

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

СОВРЕМЕННЫЕ ПРОБЛЕМЫ МАТЕМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

**Материалы VII республиканской научной конференции
молодых ученых и студентов
24–26 ноября 2011 года**

Часть I

Брест 2011

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. Головкин, БрГТУ, д.т.н., профессор – зам. главного редактора
Р.Х. Садыков, БГУИР, д.т.н., профессор
Н.В. Лазакович, БГУ, д.ф.-м.н., профессор
А.И. Калинин, БГУ, д.ф.-м.н., профессор
В.В. Голенков, БГУИР, д.т.н., профессор
В.В. Старовойтов, ОИПИ НАНБ, д.т.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
Г.Л. Муравьев, 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 VII Республиканской научной конференции молодых ученых и студентов, Брест, 24–26 ноября 2011 г.: в 2-х частях / Брестский государственный технический университет; под ред. Рубанова [и др.] – Брест: изд-во БрГТУ, 2011. – Ч. 1. – 88 с.

ISBN 978-985-493-205-7 – I ч.

ISBN 978-985-493-204-0 (общий)

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 978-985-493-204-0 (общий)
ISBN 978-985-493-205-7 – I ч.

© Издательство БрГТУ, 2011

ИНТЕЛЛЕКТУАЛЬНЫЕ ТЕХНОЛОГИИ ОБРАБОТКИ ДАННЫХ. СОВРЕМЕННЫЕ ПРОБЛЕМЫ РОБОТОТЕХНИКИ

УДК 004.021

МЕТОДЫ ГЕНЕРИРОВАНИЯ ДОВЕРЕННОЙ ЦИФРОВОЙ ПОДПИСИ

Бурич А.Ю.

*УО «Белорусский государственный университет информатики
и радиоэлектроники», г. Минск
Научный руководитель – Ярмолик С.В., к.т.н.*

Введение

Доверенная цифровая подпись является модификацией цифровой подписи, которая отвечает специфичным требованиям. В схемах доверенной подписи один пользователь, называемый оригинальным подписывающим лицом, может делегировать права и возможности подписи документа другому пользователю, называемому доверенным подписывающим.

В данной работе приведены ключевые понятия, использующиеся в схемах доверенной подписи для последующей реализации и сравнения результатов, полученных теоретически и практически.

Основная часть

При реализации схемы доверенной подписи необходимо обеспечить следующие характеристики:

- различимость. Любой желающий проверить подпись должен иметь возможность определить, что подпись сделана доверенным лицом, а не оригинальным подписывающим;
- невозможность подделывания. Возможность генерирования действительной цифровой подписи должна быть только у пользователя, делегировавшего полномочия, и у его доверенного лица;
- проверяемость. Проверяющее лицо по доверенной подписи может убедиться, что подписывающий ознакомлен и согласен с содержанием документа;
- идентифицируемость. Возможность определения подписавшее лицо по подписи;
- неотрицаемость. Доверенное лицо, подписавшее документ, не может оспорить факта подписи документа.

Однако при реализации определенной схемы доверенной подписи допускается некоторое отхождение от описанных характеристик, в зависимости от того, какой тип делегирования был выбран. Для различных схем используются следующие типы делегирования:

- Полное делегирование. Оригинальный подписывающий передает доверенному лицу свой секретный ключ. В таком случае различия между доверенным и оригинальным подписывающими не возникает.
- Частичное делегирование. Оригинальный подписывающий не передает доверенному лицу секретный ключ. Вместо этого, он из секретного ключа получает доверенный ключ и передает доверенному лицу. Доверенное лицо на основании доверенного ключа генерирует подпись.
- Делегирование по доверенности. Доверенному лицу выдается доверенность, содержащая некоторую информацию о доверенном подписывающем, на основании которой доверенное лицо генерирует ключ и подписывает документ.

- Частичное делегирование по доверенности. Доверенному лицу выдается ключ, сгенерированный на основании секретного ключа оригинального подписывающего и доверенности.

Процесс подписи документа с использованием схем доверенной подписи зачастую состоит из четырех этапов:

- настройка системы. Данный этап заключается в установке публичных или частных параметров пользователей;
- генерирование подписи. На данном этапе генерируется подпись для подписи документа;
- подпись документа. Непосредственно встраивание сгенерированной подписи в документ;
- проверка. На данном этапе происходит проверка подлинности подписи, подтверждение доверенности подписи и пр.

Заключение

В настоящее время существует достаточное количество реализаций схем доверенной подписи, которые отличаются типами делегирования, алгоритмами генерирования и подписи документов. Каждая реализация может использоваться в зависимости конкретных потребностей пользователей в быстродействии или криптографической стойкости, а также могут быть предложены новые модификации.

Список цитированных источников

1. Proxy signatures: Delegation of the power to sign messages/M.Mambo, K.Usuda, E.Okamoto. IEICE Transactions Fundamentals. – Vol. E79A, 1997.
2. Designated-Verifier Proxy Signature Schemes / G. Wang. Security and Privacy in the Age of Ubiquitous Computing (IFIP/SEC 2005). – Springer, 2005.

УДК 004.8.032.26

СИСТЕМА УПРАВЛЕНИЯ ТРАНСПОРТОМ В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ

Войцехович О.Ю.

*УО «Брестский государственный технический университет», г. Брест
Научный руководитель – Шуть В.Н., доцент, к.т.н.*

1. Постановка задачи

Задача состоит в разработке адаптивной системы управления транспортом, работающей в режиме реального времени вдоль городской магистрали, способной координировать светофоры для улучшения дорожной ситуации в целом. Еще одна задача состоит в моделировании, тестировании и оценке разработанной системы.

Для решения поставленной задачи предложен подход, координирующий время горения сигналов светофоров путем распознавания и предсказания движения групп транспортных средств (пачек) на магистрали и прилегающих к ней улицах с помощью полученных с детекторов и отфильтрованных данных. Для тестирования разработанного подхода в управлении транспортом была реализована имитационная модель.

2. Описание системы

Адаптивная система управления состоит из 3 частей: предсказание прибытий и очередей (обрабатывает данные с детектора и осуществляет предсказание); система принятия

решений (строит дерево решений и выбирает оптимальные времена и длительности горения зеленого и красного сигналов для магистрали и прилегающих дорог); продвижение (модифицирует массив, где хранятся данные о распознанных пачках транспортных средств).

Разрабатываемая система управления работает на уровне пачек автомобилей и их скоростей. Критерием оптимизации являются средние задержки, которые необходимо свести к минимуму. Система предсказывает транспортный поток (предсказание осуществляется в пространстве и времени), чтобы осуществить упреждающее управление. С помощью построения бинарного дерева решений выбираются оптимальные настройки светофоров, которые отвечают сделанным предсказаниям.

Большинство существующих подходов, управляющих транспортным потоком, используют статистические сглаженные данные. Такие системы основаны на временных планах светофоров, оперирующих временем цикла, расколами и смещением. Такой подход пригоден для медленно меняющихся характеристик, но не подходит, если рассматривать реальные колебания транспортного потока, которые статистические подсчеты не могут учесть.

В разрабатываемой системе акцент смещается от модификации временных параметров, реагирующих на уже случившиеся изменения транспортного потока, к упреждающей настройке параметров светофора для предсказываемого состояния транспортного потока. И это ее главное преимущество, которое делает систему гибкой. То есть мы не устанавливаем временные планы в терминах времени цикла, расколов и сдвигов фаз. А, скорее, в терминах длительности и последовательности фаз.

