 16/13 = 1,23$; $D(G^{(2)}) = 21/8 = 2,63$.

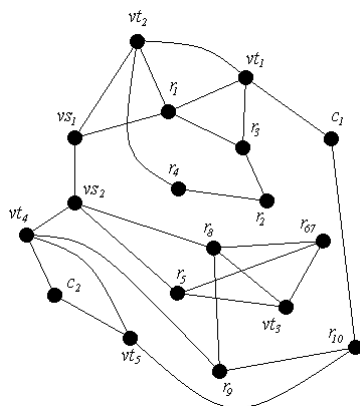


Рис. 19

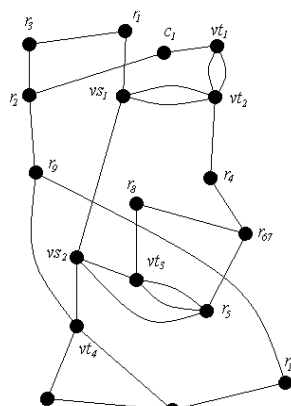


Рис. 20

Разрезание графов $G^{(1)}$ и $G^{(2)}$ осуществлялось с использованием последовательного алгоритма формирования заданных кусков. Этот алгоритм отличается высокой производительностью, но в большинстве случаев результаты его работы не оптимальны [1, 2].

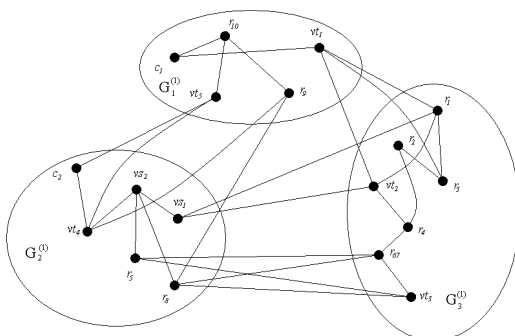


Рис. 21

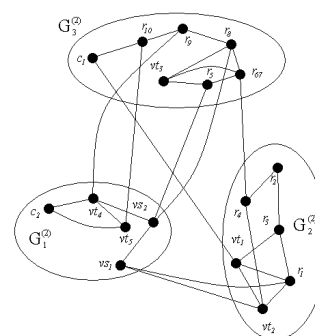


Рис. 22

В результате разрезания графов $G^{(1)}$ и $G^{(2)}$ с использованием наиболее эффективного современного алгоритма [5] были получены коэффициенты разрезания $D(G^{(1)}) = 19/10 = 1,9$; $D(G^{(2)}) = 24/5 = 4,8$. Количество внешних связей между кусками графа $G^{(2)}$ в обоих случаях на пять рёбер меньше, чем между кусками графа $G^{(1)}$.

В результате многократных исследований было установлено, что обработка инвариантных моделей принципиальных электрических схем в виде графа обеспечивает различающиеся по количеству внешних связей результаты компоновки независимо от выбранного метода разрезания графов. На основании этого можно сделать вывод: при построении математической модели принципиальной электрической схемы необходимо учитывать реальную конфигурацию связей между РЭК.

Литература

1. Методы разбиения схем РЭА на конструктивно законченные части / К.К. Морозов [и др.]; под ред. К.К. Морозова. – М.: Сов. радио, 1978. – 136 с., ил.
2. Применение графов для проектирования дискретных устройств / А.Н. Мелихов [и др.]. – М.: Наука, 1974. – 304 с., ил.
3. Шандриков А.С. Особенности построения графа принципиальной электрической схемы, влияющие на результаты компоновки РЭС / А.С. Шандриков // Современная радиоэлектроника: научные исследования, подготовка кадров: материалы международной научно-практической конференции: в 3 ч. Ч 1, Минск, 20-21 апреля 2006 г. / Минский государственный высший радиотехнический колледж. – Мн.: 2006. – С. 354-358.
4. Кац, Л. Звуковой сигнализатор предельных режимов автомобиля: сборник. Вып. 102 / сост. В.А. Никитин – М.: ДОСААФ, 1988. – С. 59-63.
5. Шандриков, А.С. Алгоритм разрезания графа методом последовательного назначения вершин в формируемые куски / А.С. Шандриков // Веснік Віцебскага дзяржаўнага ўніверсітэта імя П.М. Машэрава. – 2005. – № 4(38). – С. 111-118.

УДК 556.166

ПОСТРОЕНИЕ ПРОГНОЗНЫХ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ МАКСИМАЛЬНЫХ РАСХОДОВ ВОДЫ ДОЖДЕВЫХ ПАВОДКОВ РЕК БЕЛОРУССКОГО ПОЛЕСЬЯ

Шелест Т.А.

УО «Брестский государственный университет имени А.С. Пушкина», г. Брест

Дождевые паводки представляют одно из опасных гидрологических явлений, которые, в отличие от весенних половодий, не приурочены к какому-либо сезону года. Они возникают внезапно и развиваются стремительно, нередко причиняют ущерб экономике республики, особенно сельскому хозяйству. Паводки отмечаются во многих регионах земного шара. Проблема паводков является весьма актуальной и для территории Беларуси, особенно для ее южной части – Полесья. Неглубокое залегание грунтовых вод, плоский, практически без уклонов, рельеф создают условия, при которых даже относительно небольшие дожди вызывают затопления в понижениях поймы.

Максимальные расходы воды паводков относятся к категории наиболее сложных для исследования и обобщения гидрологических характеристик. Это связано с нерегулярностью и быстротечностью прохождения паводков, а также с тем, что они бывают неожиданными и распространяются по территории очень дискретно, захватывая часто лишь небольшие площади. Все это затрудняет прогнозирование и минимизацию негативных последствий от их воздействий.

В связи с ростом хозяйственного освоения речных долин и пойм ущерб от паводков растет. Поэтому вопросы прогнозирования их приобретают особую актуальность.