Система нуждается: 1) в обмене данными в режиме реального времени с процессором; 2) в вычислительных возможностях на уровне РС, 3) во входной информации о характеристиках транспортного потока, считываемой с датчиков в реальном времени. Система централизованная, т.к. данные со всех датчиков собираются в центр управления, где происходит прогнозирование и выбор оптимальных фаз.

Для предсказания необходимы следующие входные данные: 1) время проезда от детектора к детектору; 2) коэффициент очистки очереди и 3) доля сворачиваемого транспорта. Выходные данные используются алгоритмом управления.

За основу был взят алгоритм управления, предложенный P. Dell'Olmo и P.V. Mirchandani [1]. Если прогнозируется, что две или более пачки подъедут к перекрестку и создадут конкурирующий спрос на время горения зеленого сигнала светофора для конфликтных направлений, тогда должно быть определено, какому направлению движения отдать время горения зеленого сигнала. Решение что сделать зависит от полученного значения выбранного критерия эффективности. Оптимальное разрешение конфликтов в реальном времени, или иными словами, оптимизация движения распознанных пачек автомобилей – это основная цель алгоритма управления.

Для разрешения конфликтных ситуаций алгоритм заблаговременно строит дерево решений. Каждый возникающий конфликт формирует узел в дереве решений; типы решений в этом узле включают: а) дать зеленое время пачке А, т.е. остановить пачку Б (пачка А подъезжает раньше); б) расколоть пачку А (т.е. зеленое время предоставить пачке Б). Каждая ветка дерева рассматривается далее, чтобы сохранить путь от начального узла к потенциальному решению. Построение дерева заканчивается, когда разобраны все конфликты. Конечные узлы будут представлять собой полную стоимость всех решений, которые идут от корня к конечному узлу дерева решений. Выбор единственного решения с минимальной задержкой, дает конечную стоимость траектории решения конфликтов. Путь по дереву от корня к выбранному листу обеспечивает фазовый план – конечная цель всего алгоритма.

3. Вопросы реализации

В системе был использован алгоритм идентификации пачек, основанный на двух пороговых параметрах: максимальное расстояние между двумя автомобилями в пачке и минимальное число автомобилей, которые составляют пачку.

Алгоритм начинает работу с начального решения о распределении фаз, которое может быть получено с использованием статистических данных и определяет первый узел в дереве решений. Критерий эффективности, связанный с начальными фазами, становится верхней границей при выполнении алгоритма.

4. Моделирование

Для тестирования системы была создана микроскопическая стохастическая имитационная модель, и в настоящий момент происходит ее кооперация и синхронизация с системой управления. Генерация автомобилей рассматривается как неоднородный пуассоновский процесс. Количество прибытий автомобилей следует распределению Пуассона с параметром λ , где λ – среднее количество прибытий в единицу времени. Моделирование транспортных потоков было выполнено с помощью клеточного автомата (КА) [2]. КА – это модели, которые являются дискретными в пространстве, времени и переменных состояния. Из-за дискретности КА являются чрезвычайно эффективными в реализации на компьютере. Самый простой набор правил, который приводит к реалистичному поведению, был введен в 1992 году учеными Nagel и Schreckenberg [3]. Он состоит из 4 шагов, которые должны применяться одновременно для всех автомобилей (параллельно или синхронно). Вышеприведенный набор правил является минимальным в том смысле, что отсутствие одного из 4 шагов будет вести к не реалистичному поведению.

Шаг 1: разгон. Все машины, не достигшие максимальной скорости v_{\max} , ускоряются на одну единицу: $v \rightarrow v+1$ **Шаг2: безопасная дистанция.** Если у машины есть d пустых ячеек перед собой и ее скорость v (после шага 1) больше, чем d , то она уменьшает скорость до d : $v \rightarrow \min\{d, v\}$ **Шаг 3: эффект случайности.** С вероятностью p транспортное средство уменьшает скорость на одну единицу (если v после шага 2): $v \rightarrow v-1$ **Шаг 4: езда.** После шагов 1-3 новая скорость v_n для каждой машины n определяет продвижение на v_n ячеек: $X_n \rightarrow X_n + v_n$.

Заключение

Была создана адаптивная система управления транспортным потоком, способная координировать сигналы светофоров, с целью достичь минимальной средней задержки и снизить необходимость постоянного наблюдения за перекрестком и настройки светофоров. Целью алгоритма управления является гибкое реагирование на стохастическое поведение транспортного потока. В данный момент проводится тестирование системы управления с помощью имитационной модели.

Список цитированных источников

1. Dell'Olmo, P. REALBAND: An Approach for Real-Time Coordination of Traffic Flows on a Network / P. Dell'Olmo, P.B. Mirchandani // Transportation Research Record. – 1995. – 1494. – P. 106-116.
2. Chowdhury, D. Statistical physics of vehicular traffic and some related systems / D. Chowdhury, L. Santen, A. Schadschneider // Physics Reports. – 2000. – 329. – P. 199.
3. Nagel, K. A cellular automaton model for freeway traffic / K. Nagel, M. Schreckenberg // J. Physique. – 1992. – 2. – P. 2221.

УДК 004.8

НЕЙРОСЕТЕВОЙ АНАЛИЗ СИГНАЛОВ ЭЭГ

Давидюк Ю.И., Савицкий Ю.В.

УО «Брестский государственный технический университет», г. Брест

Нейросетевые методы анализа хаотических сигналов находят все большее применение в различных областях благодаря ряду преимуществ по сравнению с традиционными методами: возможностью исследования систем, математическая модель которых неизвестна (неизвестны математические соотношения, характеризующие поведение динамической системы); использованием для исследований выборки данных ограниченного объема [1]. Высокая актуальность данного направления объясняется всё возрастающей потребностью в наличии эффективных средств для решения сложных нетривиальных задач в плохо формализуемых областях обработки информации.

Хаос в динамике означает чувствительность динамической эволюции к изменениям начальных условий. Старший показатель Ляпунова характеризует степень экспоненциального расхождения близких траекторий. Наличие у системы положительной экспоненты Ляпунова свидетельствует о том, что любые две близкие траектории быстро расходятся с течением времени, то есть имеет место чувствительность к значениям начальных условий.

В результате экспериментов установлено, что наиболее приемлемой для цели данного исследования является модель гетерогенной многослойной нейронной сети (НС) с нейронами сигмоидального типа в скрытом слое и линейными нейронами выходного слоя сети.

Для обучения НС применяется алгоритм обратного распространения ошибки (и его более быстродействующие модификации), использующий метод градиентного спуска для минимизации функции среднеквадратичной погрешности. Благодаря высокой точности алгоритм позволяет достигать малой погрешности обучения, что является крайне важным фактором для решения большинства практических задач в нейросетевом базисе.

В общем виде разработанный алгоритм состоит из следующих этапов: 1) нормализация исходного временного ряда, состоящего из N точек, выбранных с учетом задержки τ ; 2) сегментация исходного временного ряда методом фиксированных отрезков; 3) обучение нейронной сети прогнозированию по методу скользящего окна; 4) расчет старшего показателя Ляпунова на базе сформированной нейросетевой прогнозной модели по методу отклонений траекторий прогнозов.

Существует проблема в выборе метода сегментации исходной выборки [2]. Для сегментации исходной выборки сигнала ЭЭГ применяются следующие методы: а) метод фиксированных отрезков; б) метод наложения отрезков друг на друга; в) адаптивный метод при помощи нейронных сетей.

Наиболее приемлемым для решения нашей задачи является метод фиксированных отрезков.