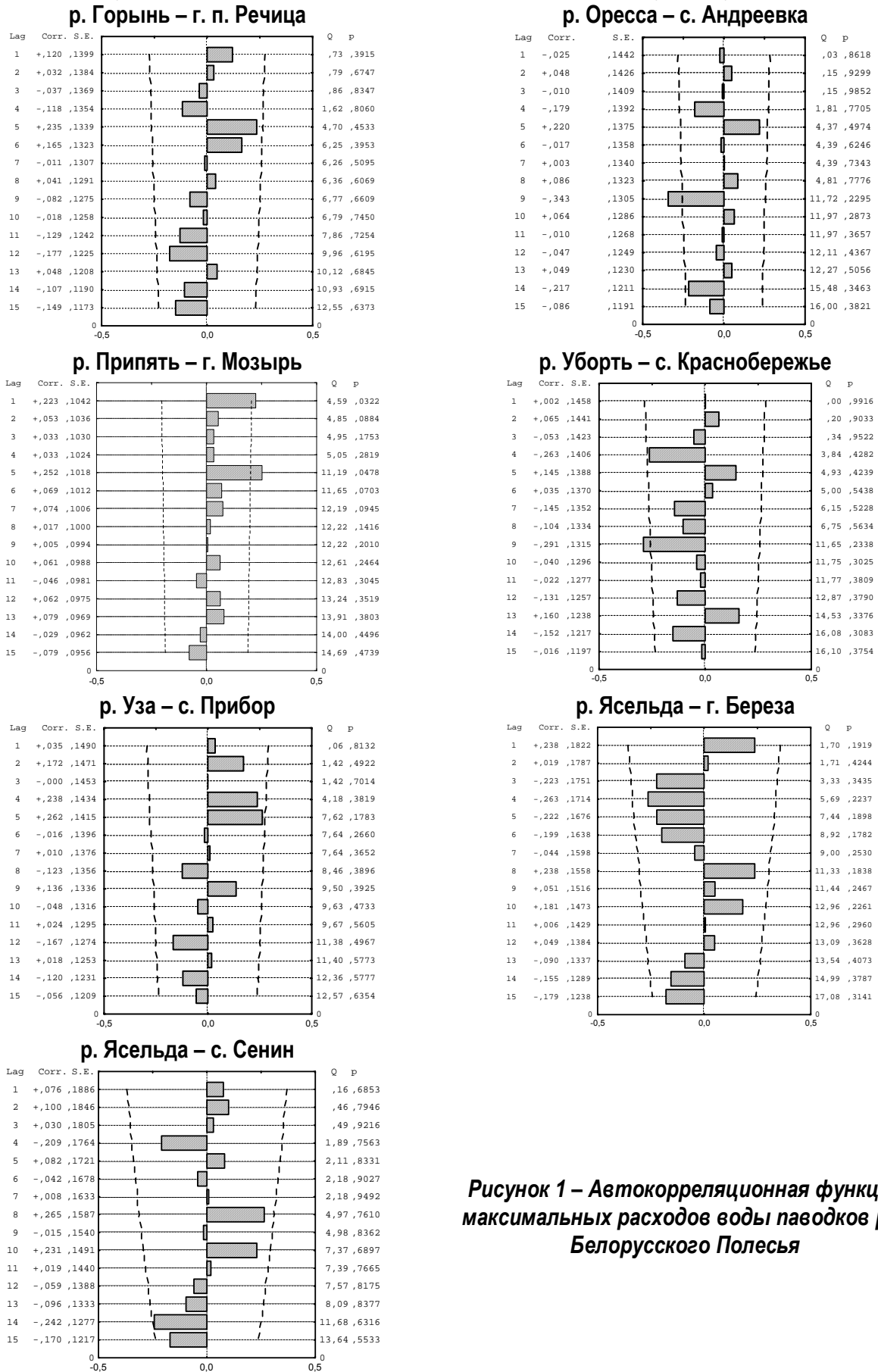


Рисунок 1 – Автокорреляционная функция максимальных расходов воды паводков рек Белорусского Полесья

Прогноз дождевых паводков является одним из важнейших направлений гидрологических прогнозов. В связи с тем, что дождевые паводки возникают внезапно, даже небольшая заблаговременность их предсказания во многих случаях позволяет сократить ущербы.

Целью настоящего исследования является выявление цикличности и построение прогнозных математических моделей максимальных расходов воды дождевых паводков рек Белорусского Полесья. Исходными данными послужили многолетние ряды наблюдений за максимальными расходами воды паводков рек Полесья.

Процессы паводкообразования обладают региональной обусловленностью, связанной с локальностью выпадения осадков, вызывающих паводки. Поэтому универсальную математическую модель, пригодную повсеместно, построить не представляется возможным. Модели паводков должны иметь региональный характер. В связи с этим модели строились отдельно по речным бассейнам.

Для описания многолетних колебаний максимального стока воды рек Полесья использовалась сложная цепь Макова (допускается наличие ближних и дальних внутрирядных связей) со сдвижкой до 35 лет. В этом случае важным этапом для описания колебаний является автокорреляционная функция, позволяющая устанавливать циклы.

В результате исследования установлено, что для прогнозных оценок наиболее приемлемой является сложная цепь Маркова. Полученные модели представлены в таблице.