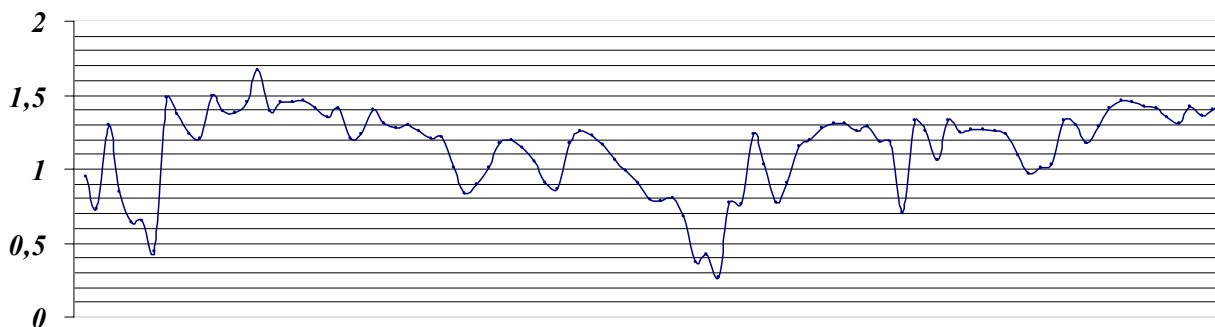
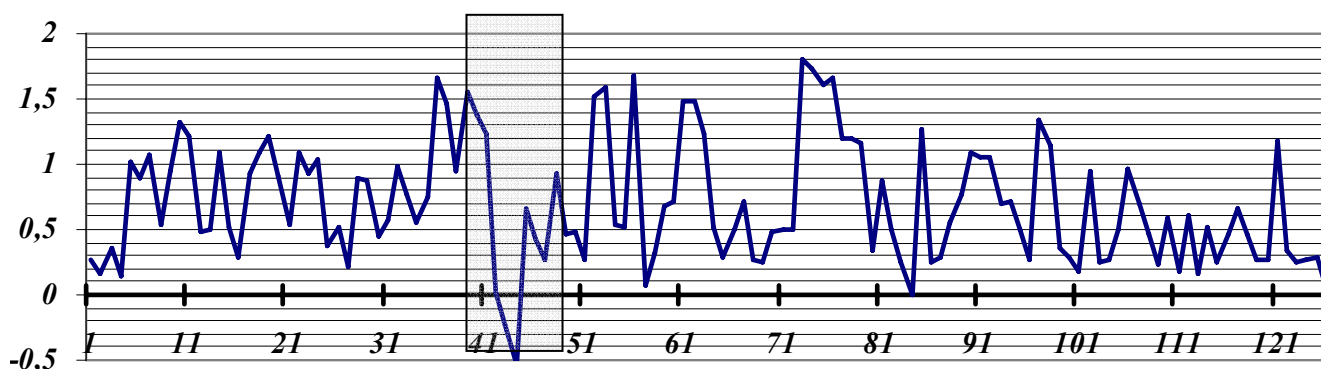
Для тестирования программы были проанализированы наборы сигналов ЭЭГ человека (A,D,E) [3]. Каждый набор содержит в себе 100 сигналов определенной группы (в зависимости с эпилептической активностью). Каждый сигнал имеет 4096 амплитудных точек. Результаты анализа сведем в таблицу 1.

Таблица 1– Результаты расчета показателя Ляпунова

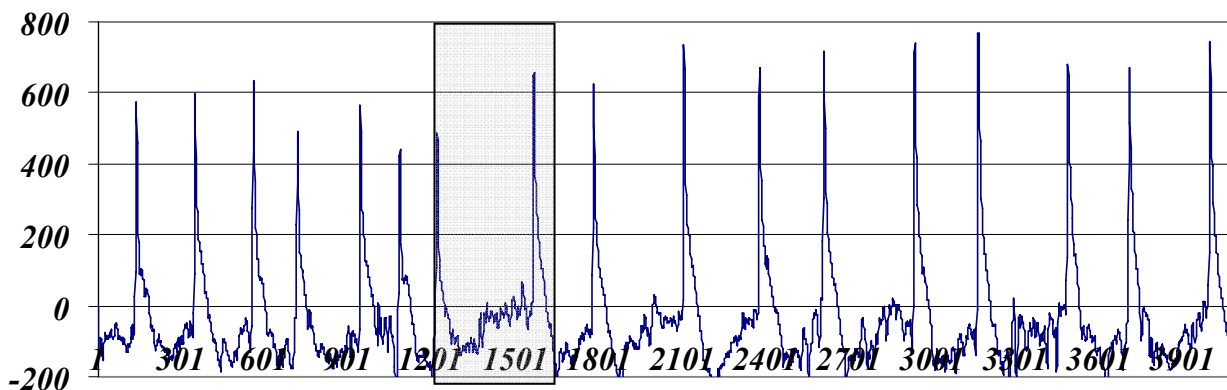
Набор сигналов	Значение показателя Ляпунова		
	максимум	минимум	среднее
A	1,67012	0,015406	1,072971
D	0,655647	-1,1907	-0,03615
E	1,84311	-1,08277	0,703402

В ходе анализа было выявлено, что группа сигналов D (рис. 2) и E является ЭЭГ с эпилептической активностью, группа сигналов A (рис. 1) – ЭЭГ здорового человека.

Секторы эпилептической активности на ЭЭГ выделены на рис.2-4 серым цветом.

**Рисунок 1– Группа сигналов A****Рисунок 2 – Группа сигналов D**

На рис.3 приведен один сигнал ЭЭГ человека с эпилептической активностью. Данный аномальный сегмент мы получили с помощью вычисления старшего показателя Ляпунова на каждом сегменте, включающем в себя 30 точек (рис.2).

**Рисунок 3 – Сигнал с аномальной активностью из группы D**

Таким образом, применение разработанного алгоритма и программных средств показало потенциальные возможности эффективного распознавания эпилептической активности мозга. Следует отметить, что результаты вычислительных экспериментов с группами сигналов А, D, Е в достаточной степени коррелируют с полученными ранее результатами, опубликованными другими авторами [2].

Список цитированных источников

1. Golovko, V. Neural Networks for Signal Processing in Measurement Analysis and Industrial Applications: the Case of Chaotic Signal Processing / V. Golovko, Y. Savitsky, N. Maniakov // chapter of NATO book "Neural networks for instrumentation, measurement and related industrial applications". - Amsterdam: IOS Press, 2003. – P. 119-143.
2. Bezobrazova, S. Neural-network segmentation of electroencephalogram signal for epileptiform activity detection / S. Bezobrazova, V. Golovko // Computing. – 2008. – Vol. 7, Issue 3 – P. 30–37.
3. Временные сигналы ЭЭГ. – Режим доступа: <http://www.meb.uni-bonn.de/epileptologie/science/physik/eegdata.html>

УДК 004.942

ОБ ОДНОМ МЕТОДЕ ПОСТРОЕНИЯ КОМПЬЮТЕРНОЙ СИСТЕМЫ ДИАГНОСТИКИ НА ОСНОВЕ НАБЛЮДАЕМЫХ ДАННЫХ

Жукевич А.И.

УО «Гродненский государственный университет имени Янки Купалы», г. Гродно,
Научный руководитель – В.Г. Родченко, к.т.н., доцент

Введение

При проведении научных исследований и решений целого ряда прикладных задач, связанных с анализом многомерных объектов сложной природы, высокую эффективность продемонстрировали подходы, базирующие на использовании методов математической теории распознавания образов [1]. Процесс распознавания реализуется через выполнение двух основных процедур, первая из которых ориентирована на *обучение*, а вторая – непосредственно на *распознавание*. Если процедуру обучения удастся реализовать эффективно, то и выполнение второй принципиальных затруднений не вызывает. В практических задачах именно процесс обучения является наиболее трудоемким с точки зрения реализации, поскольку в реальных системах исследуются объекты, которые характеризуются большим количеством разнообразных признаков, имеющих сложную природу и распределенных по разным законам [2].

Процедура обучения осуществляется путем анализа данных, которые предварительно формируются в матричном виде и представляют собой классифицированную обучающую выборку. Указанная обучающая выборка формируется в результате выполнения подготовительной относительно простой процедуры первичной обработки наблюдаемых данных на основе использования априорного словаря признаков.

Задачи, связанные с определением алфавита классов, набора наблюдаемых данных и построением априорного словаря признаков носят проблемно-ориентированный характер и решаются путем привлечения экспертов в данной области.

При построении реальных систем диагностики часто оказывается, что только относительно небольшое число признаков, из первоначально включенных в априорный словарь, представляют интерес для качественного выполнения заключительной процедуры принятия решения [3].

В данной работе представлен метод построения компьютерной системы диагностики, который предусматривает автоматическую процедуру построения пространства решений, в которой в кластерном виде формируются компактные эталоны, представляющие собой формальные образы классов, для последующей процедуры распознавания.

Описание структуры компьютерной системы диагностики

Структурно компьютерная система диагностики включает в себя три функциональные части: 1) предварительной обработки и преобразования данных; 2) анализа исходного набора признаков и сепарирования их по степени информативности; 3) реализации непосредственно процедуры распознавания.

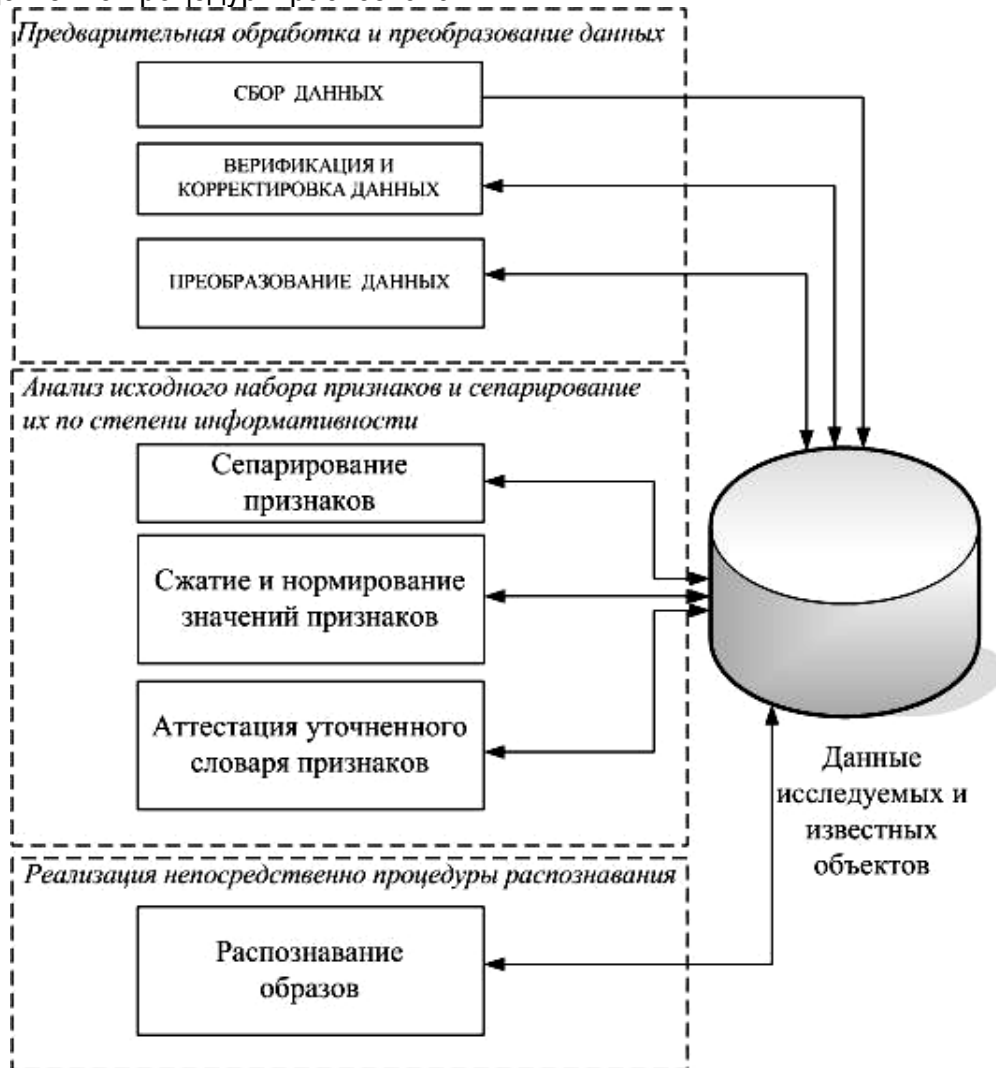


Рисунок 1 – Структурная схема компьютерной системы диагностики

Первая часть программного комплекса является проблемно-ориентированной и состоит из программ, которые, во-первых, позволяют осуществить ввод и сбор наблюдаемых данных, произвести верификацию введенной информации и провести при необходимости корректировку; во-вторых, реализовать необходимые операции по преобразованию исходных данных к матричному виду типа “объект-свойство” в универсальный формат представления. Все данные, необходимые для работы системы, сохраняются в базе данных. Процесс автоматизированного сбора информации может предусматривать применение как программных, так и аппаратных средств.

Вторая часть системы предназначена для формирования базы знаний, состоящая из компактных эталонов, реализованных в кластерном виде и представляющие собой фор-

мальные образы классов, для последующей процедуры распознавания. С этой целью система предусматривает сепарирование признаков из исходного априорного словаря на три вида по степени их информативности, сжатие описания исследуемых объектов и построение уточненного словаря признаков, нормировка значений к единичному интервалу, аттестацию возможности использования построенного уточненного словаря признаков для анализа данных и в итоге формирование классифицированной выборки эталонных значений в признаковом пространстве решений. Затем строятся эталоны-кластеры классов, и процедура обучения системы распознавания завершается.

Третья часть системы, в которой реализован алгоритм метода автоматизации анализа данных, предусматривает выполнение непосредственно процедуры распознавания исследуемого объекта.

Заключение

Разработан метод построения компьютерной системы диагностики, который базируется на использовании методов теории распознавания образов и кластерного анализа. Для качественной реализации процедуры распознавания предусматривается обязательное выполнение процедуры обучения, которая осуществляется на основе анализа данных классифицированной обучающей выборки. Признаки из исходного априорного словаря сепарируются по степени информативности с точки зрения разделения эталонов исследуемых классов в соответствующем многомерном признаковом пространстве решений.

Метод построения системы предусматривает автоматизацию процедур обучения и распознавания исследуемых объектов. Он характеризуется универсальностью и позволяет исследовать объекты на основе анализа различных по своей природе исходных признаков.