Таблица 1– Прогнозные математические модели максимальных расходов воды дождевых паводков рек Белорусского Полесья

Река–створ	R	Модель
Горынь–Речица	0,51	$Q_{t+1} = 360 + 0,31 \cdot Q_{t-6} - 0,29 \cdot Q_{t-12} - 0,35 \cdot Q_{t-30} - 0,29 \cdot Q_{t-33}$
Оресса–Андреевка	0,45	$Q_{t+1} = 53,4 - 0,36 \cdot Q_{t-9} - 0,32 \cdot Q_{t-25}$
Припять–Мозырь	0,34	$Q_{t+1} = 207 + 0,22 \cdot Q_{t-1} + 0,25 \cdot Q_{t-5}$
Уборть–Краснобережье	0,35	$Q_{t+1} = 56,3 + 0,36 \cdot Q_{t-16}$
Уза–Прибор	0,81	$Q_{t+1} = 9,09 - 0,19 \cdot Q_{t-14} - 0,26 \cdot Q_{t-15} + 0,38 \cdot Q_{t-16} -$ $- 0,23 \cdot Q_{t-23} - 0,25 \cdot Q_{t-25} - 0,21 \cdot Q_{t-28}$
Ясельда–Береза	0,91	$Q_{t+1} = 19,8 + 0,22 \cdot Q_{t-6} - 0,24 \cdot Q_{t-12} - 0,15 \cdot Q_{t-20} - 0,13 \cdot Q_{t-21} -$ $- 0,22 \cdot Q_{t-27} + 0,14 \cdot Q_{t-30} - 0,45 \cdot Q_{t-33} - 0,35 \cdot Q_{t-34}$
Ясельда–Сенин	0,86	$Q_{t+1} = 61,1 - 0,24 \cdot Q_{t-19} - 0,24 \cdot Q_{t-21} - 0,21 \cdot Q_{t-26} -$ $- 0,32 \cdot Q_{t-32} - 0,17 \cdot Q_{t-34}$

где Q_t – максимальные расходы воды паводков в t -й год, R – коэффициенты множественной корреляции.

Для полученных моделей коэффициент множественной корреляции принимает значения от 0,34 (Припять–Мозырь) до 0,91 (Ясельда–Береза). Наиболее высокие показатели коэффициента множественной корреляции отмечены для реки Ясельда, что связано с большой зарегулированностью стока. Анализ автокорреляционных функций показал, что для моделей характерно наличие в основном дальних внутрирядных связей.

Таким образом, прогнозные модели, построенные с использованием сложных цепей Маркова, в большинстве случаев дают приемлемые результаты.

Литература

1. Пузаченко, Ю.Г. Математические методы в экологических и географических исследованиях / Ю.П. Пузаченко – М.: Академия, 2004. – 416 с.
2. Рождественский, А.В. Статистические методы в гидрологии / А.В. Рождественский, А.И. Чеботарев – Л.: Гидрометеоздат, 1974. – 424 с.

УДК 51-74

АЛГОРИТМ ВОССТАНОВЛЕНИЯ ВРЕМЕННЫХ РЯДОВ УРОВНЕЙ ГРУНТОВЫХ ВОД

Шешко Н.Н.

УО «Брестский государственный технический университет», г. Брест

Современное развитие вычислительной техники и математических методов позволяет по-новому взглянуть на решение известных проблем природопользования. Используя стандартную гидрометеорологическую информацию, можно выявить скрытые закономерности в формировании тех или иных природных процессов.

В данной работе ставится задача разработки алгоритма наиболее полного восстановления и продления данных наблюдений за уровнями грунтовых вод (УГВ).

Данными для реализации задач исследования послужили результаты многолетних инструментальных наблюдений за УГВ, полученные Гидрогеологической экспедицией Республики Беларусь на территории НП «Беловежская пуца». Наблюдения ведутся на более чем 70 гидрогеологических скважинах. В рамках настоящего исследования использовано 55 скважин, расположенных в наиболее характерных районах Беловежской пуцы. Глубина закладки скважин варьируется в широких пределах (2...140 м) в зависимости от гидрогеологических условий. Наблюдения за УГВ до 1986 года проводились ежедневно, но в последующие годы в связи с уменьшением финансирования замеры начали проводиться раз в два дня. В связи с тем, что колебания грунтовых и подземных вод являются достаточно инертным процессом, дискретность является допустимой. В хронологическом ходе среднемесячных УГВ по всем гидрогеологическим скважинам имеют место пропуски, которые для условия решаемых задач должны быть восстановлены. Кроме того, для совместного анализа временных рядов немаловажным является наличие равных периодов наблюдений по всем скважинам. Для большинства скважин, кроме скв. №№ 517... 520, 522, 562, 563, 564, 770...777, наблюдения начаты с 1971 года, что на 6 лет меньше, чем у вышеупомянутых скважин. Период с 1965 по 1974 год представляет большой интерес в связи с тем, что с середины 70-х годов прошлого столетия в пределах парка и на прилегающих территориях начиналось крупномасштабное гидромелиоративное строительство, повлекшее изменение гидрологического и гидрогеологического режимов.

В таблице приведено общее количество пропусков в данных наблюдений за среднемесячными УГВ по скважинам. Согласно нормативным документам, в ряду может быть не более 30 % восстановленных значений. Из таблицы видно, что по всем скважинам это условие выполняется, кроме скв. № 1349. Поэтому, результаты, полученные по данной скв. № 1349, требуют дополнительного анализа.

Восстановление отсутствующих данных наблюдений за среднемесячными значениями УГВ проводилось по алгоритму, представленному в виде блок-схемы (рисунок).

В основе данного алгоритма положено три методики подбора аналога. Применение данных методик выстроено в порядке их наибольшей достоверности. В первую очередь для исходной скважины за конкретный месяц подбирается скважина-аналог, с учетом обязательного выполнения ряда условий:

$$\left\{ \begin{array}{l} n' \geq 10 \\ r \geq R_{krit} \vee R \geq R_{krit} ; \\ \frac{k}{\sigma_k} \geq 2 \end{array} \right. \quad (1)$$

Таблица 1 – Количество пропущенных значений в данных наблюдения за среднемесячным УГВ