Список цитированных источников

1. Загоруйко, Н.Г. Прикладные методы анализа данных и знаний / Н.Г. Загоруйко. – Новосибирск: Изд-во Института математики СО РАН, 1999. – 264 с.
2. Васильев, В.И. Проблема обучения распознаванию образов / В.И. Васильев – К., 1989.
3. Гуца, Ю.В. Об одном методе автоматического построения пространства решений при реализации компьютерных систем диагностики / Ю.В. Гуца, А.И. Жукевич, Е.В. Олизарович, В.Г. Родченко // Научные исследования преподавателей факультета математики и информатики: сб. науч. ст. / ГрГУ им.Я.Купалы; редкол.: И.П. Мартынов (отв. ред) [и др.]. – Гродно: ГрГУ, 2010. -61-64 с.

УДК 004.896

ОЦЕНКА ПОГРЕШНОСТИ ОДОМЕТРОВ МОБИЛЬНОГО РОБОТА С ПОМОЩЬЮ НЕЙРОННЫХ СЕТЕЙ

Касьяник В.В., Дунец А.П.

УО «Брестский государственный технический университет», г. Брест

Постановка задачи

Одной из важнейших задач в робототехнике является задача определения точного местоположения робота – проблема локализации или позиционирования. Данная проблема очень важна, так как информация о точном местоположении робота необходима для решения более сложных и комплексных задач навигации, построения пути и построения карты окружающей среды. На сегодняшний день существует несколько различных подходов к решению проблемы локализации. Эти подходы используют различные сенсоры и алгоритмы обработки данных. Так, одним из подходов к локализации является

вероятностный подход на основе SLAM-методик с использованием лазерного сканера или дальномеров [1]. Метод оценки ошибки одометров на основе данных навигации мобильного робота представлен в [2]. Классической в данной области является работа [3], где разработана методика калибровки и коррекции погрешности одометров, предложены методы проведения эксперимента для оценки различных факторов, влияющих на погрешность. В работе [4] выполнен сравнительный анализ нескольких различных подходов к оценке погрешности одометров, одним из которых является метод нейронных сетей. В работе применен многослойный персептрон для оценки ошибки одометров, который показал лучшие результаты из рассмотренных методов. Этим обусловлена актуальность исследований в области нейросетевых технологий оценки погрешностей сенсоров позиционирования.

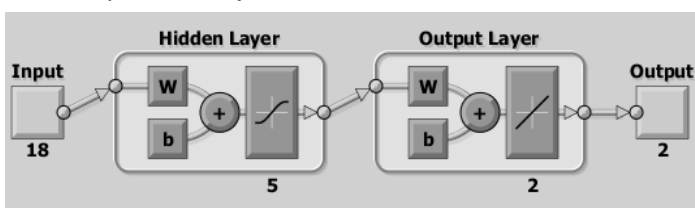
Данная работа является продолжением работы [4] в области применения нейронных сетей для мобильных роботов малого размера. Малый размер робота является причиной более существенных погрешностей и ограничивает использование дополнительных сенсоров. Таким образом, необходимо провести исследование эффективности применения нейронных сетей в данных условиях. В данной работе также предлагается использовать искусственную нейронную сеть, которая будет оценивать изменение ошибки с течением времени. Так как погрешность одометров зависит от множества различных случайных и систематических факторов, то изменение погрешности является нелинейной динамической системой.

Для оценки поведения такой системы и последующей коррекции позиции робота был применен двухслойный персептрон с тангенциально-сигмоидной функцией активации в скрытом слое и линейными выходными нейронами. Для обучения использовался метод обратного распространения ошибки. На вход нейронной сети подавались образы из 18 элементов:

$$\{x_1, y_1, \dots, x_8, y_8, t_b, t_e\},$$

где, x_1, y_1 – координаты робота по данным одометров в течение эксперимента,
 t_b, t_e – время старта и окончания эксперимента.

Количество нейронов скрытого слоя варьировалось в процессе экспериментов, выходной слой содержал 2 нейрона, которые определяли координату робота x и y в качестве оценки нейронной сети.



Архитектура нейронной сети представлена на рис. 1.

Рисунок 1 – Архитектура нейронной сети

Для обучения нейронной сети в качестве эталонных значений использовались координаты реальной позиции робота. Реальная позиция робота оценивалась с помощью двух подходов – видеорегистрации робота с помощью камеры, закрепленной на потолке, и ручного измерения. После обучения нейронной сети проводились эксперименты по оценке позиции робота на новых данных одометров. В следующем разделе приведены и проанализированы результаты работы нейронной сети в сравнении с реальной позицией робота.

Результаты экспериментов

На первом этапе экспериментов было произведено исследование характеристик механи-

ки мобильного робота. Малые габаритные размеры робота (12см X 12 см X 8 см), легкий вес (200 гр) приводят к существенному увеличению погрешностей курсового угла робота.

Для оценки эффективности применения нейронной сети в задаче уточнения данных одометров был проведен 21 эксперимент. В данном эксперименте робот двигался по прямой на расстояние 1,5 м. На вход нейронной сети подавалось время старта и финиша робота, 8 пар координат положения робота, рассчитанных на основании одометров. Эталонными значениями для обучения сети были данные о реальном положении робота – пара значений координат x и y .

В процессе обучения нейронной сети были произведены исследования влияния количества нейронов скрытого слоя на ошибку нейронной сети в задаче оценки позиции одометров. Результаты исследования приведены в таблице 1. Видно, что наиболее оптимальным количеством нейронов в скрытом слое является интервал 9-11 нейронов. Также можно отметить, что при увеличении количества нейронов до 15, нейронная сеть запоминает входные образы, а на новых данных показывает неудовлетворительные результаты.

Таблица 1 – Результаты оценки погрешности одометров

Эксперимент	Нейронов в скрытом слое	Среднеквадратичная ошибка на этапе обучения	Тестирование сети на новых данных
1	5	36,31	29,876
2	7	20,49	35,502
3	9	4,72	12,255
4	11	5,99	12,258
5	13	28,19	14,401
6	15	2,29	28,561

По этим результатам для дальнейших экспериментов была выбрана нейронная сеть с 11 нейронными элементами в скрытом слое. Далее было проведено исследование 21 эксперимента с помощью обученной нейронной сети.

На рис. 2 и рис. 3 представлены результаты оценки нейронной сетью ошибки одометров по оси X и Y соответственно относительно реальной позиции робота.



Рисунок 2 – Аппроксимация ошибки одометров нейронной сетью

Рисунок 3 – Аппроксимация ошибки одометров нейронной сетью



С каждым экспериментом систематическая ошибка

нарастает и ухудшается оценка нейронной сети. По координате Y заметное ухудшение про-

исходит после 14 эксперимента, что соответствует примерно 200 метрам пройденного расстояния. В итоге при оценке погрешности по оси X точность нейронной сети была выше на 8 %, по оси Y – на 50 %.

Выводы

В результате проведенного исследования были получены характеристики механики для конкретного реального робота. На основании данных характеристик проведена калибровка подсистем управления и позиционирования робота для решения задачи локализации. Для уточнения позиции робота и повышения качества и надежности информации, выдаваемой подсистемой локализации был предложен нейросетевой подход. Результаты экспериментов доказали эффективность предложенного подхода, особенно в продольном направлении, где ошибка наиболее существенна. Однако в данном подходе имеется ряд недостатков, таких как: необходимость предварительной настройки нейросетевого модуля для конкретного робота и окружающей среды, требование к производительности бортового оборудования робота. В дальнейшем для решения данных проблем планируется создание самообучающейся интеллектуальной системы позиционирования, которая смогла бы адаптироваться во время работы к параметрам робота, внешней среды и использовала для оценки позиции кроме одометров, другие сенсоры робота.

Благодарности

Данная работа выполнялась при поддержке гранта Ф11-ЛИТ003 Белорусского республиканского фонда фундаментальных исследований и гранта ГБ 11/117 Министерства образования Республики Беларусь.

Список цитированных источников

1. Montemerlo, M. Fast-SLAM2.0: An Improved Particle Filtering Algorithm for Simultaneous Localization and Mapping that Provably Converges, In Proc. Of the Int. Confs. On Artificial Intelligence (IJCAI) / M. Montemerlo, S. Thrun, D. Kollerand B. Wegbreit. – 2003. – P. 1151–1156.
2. Martinelli, A. Estimating the Odometry Error of a Mobile Robot during Navigation, In Procs. Of European Conf. on Mobile Robots / A.Martinelli, R.Siegwart. – 2003.
3. Borenstein, J. Measurement and correction of systematic odometry errors in mobile robot, IEEE Transactions on Robotics and Automation / J.Borenstein, L.Feng. – 1996. – P.12(6):869-880.
4. Haoming Xu and John James Collins. Estimating the Odometry Error of a Mobile Robot by Neural Networks, In Proc. of International Conference on Machine Learning and Applications, 2009.

УДК 621.397.13:004.932.7

ВЫБОР ПРИЗНАКОВ И МОДЕЛИ КЛАССИФИКАТОРА В ЗАДАЧЕ РАСПОЗНАВАНИЯ ИЗОБРАЖЕНИЙ РЕАЛЬНЫХ СЦЕН

Кузьмицкий Н.Н.

*УО «Брестский государственный технический университет», г. Брест
Научный руководитель – Дереченник С.С., к.т.н., доцент*

Введение

Классификация – одна из наиболее часто встречаемых научно-практических задач, необходимость повседневного решения которой следует из удобства иерархического восприятия действительности. Автоматизированная модель классификации основана на использовании функций определенных параметрических семейств, объединенных в решающем правиле, позволяющем по набору значений признаков объекта определить его класс.

Многие классификационные задачи имеют эффективное решение благодаря хоро-

шему набору признаков, однако в ряде ситуаций составить их исчерпывающий перечень достаточно проблематично, например, в случае классификации изображений реальных сцен. Причиной тому являются как внешние факторы (ракурс, освещение и др.), так и внутренние (изменчивость композиции, свойств объектов и др.), значительно осложняющие применение стандартных методик анализа изображений.

Кроме того, хорошие признаки являются лишь необходимым условием эффективной классификации, ввиду сложности "ручной" настройки пороговых величин, разделяющих классы. Поэтому целесообразным выглядит применение методов машинного обучения, однако выбор подходящего метода является нетривиальной задачей, зависящей от статистической природы признаков, наличия обучающей выборки, требований к качеству обобщения и др.

Постановка задачи

Целью описываемой исследовательской работы является создание классификатора изображений реальных сцен. При этом основное внимание уделяется двум задачам: формированию признаков изображений и обучению классификатора. Анализ известных методик решения первой задачи показал, что наиболее перспективным для рассматриваемого типа изображений является применение SIFT-детектора визуальных слов ввиду его инвариантности к различным искажениям. Для решения второй задачи был выбран метод опорных векторов (SVM, Support Vector Machine), являющийся гибким механизмом построения нелинейных разделяющих границ и имеющий высокую обобщающую способность даже для небольших обучающих выборок.

Схема построения классификатора представлена на рисунке 1.

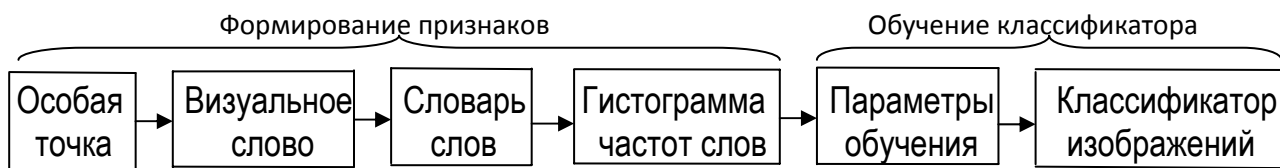


Рисунок 1 – Схема построения классификатора

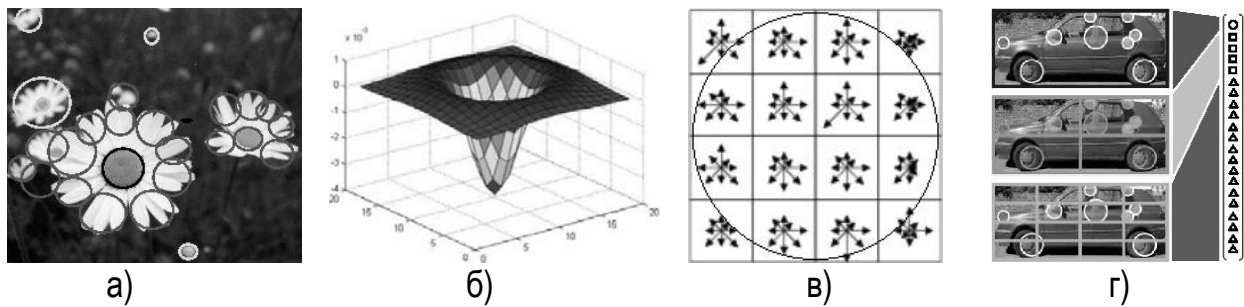
Формирование признаков

Визуальное слово – это фрагмент изображения, являющийся окрестностью особой точки. Часто встречаемые слова объединим в "словарь", проведем квантование остальных фрагментов по нему, тогда итоговым признаком изображения будет являться гистограмма частот слов [1].

В качестве особых будем рассматривать точки, имеющие окрестности с однородной яркостью (см. рисунок 2а), называемые *блобами* ("blob" – капля). *SIFT* (Scale-Invariant Feature Transform) – детектор блобов, применяющий для их обнаружения следующее правило: в центре блоба ЛОГ-фильтр (лапласиан гауссиана: $\Delta^2 g = \partial^2 g / \partial x^2 + \partial^2 g / \partial y^2$, где g – функция Гаусса, см. рисунок 2б) имеет локальный максимум, если размер фильтра и блоба совпадают. Таким образом, вычисляя свертки изображения с ЛОГ-фильтрами, можно локализовать блоб и определить его размер, при этом исключить из дальнейшего рассмотрения нестабильные блобы с низкой контрастностью [2].

Для построения дескриптора особой точки воспользуемся характеристиками градиента яркости пикселей ее окрестности, на основе которых сформируем вектор-дескриптор. Процесс его формирования состоит из следующих шагов:

1) найдем доминантное направление градиента и повернем окрестность так, чтобы оно стало строго вертикальным;



а) изображение с блобами, б) ЛОГ-фильтр; в) SIFT-дескриптор; г) пирамидальная гистограмма

Рисунок 2 – Детектирование особых точек

2) разобьем окрестность прямоугольной сеткой размера 4×4 и вычислим гистограммы направлений градиента в ячейках с шагом 45° , при этом значения градиента взвесим по гауссиане в центре окрестности (см. рисунок 2в);

3) объединим гистограммы в одном векторе длины $128 (4 \cdot 4 \cdot 8)$.

Таким образом, визуальное слово – это блок, обнаруженный SIFT-детектированием, описанный вектором фиксированной длины, обладающим инвариантностью относительно масштабирования, переносов, поворотов, изменений освещения и небольшим сдвигам.