№ по порядку	№ скважины	Количество отсутствующих среднемесячных значений		№ по порядку	№ скважины	Количество отсутствующих среднемесячных значений	
		месяцев	%			месяцев	%
1	633	87	17	29	1352	148	29
2	653	70	13	30	1353	137	26
3	654	86	17	31	634	76	15
4	655	80	15	32	637	77	15
5	656	87	17	33	638	101	19
6	657	93	18	34	517	7	1
7	658	118	23	35	518	17	3
8	659	97	19	36	519	14	3
9	661	113	22	37	520	16	3
10	704	79	15	38	522	16	3
11	705	87	17	39	562	20	4
12	706	91	18	40	563	30	6
13	707	101	19	41	564	6	1
14	708	111	21	42	770	18	3
15	710	83	16	43	771	27	5
16	711	92	18	44	772	16	3
17	712	86	17	45	773	16	3
18	646	87	17	46	774	27	5
19	647	96	18	47	775	25	5
20	648	92	18	48	776	49	9
21	649	80	15	49	777	16	3
22	650	78	15	50	662	89	17
23	651	87	17	51	663	89	17
24	652	78	15	52	664	98	19
25	1348	146	28	53	665	89	17
26	1349	172	33	54	666	90	17
27	1350	138	27	55	667	118	23
28	1351	146	28				

Кроме соблюдения условия (1), необходимо иметь значения УГВ по скважине-аналогу в расчетный момент времени. Проверка данного условия проводится до тех пор, пока все возможные скважины-аналоги не будут проверены на их применимость. В том случае, если ни одна скважина-аналог не соответствует вышеприведенному условию, переходят ко второму методу. Второй метод основан на использовании одновременно нескольких скважин-аналогов. Постепенно производят увеличение количества используемых аналогов до тех пор, пока прирост коэффициента множественной корреляции не станет меньше заданного числа β . После чего повторно проводится проверка условия (1).

При повторном невыполнении условия переходят к третьему методу, который основан в подборе месяца-аналога исходной скважины (используются ряды наблюдений только по исходной скважине). Выбрав месяц-аналог, имеющий наиболее тесную связь с исходным месяцем, выполняют проверку условия (1).

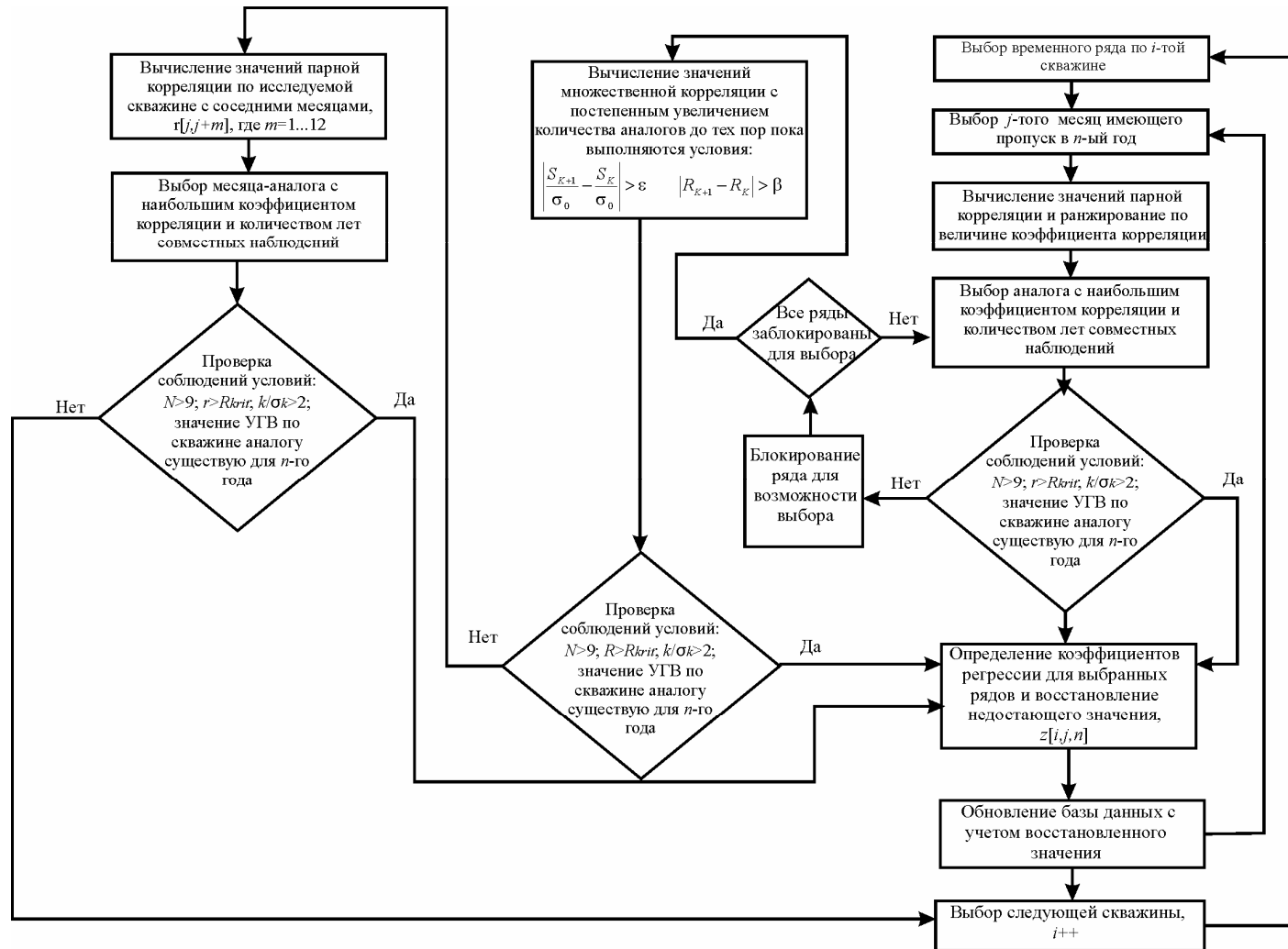


Рисунок 1 – Алгоритм восстановления пропусков в наблюдениях за грунтовыми и подземными водами

На рисунке применены следующие обозначения: $z[i,j,n]$ – величина УГВ по i -той j -того месяца скважине за n -ый год; N – число лет совместных наблюдений; r – коэффициент парной корреляции; R – коэффициент множественной корреляции K -того порядка; R_{crit} – критическое значение коэффициента корреляции при заданном уровне значимости; k – коэффициент регрессии; σ_k – среднее квадратическое отклонение коэффициента регрессии; S_k и S_{k+1} – относительные ошибки при использовании K и $K+1$ аналогов; R_k и R_{k+1} – аналогичные коэффициенты множественной корреляции; ε и β – заданные числа; σ_0 – среднеквадратическое отклонение ряда

Реализация алгоритма возможна с привлечением компьютерного программирования. В данном случае использована компьютерная программа Mathematica. Таким образом возможно восстановить пропущенные значения среднемесячных УГВ в зависимости от критического значения коэффициента корреляции, который приведен в [1].

В результате того, что постоянно обновляется исходная база данных за счёт восстановленных значений, возможно реализовать данную задачу в полном объеме.

Литература

1. Определение расчётных гидрологических характеристик: Пособие П1-98 к СНиП 2.01.14-83. – Минск, 2000 – 220с.

УДК 519.6

МЕТОДЫ ДИСКРЕТИЗАЦИИ СЛОЖНЫХ ПРОСТРАНСТВЕННЫХ ОБЛАСТЕЙ

Щеглов И.А.

Институт прикладной математики им. М.В. Келдыша РАН, г. Москва, Россия

В связи со значительным прогрессом в области вычислительной техники основным инструментом математического моделирования различных физических процессов стали численные методы, среди которых наибольшее распространение получили проекционно-сеточные методы, в частности метод конечных элементов (МКЭ) и его многочисленные варианты. Использование проекционно-сеточных методов предполагает предварительное построение т.н. расчетной сетки, т.е. ее дискретизацию на множество элементов определенной формы (в качестве элементов сетки, как правило, используются геометрические симплексы, т.е. треугольники в двумерном и тетраэдры в трехмерном случае).

Все методы триангуляции по принципу построения можно разбить на две большие группы: прямые методы [1] и итерационные методы [2]. В прямых методах сетка строится за один этап, причем ее топология (граф связей между узлами) и координаты всех узлов известны изначально. Как правило, такие методы основаны на использовании шаблона дискретизации области определенной формы (шар, куб, цилиндр и т.п.), поэтому не являются универсальными. Итерационные методы, напротив, универсальны и, как правило, применимы для областей достаточно произвольной формы. Сетка такими методами строится последовательно, элемент за элементом, причем координаты узлов и топология сетки могут меняться в процессе построения. Разработано несколько различных подходов, которые можно разделить на три класса: методы граничной коррекции, методы на основе критерия Делоне и методы исчерпывания.

Построение сеток в методах граничной коррекции осуществляется в два этапа. На первом этапе производится триангуляция некоей простой "суперобласти", полностью включающей в себя заданную область. Как правило, эта суперобласть представляет собой прямоугольник или параллелепипед, триангуляция которого осуществляется на основе одного из многочисленных шаблонов. На втором этапе все узлы полученной сетки, лежащие вблизи границы заданной области, проецируются на поверхность границы; а узлы, лежащие вне заданной области - удаляются. Недостатками этих методов являются неизбежные геометрические искажения элементов сетки вблизи границ, а также невозможность дискретизации областей с заданной триангуляцией границы [3].

Сущность методов исчерпывания заключается в последовательном "вырезании" из заданной области фрагментов необходимой формы до тех пор, пока вся область не окажется "исчерпана". Исходными данными на каждой итерации является "фронт", то есть триангуляция границы еще не "исчерпанной" части области. Каждый элемент границы является основанием для построения нового элемента сетки. При этом на каждой итерации может изыматься либо один элемент, либо сразу целый слой элементов. После изъятия элемента (-ов) "фронт" обновляется, после чего происходит переход к следующей итерации. Методы исчерпывания универсальны и могут быть использованы для областей произвольной формы и конфигурации (даже для несвязных областей), что объясняет их популярность. Вместе с тем следует отметить их высокую ресурсоемкость и низкую скорость работы [4].

Методы на основе критерия Делоне часто называют просто методами Делоне, хотя это не совсем корректно, поскольку сам Б.Н. Делоне никаких методов не разрабатывал, а лишь предложил простой и эффективный критерий, использующийся при установке связей между узлами. Соответственно, идеей этого класса методов является размещение в заданной области узлов и последующая расстановка между ними связей согласно критерию Делоне (либо иному схожему критерию). В двумерном случае этот подход получил наибольшую популярность, поскольку он позволяет быстро и эффективно конструировать сетки с априори высоким качеством триангуляции. Однако при переходе к трем измерениям исследователи столкнулись с рядом проблем, затрудняющих использование этого критерия. Тем не менее, эти методы получили достаточно хорошее развитие и пользуются заслуженной популярностью [5].

Области решения многих задач современной вычислительной математики являются *сложными*, т.е. состоят из нескольких подобластей или имеют различные внутренние или поверхностные ограничения, обусловленные резким изменением свойств материала области, конструктивными особенностями моделируемого объекта или самой постановкой задачи. При построении расчетных сеток в таких областях необходимо учитывать эти особенности и использовать особые подходы к дискретизации. Например, при решении задачи об упруго-пластичной деформации композитного материала методом конечных элементов сетку в области требуется строить так, чтобы каждый элемент сетки целиком принадлежал либо матрице, либо включению. Это условие равнозначно тому, что ребра сетки не должны пересекать поверхности раздела между матрицей и включением. Для дискретизации таких областей требуется использовать особые подходы [6].