Формирование словаря визуальных слов проведем на основе одного из методов кластеризации, например, K -средних, где K – предполагаемое количество центров. Тогда словарь составим из образованных кластеров, квантование слов проведем по мере близости до их центров, а итоговым признаком изображения будет гистограмма частот слов длины K . Чтобы учесть пространственную информацию, можно воспользоваться пирамидальными гистограммами: сетками разбить изображение на N ячеек, построить для каждой гистограмму и объединить их в вектор длины $K \cdot N$. (см. рисунок 2г).

Обучение классификатора

Для создания классификатора воспользуемся методом опорных векторов [3], который с помощью гиперплоскости осуществляет линейное разделение образов двух классов. Достоинством метода является работоспособность в случае линейно неразделимых классов, образы которых отображаются в пространство большей размерности, при этом решается задача квадратичного программирования, имеющая единственное решение. Основным параметром метода – мера регуляризации C , позволяющая находить компромисс между максимизацией расстояния от гиперплоскости до образов и минимизацией ошибок обучения. Для выбора величины C проводятся перекрестные проверки: классификатор, обученный на одном подмножестве обучающей выборки, тестируется на другом. В случае M -классовой задачи строится M бинарных классификаторов и используется голосование "один против всех" либо $M \cdot (M-1) / 2$ классификаторов с голосованием "один против одного".

Апробация классификатора

Представленная выше схема построения классификатора была реализована для распознавания изображений интерьеров в рамках конкурса Microsoft Computer Vision School 2011. Его целью было создание классификатора изображений 4-х классов сцен: жилых помещений, учебных корпусов МГУ; городских улиц, цветов. Размеры предложенной обучающей / тестовой выборки составили: 133 / 100, 260 / 100, 205 / 100, 210 / 100 экземпляров соответственно. Разработанный в системе MatLab классификатор позволил отнести к верному классу 93,5% (100%) изображений тестовой (обучающей) выборки (см. рисунок 3), что является вторым из лучших результатов (95,5% - победитель).



Рисунок 3 – Примеры четырех классов изображений тестовой выборки

Заключение

Рассмотренная методика построения классификатора изображений реальных сцен является достаточно общей и применима для аналогичных типовых задач. Ее преимущество – устойчивость выбранных признаков ко многим видам искажений и гибкость модели классификатора. Недостаток – необходимость кластеризации с заранее неизвестным числом центров и довольно длительное время обучения.

Список цитированных источников

1. Sivic, J. Video Google: A Text Retrieval Approach to Object Matching in Videos / J. Sivic, A. Zisserman // ICCV – 2003. – P. 1470-1477.
2. Lowe, David G. Distinctive image features from scale-invariant keypoints / David G. Lowe // IJCV 2004. – P. 91-110.
3. Vapnik, V. Statistical Learning Theory / V. Vapnik. – Wiley. – 1998. – P.768.

УДК 004.5

ИЗМЕРЕНИЕ ПРОСТРАНСТВЕННЫХ КООРДИНАТ ТЕСТОВОГО ОБЪЕКТА ПРИ ПОМОЩИ WEB-КАМЕР

Кулажевко С.В., Бобров Д.В., Петров П.В., Кольчевский Н.Н.
 УО «Белорусский государственный университет», г. Минск

Обычная web-камера с успехом может улучшить стандартные регистрирующие и измерительные приборы. Достоинства применения web-камеры [1]: использование обычного компьютера, лёгкая адаптация к условиям измерений и исследований, возможность быстрого обновления и расширения, совместимость с любыми измерительными устройствами, экономичность. Известным ограничением на пути использования компьютера в области измерений и регистрации аналоговых сигналов является то, что компьютер не способен принимать аналоговые данные, так как является полностью цифровым устройством. Для решения этой проблемы существуют специализированные устройства - Аналого-Цифровые Преобразователи (АЦП), которые осуществляют преобразование аналоговых сигналов в цифровую форму. В качестве АЦП в нашем случае используется web-камера, которая преобразует аналоговый сигнал от измерительного или регистрирующего прибора в цифровой вид, пригодный для приёма его компьютером. Программное обеспечение является инструментом, позволяющим использовать, обрабатывать данные, полученные от измерительных приборов.

Целью работы является разработка программного обеспечения для применения web-камеры в качестве бесконтактного измерительного или регистрирующего прибора.

Очень часто для достижения необходимого результата требуется долгое и рутинное наблюдение за объектом исследования в связи с медленно изменяющимися его характеристиками. Для упрощения наблюдения была написана программа Time Catcher,

предназначенная для получения изображения тестового объекта в заданном промежутке времени. Программа была написана на языке программирования Delphi. В ходе тестирования программы выявились некоторые особенности программного пакета:

-При длительной работе программы, и большой частоте получения кадров мы получаем большое количество снимков, соответственно жёсткий диск компьютера должен иметь необходимое количество свободного пространства. Полученные снимки в процессе обработки необходимо сортировать, например, отсеять кадры с плохой освещённостью, для этого необходим алгоритм сортировки снимков.

-Так же при длительной съёмке освещённость объекта может изменяться. Причин этому несколько: солнце, дополнительный источник освещения и т.д. Не всегда возможно снимать объект в помещении с постоянной освещённостью, значит требуется коррекция освещённости.

Программа Virtual mouse была написана с целью имитации компьютерной мыши и управления движением курсора через web-камеру. Программа демонстрирует возможности камеры в качестве инструмента для получения информации о положении объекта, его координатах.

Возможности программы Virtual mouse: перемещение курсора в горизонтальной и вертикальной плоскостях; имитация нажатия левой и правой клавиши мыши. Предусмотрено 3 режима: движение курсора в вертикальной плоскости, движение в горизонтальной плоскости и имитация нажатия левой и правой клавиши мыши. Для работы с программой подходит любой объект, действия можно выполнять даже руками.

Кадр, получаемый с камеры, делится на 4 области: 1-я область отвечает за переключение режимов работы; 2-я область предназначена для перемещения курсора вверх, вправо и имитации нажатия правой клавиши мыши; 3-я область предназначена для перемещения курсора вниз, влево и имитации нажатия левой клавиши мыши. 4-я область сделана для облегчения позиционирования объекта, с помощью которого происходит управление курсором.

На начальном этапе программа получает изображение с камеры и считывает сумму пикселей в каждой из 3-х рабочих областей. Далее с заданным интервалом, указанным в графе «интервал», программа получает снимок с камеры, просчитывает сумму пикселей в областях нового кадра и сравнивает с исходной суммой пикселей. Если изменения в пределах заданного порога чувствительности, то программа не выполняет никаких действий и по истечении заданного интервала получает новый снимок, который впоследствии так же просчитывает. Если программа обнаруживает изменения в какой-либо области кадра, то принимает решение на выполнение запрограммированных действий, в зависимости от режима работы.

Во время работы с программой была замечена особенность, связанная с частотой получения снимков с web-камеры. При довольно большом интервале обновление изображения происходит медленно и не позволяет камере захватить быстрые перемещения управляющего объекта. В то же время большая частота захвата изображения приводит к ненужным нагрузкам на вычислительную систему компьютера. Поэтому при разработке подобных приложений приходится искать компромисс между быстродействием программы и удобством работы.

В этой работе презентуются разработанные продукты Time Catcher и Virtual mouse, демонстрирующие систему «web-камера и компьютер» в качестве установки для измерений.

Список цитированных источников

1. Шарыгин, М. Сканеры и цифровые камеры. – БХВ-Петербург, 2001.