Всего можно предложить два основных подхода к решению этой проблемы. Первый подход заключается в предварительном построении сетки без учета ограничений и последующей локальной перестройке сетки для учета этих ограничений путем перемещения узлов (как в методе граничной коррекции) и/или разбиения пересекаемых ограничениями элементов на более мелкие. Второй подход основан на том факте, что входными данными для методов исчерпывания является триангуляция границы. Т.е. если предварительно все линии границы и ограничений разбить на отрезки, далее на основе этого разбиения триангулировать границу и поверхности ограничений (двумерным вариантом алгоритма исчерпывания) и использовать полученные триангуляции в качестве входных данных для дискретизации подобластей (трехмерным вариантом алгоритма исчерпывания), можно автоматически получить согласованную дискретизацию всей области.

Именно второй подход использован в разработанном автором программном комплексе под названием "Gridder". Комплекс позволяет строить сетки (в том числе со сгущением) в сложных двумерных и трехмерных областях, а также содержит утилиты для визуализации и оценки качества сеток.

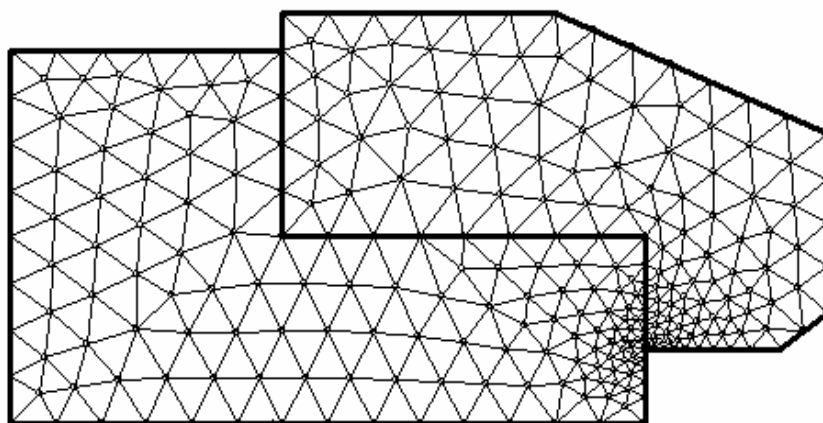


Рисунок 1 – Пример дискретизации сложной области (ПК "Gridder")

Работа выполнена при частичной финансовой поддержке РФФИ (проект № 09-01-00151).

Литература

1. Галанин, М.П. Разработка и реализация алгоритмов трехмерной триангуляции сложных пространственных областей: прямые методы / М.П. Галанин, И.А. Щеглов – 2006. 32с. (Препринт ИПМ им. М.В. Келдыша РАН; № 10).
2. Галанин, М.П. Разработка и реализация алгоритмов трехмерной триангуляции сложных пространственных областей: итерационные методы / М.П. Галанин, И.А. Щеглов – 2006. 32с. (Препринт ИПМ им. М.В. Келдыша РАН; № 9).
3. Шайдуров, В.В. Многосеточные методы конечных элементов. – М.: Наука, 1989. – 288 с.
4. S.H. Lo. Volume Discretization into Tetrahedra - II. 3D Triangulation by Advancing Front Approach // *Computers and Structures*, Pergamon, Vol. 39, № 5, P. 501-511. – 1991.
5. T.J. Baker. Automatic Mesh Generation for Complex Three-Dimensional Regions Using a Constrained Delaunay Triangulation // *Engineering With Computers*, Springer-Verlag, № 5. – 1989. – P. 161-175.
6. Дворников, М.В. Триангуляция произвольной многосвязной области со сложной границей / М.В. Дворников, В.Ф. Тишкин, А.Ю. Филатов // М.: Инст. математ. моделир. РАН. – 1995. – №7.

УДК 621.391.83

ПЕРЕХОД ОТ НЕПРЕРЫВНОЙ МОДЕЛИ АЧХ МНОГОЛУЧЕВОЙ ИОНОСФЕРНОЙ КВ РАДИОЛИНИИ К ДИСКРЕТНОЙ МОДЕЛИ

Щирый А.О.

Марийский государственный университет, г. Йошкар-Ола, Россия

Коротковолновым (КВ) ионосферным радиополосам протяженностью сотни-тысячи километров присуща многолучевость распространения радиосигнала. Использование узкополосных каналов в традиционных системах связи не позволяет разделять по задержке принимаемые парциальные моды, соответствующие разным лучам, и они интерферируют между собой, в результате амплитудно - частотная характеристика (АЧХ) линии связи становится изрезанной. Ей присущи подъемы, соответствующие полосам конструктивной интерференции, и провалы, обусловленные деструктивной интерференцией. Для устойчивой работы радиотехнической системы ее радиоканал должен быть вложен в полосу конструктивной интерференции. Таким образом, возникает проблема определения АЧХ линии связи с высоким частотным разрешением. Использование широкополосных

сигналов (в широкополосных каналах) позволяет разделить принимаемые моды и, кроме того, предполагает оптимальную обработку сигнала в приемнике. Для компенсации искажений широкополосных радиосигналов также должна быть решена задача определения АЧХ радиолинии. Наклонное зондирование ионосферы (НЗИ) широкополосными непрерывными сигналами с линейно-частотной модуляцией (ЛЧМ) потенциально позволяет получить полную информацию о прохождении сигнала и наличии помех в радиоканале [1].

В [2] в приближении геометрической оптики получено выражение для АЧХ многолучевой КВ радиолинии, как модуля передаточной функции. Передаточная функция многолучевой КВ радиолинии равна сумме m передаточных функций отдельных мод, при этом параметрами модели для каждой моды являются зависимости времени группового запаздывания $\tau_j(f)$ и амплитуды $|H_j(f)|$ для каждой j -ой моды, а также начальные фазы мод $\varphi_{0j}(f, t)$; указанные параметры заданы для канала с величиной частотной полосы Δf . В результате аналитическое выражение для оценки АЧХ КВ радиолинии имеет следующий вид:

$$|H(f_p, t)| = \left| \sum_{j=1}^m |H_j(f_p)| \exp \left[-i \left(\varphi_j(f_p, t_0) + 2\pi \cdot \int_{f_p}^{f_p + \Delta f} \tau_j(f) df + 2\pi \cdot F_{ди} \cdot \Delta t \right) \right] \right| \quad (1)$$

Также в выражении фазы учитывается вклад доплеровского смещения частоты $F_{ди}$. Оно влияет не на форму АЧХ, а на ее дрейф по оси частот во времени t .

Практически возможно получить точечные оценки параметров (как правило, точечные оценки параметров $\tau_j(f)$ и $|H_j(f)|$ по ионограмме НЗИ получают с дискретом 10-100 кГц). Поэтому для получения передаточной функции радиолинии с высоким частотным разрешением (10-100 Гц) возникает проблема ее восстановления.

Как следует из формулы (1), для этого нужно решить задачу получения по дискретным моделям $\tau_j(f_p)$ и $|H_j(f_p)|$ их непрерывных моделей. Заметим, что дискретные модели (т.е. точечные оценки) содержат регулярную и случайную компоненты. При этом для получения непрерывной модели $|H_j(f)|$ воспользуемся кусочно-линейной интерполяцией, а для $\tau_j(f_p)$ удобнее получить аналитическое выражение непрерывной модели, так как $\tau_j(f_p)$ стоит в подынтегральном выражении.

Дискретная модель $\tau_{Dj}(f)$ определяется множеством точек с координатами $(f_i, \tau_{Dj}(f_i))$ или, иначе говоря, последовательностью $\{\tau_{Dj}(f_i)\}$, где $t = 0, 1, \dots, N$. Задачу построения и анализа непрерывной модели $\tau_j(f)$ можно свести к минимизации функционала $\min_{\tau \in T_{fp}} J(\tau)$ [3]:

$$J(\tau) = \sum_{i=0}^N [\tau_{Dj}(f_i) - \tau(f_i)]^2, \quad (2)$$

где τ – искомая непрерывная функция; T_{fp} – множество непрерывно дифференцируемых на отрезке $[f_p, f_p + \Delta f]$ функций.

Решим эту задачу, используя полиномы Чебышева [3,4]. Они применяются потому, что для них, в отличие от полиномов вообще, строго доказана сходимость данного метода.

Полиномы Чебышева $P_n(x)$, $n \geq 0$ определяются соотношениями:

$$\begin{aligned} P_0(x) &= 1, \quad P_1(x) = x, \\ P_{n+1}(x) &= 2xP_n(x) - P_{n-1}(x) \quad \text{при } n > 0. \end{aligned} \quad (3)$$

Полиномы Чебышева ортогональны на системе равноотстоящих точек:

$$\sum_{t=0}^N P_j(t) \cdot P_h(t) = \delta_{jh} \quad (j, h \leq N) \quad (4)$$

В этом случае искомая функция $\tau(f)$ может быть представлена в виде:

$$\tau(f) = \sum_{j=0}^M c_j P_j(f) \quad (5)$$

Коэффициенты c_j можно найти из условия:

$$\frac{\partial J(c_0, c_1, c_2 \dots c_k)}{\partial c_k} = 0, \quad (6)$$

где $J(c_0, c_1, c_2 \dots c_k) = \sum_{t=0}^N \left[\tau_D(f_t) - \sum_{j=0}^M c_j P_j(f_t) \right]^2,$

$(f_t, \tau_D(f_t))$ – точки дискретной модели.

Получим:

$$c_j = \sum_{t=0}^N \tau_D(f_t) \cdot P_j(f_t) \quad (7)$$

Применение полиномов Чебышева позволяет решить важную задачу: определить оптимальную степень M аппроксимирующего полинома на основе критерия Гаусса [4]. Для этого необходимо вычислить эмпирическую дисперсию σ_M^2 :

$$\sigma_M^2 = \frac{J(c_0, c_1, c_2 \dots c_k)}{N - M} \quad (8)$$

Тогда, согласно критерию Гаусса, будем постепенно увеличивать M , вычисляя новые коэффициенты (7) и добавляя новые слагаемые в разложение (5). При этом величина σ_M^2 уменьшается и, достигнув минимума при некотором M , затем начинает возрастать; этот минимум σ_M^2 и дает оптимальное значение M .

Решение, полученное в виде (5), путем перегруппировки слагаемых можно преобразовать к виду:

$$\tau(f) = \sum_{k=0}^n \alpha_k \cdot f^n, \quad \text{где } n \in N \quad (9)$$

Преобразуем в выражении (1) слагаемое $2\pi \cdot \int_{f_p}^{f_p + \Delta f} \tau_j(f) df$, интегрируя $\tau_j(f)$ в виде

(9). В результате всех преобразований получим выражение для определения АЧХ многолучевой КВ радиолнии:

$$\left| H(f_p, t) \right| = \left| \sum_{j=1}^m H_j(f_p) \exp \left[-i \left(\varphi_j(f_p, t_0) + 2\pi \cdot \sum_{k=0}^n \frac{a_{jk}}{k+1} \left((f_p + \Delta f)^{k+1} - f_p^{k+1} \right) + 2\pi \cdot F_{Д} \cdot \Delta t \right) \right] \right| \quad (10)$$

Полученное выражение (10) позволяет восстановить АЧХ многолучевой КВ радиолнии с высоким частотным разрешением.

Литература

1. Иванов, В.А. Основы радиотехнических систем ДКМ диапазона: учебное пособие / В.А. Иванов, Н.В. Рябова, В.В. Шумаев – Йошкар-Ола: МарГТУ, 1998. – 204 с.
2. Щирый, А.О. Методика и результаты исследования АЧХ многолучевой ионосферной КВ-радиолнии с использованием ЛЧМ-иозонда / А.О. Щирый // Труды 5-ой Сессии молодых учёных "Гелио- и геофизические исследования" Байкальской молодежной научной школы по фундаментальной физике. – Иркутск: ИСЗФ СО РАН, 2002. – С.88-90.
3. Бахвалов, Н.С. Численные методы / Н.С. Бахвалов – М.: Наука, 1975. – 632 с.
4. Вайнштейн, Л.А. Разделение частот в теории колебаний и волн / Л.А. Вайнштейн, Д.Е. Вакман – М.: Наука, 1983. – 288 с.

СОДЕРЖАНИЕ

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И НЕЙРОННЫЕ СЕТИ	3
Безобразов С.В., Рубанов В.С. Алгоритмы искусственных иммунных систем и нейронных сетей для обнаружения вредоносных программ	3
Войцехович Л.Ю. Мультиагентный подход к обеспечению безопасности компьютерных сетей	6
Иванюк Д.С., Головкин В.А., Шуть В.Н. Применение последовательного нейроконтроллера в АСУТП	9
Кабыш А.С. Коллективное поведение в многоагентных системах на основе подкрепляющего обучения	13
Комар М.П. Использование искусственных иммунных систем и нейронных сетей для обнаружения компьютерных атак	16
Кочурко П.А. Настройка порогов нейросетевых детекторов для распознавания классов сетевых атак	18
Лаврентьева С.В. Оптимизация нейросетевой системы для анализа электроэнцефалограмм	20
Масалитина Н.Н. Оценка антикризисной устойчивости промышленного предприятия на основе нейросетевого моделирования	24
Новик Е.Ю., Семенюк Д.И. Применение нейронных сетей в экономике	27
Перетрухина В.В., Лобик В.Г., Путицкая И.И. Применение нейронных сетей для анализа стабильности экономической среды	29
Шевеленков В.В. Нейросетевой подход к архивации файлов	31
РАСПОЗНАВАНИЕ ОБРАЗОВ И АНАЛИЗ ИЗОБРАЖЕНИЙ	35
Войцехович Г. Ю. Идентификация человека по фотопортрету с применением нейросетевых методов	35
Дунец И. П. Принципы работы системы детектирования автономного мобильного робота и препятствий для его движения	38
Касьяник В.В., Шуть В.Н. Применение алгоритмов цифровой обработки динамических изображений для детектирования транспортных средств и управления дорожным движением	41
Корчевская Е.А., Мироненко В.М., Жигалко О.С. Автоматизированная система распознавания паразитологических объектов	44
Краснобаев Е. А. Сопровождение движущихся объектов методом корреляционного анализа изображений в системах компьютерного зрения	46
Кузьмицкий Н.Н. Организация эффективной обработки цифровых изображений средствами .NET-платформы	48
Лисица Е.В. Анализ и распознавание изображений костей кисти	50

АНАЛИЗ И МОДЕЛИРОВАНИЕ НЕЛИНЕЙНЫХ ДИНАМИЧЕСКИХ ПРОЦЕССОВ	53
Anatole D. Ruslanov, Anton V. Bashylau Algorithmic simulation in studying the kinetics of lipid peroxidation	53
Бондарь С.Г. Подход к численному анализу элементов РЭА	57
Воронов В.П., Муравьев Г.Л. Особенности организации моделирования стохастических сетей в многозадачных средах	59
Дмитриева А.В. Методика пересчета прогнозных значений снеговой нагрузки для различных периодов повторяемости	62
Емельянченко Н.С. Построение математической модели автоматизированного технологического процесса	65
Лысюк А.Н. Объектно-реляционная модель данных	68
Миколенко Е. А. Тестирование данных для анализа факторов, влияющих на динамику ВВП	71
Мороз А.Н. Разработка адаптивного многогипотезного алгоритма комплексной вторичной обработки информации в инерциально-спутниковой навигационной системе	74
Пашкевич А.А. Использование системы распределения памяти на примере выделения памяти под массив псевдослучайных чисел средствами ООП	77
Печко Е.В. Гибридный подход к моделированию управления социально-экономическим развитием региона	80
Сазончик А.А. Тестирование данных биржевых сводок IT компаний	83
Сашко А.Н., Теленкевич Р.С. Исследование влияния сдвига рабочего графика предприятий города на основные характеристики автотранспортных потоков	86
Семёнов Г.А. Модель корреляционного детектора нестационарных состояний динамической системы	90
Сидорович О.В. Упрощенный метод статистического анализа систем с переменной структурой	92
Согоян А.Л., Кочурко В.А. Подход к организации системы моделирования	95
Филипенко Е.В. Математическое моделирование свойств сырой песчано-глинистой формовочной смеси	97
Фурманова А.В. Применение графо-математического метода «Дерево принятия решений»	100
Шандриков А.С. Учёт конфигурации принципиальных электрических схем в процессе построения их математических моделей	103
Шелест Т.А. Построение прогнозных математических моделей максимальных расходов воды дождевых паводков рек Белорусского Полесья	107
Шешко Н. Н. Алгоритм восстановления временных рядов уровней грунтовых вод	110
Щеглов И.А. Методы дискретизации сложных пространственных областей	113
Щирий А.О. Переход от непрерывной модели АЧХ многолучевой ионосферной КВ радиоперелинии к дискретной модели	115

Научное издание

СОВРЕМЕННЫЕ ПРОБЛЕМЫ МАТЕМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

Материалы VI Республиканской научной конференции
молодых ученых и студентов

26–28 ноября 2009 года

Часть I

Ответственный за выпуск: **Махнист Л.П.**

Редактор: **Строкач Т.В.**

Компьютерная вёрстка: **Боровикова Е.А.**

Корректор: **Никитчик Е.В.**

Лицензия № 02330/0549435 от 8.04.2009 г.

Подписано к печати 9.11.2009 г.

Формат 60x84 1/16. Бумага «Снегурочка».

Гарнитура Arial Narrow. Усл. печ. л. 7,0. Уч.-изд. л. 7,5.

Заказ N 1014. Тираж 76 экз.

Отпечатано на ризографе учреждения образования
«Брестский государственный технический университет».

224017, г. Брест, ул. Московская, 267.