УДК 004.418.2

АДДИТИВНАЯ СИСТЕМА БИЗНЕС-АНАЛИЗА НА ОСНОВЕ OLAP-ТЕХНОЛОГИЙ

Мартиневич О.А.

*УО «Белорусский государственный экономический университет», г. Минск
Научный руководитель – Литвинец В.И., доцент, к.т.н., ст. научный сотрудник*

Summary.

Due to installation BI system, based on multivariate data processing (OLAP-technology) which is created with the help dimensions hierarchy and devoted to the most important economic indicators analysis of enterprise, processes of financial analysis, financial planning and business modeling are simplified.

В современном мире успех напрямую зависит от того, как быстро менеджмент компании может *распознать изменения* динамики рынка и, исходя из существующих реалий, насколько своевременно отреагировать на них с целью увеличения прибыли. Внедрение системы бизнес-аналитики позволит создать единую информационно-аналитическую среду, позволяющую решать задачи стратегического развития предприятия и оперативного управления структурными подразделениями.

Целью методологии является - информационная поддержка управленческих функций, основанная на *консолидированной выборке и анализе* данных, позволяющая прогнозировать развитие бизнеса и, соответственно, принимать обоснованные решения.

Основные задачи методологии - консолидация информационных потоков из разнородных источников, расчет экономических показателей и статистических характеристик на основании ретроспективной информации, определение взаимосвязи показателей (причина-следствие), формирование графических и табличных представлений результатов вычислений, визуализация данных по имеющейся информации, применение имитационных методов моделирования, оптимизационного и статистического прогноза.

Область применения методологии – анализ экономических показателей организаций, функционирующих в сфере влияния множеств внешних и внутренних факторов.

Система бизнес-аналитики, разрабатываемая на базе MS Office PerformancePoint Server, включает в себя следующие компоненты: Business Modeler, Excel Add-in, SharePoint Web Parts, Dashboard Builder.

-Business Modeler – предназначен для формирования аналитических моделей.

-Excel Add-in – специальный модуль, встраиваемый в Excel, позволяет осуществлять ввод данных в аналитические модели, а также получать сводную аналитику;

-SharePoint Web Parts – специализированный набор веб-частей для портала SharePoint, позволяющий выводить аналитические модели и отчеты на веб-портал;

-Dashboard Builder - инструмент для проектирования систем показателей (Scorecards), систем индикаторов и цифровых панелей (Dashboards).

По завершении проектирования систем показателей и индикаторов данные системы будут выложены на портал, с которым будет работать основная часть пользователей. Для реализации данного решения, загрузки фактических данных и анализа деятельности фирмы разрабатываются соответствующие измерения.

В *Business Modeler* производится загрузка данных о клиентах, по каналам сбыта, товарам, счетам, складам. Так как данные будут храниться в *многомерном* представлении и каждое измерение представлено в виде иерархий, то появляется возможность проводить анализ в целом, так и *детализируя* возможную *информацию по любым измерениям*.

Для организации достоверного расчета финансовых показателей в измерения вводятся соответствующие статьи, а процедуры осуществляются с помощью написания *скриптов* на MS SQL-сервере, которые позволяют производить расчеты любой сложности. Помимо написания расчетов на сервере, создаются правила с использованием интерфейса Business Modeler. Путем задания соответствующих срезов, а также - введением специальных функций, встроенных в правила MDX- и SQL- имплементации, с учетом иерархии измерений настраиваются расчеты, участвующие в определении финансовых показателей.

Аналогично будут осуществляться расчеты прочих финансовых показателей (маржа, валовой доход, чистая прибыль, рентабельность и другие).

Для организации автоматической выгрузки данных из 1С: Предприятия в PerformancePoint Server будут разработаны *скрипты* и на SQL-сервере настроен запуск *автоматического обновления*, загрузки данных и пересчета всех показателей *ежедневно*, а также - за весь предыдущий месяц на первое число текущего.

Достоинства аддитивной системы бизнес - аналитики состоят в обеспечении полного контроля финансовых процессов организации, *получение актуальной информации* ежедневно. При этом система позволяет решать следующие задачи:

- *консолидировать информацию* из разнородных источников (внутренних систем оперативного учета данных, внешних источников) в хранилище данных, с предварительной очисткой, преобразованием и приведением информации к *корпоративной модели* данных;
- *рассчитывать* требуемые показатели и статистические *характеристики* на основании ретроспективной информации из хранилища данных; *определять взаимосвязи* показателей (производить проверку статистических гипотез, кластеризацию и пр.);
- *формировать наглядные графические и табличные представления* результатов вычислений и имеющейся информации (визуализация данных);
- *проводить эксперименты* с математическими моделями, описывающими поведение сложных систем, что позволит оценить обоснованность операций и их эффективность;
- *применять имитационные модели*, оптимизационные и поисковые процедуры, статистические методы прогнозирования и бюджетного управления;
- *осуществлять мониторинг процессов* благодаря системе показателей (табло);
- *анализировать причины отклонений* от ожидаемых показателей;
- *корректировать бюджет* и корпоративные планы в связи с текущим изменением условий хозяйствования.

Дополнительные возможности системы бизнес - аналитики позволяют осуществлять:

- *управление бизнес - знаниями* корпорации в реальном масштабе времени;
- простой *доступ к информации* для сотрудников компании различных уровней;
- сбор дополнительных показателей и соответственно рост объема перерабатываемой информации, *повышение конкурентоспособности* (следствие) и увеличение объемов продаж;
- проведение более *эффективного анализа доходов и расходов*;
- предоставление менеджерам корпорации *комплексной и наглядной информации* по всем направлениям развития бизнеса;
- *отслеживание текущих состояний* в корпорации (мониторинг) и ее региональных подразделениях.

Данная методология разработана при модификации структуры корпоративной информационной системы группы частных организаций и является основой их инкорпорирования в холдинг для более эффективного использования ресурсной части общего капитала.

УДК 681.3

ПРИНЯТИЕ РЕШЕНИЯ ПО УПРАВЛЕНИЮ ПОТЕНЦИАЛЬНО КРИЗИСНЫМИ ПРОЦЕССАМИ НА ОСНОВЕ НЕЙРОННОЙ СЕТИ С АДАПТИВНОЙ СТРУКТУРОЙ

Масалитина Н.Н.

*УО «Гомельский государственный технический университет им. П.О. Сухого», г. Гомель
Научный руководитель – Курочка К.С., к.т.н., доцент*

Принятие решения по управлению потенциальными кризисными изменениями сложных объектов сопряжено со сложностью оценки состояния объекта, а также определения последствий возможных управленческих решений. Кризисные процессы характеризуются высокой скоростью изменений, значительным уровнем ущерба от принятия решений, не соответствующих текущей ситуации, большим числом альтернативных решений и отсутствием априорной информации о критериях оптимальности выбора альтернативы. В таких условиях возрастает роль систем поддержки принятия решения. В качестве основы для разработки такой системы достаточно успешно могут быть применены искусственные нейронные сети, позволяющие исследовать сложные нелинейные зависимости между характеристиками объекта управления и изменением его состояния. Однако точность данного типа математических моделей в значительной степени зависит от выбора структуры сети: состава входных и выходных параметров, вида активационной функции, количества слоев и нейронов в каждом слое. Отсутствие в научной литературе рекомендаций выбору структуры искусственных нейронных сетей определяют необходимость выполнения исследования, направленного на решение перечисленных вопросов. Исследование выполнено на примере промышленных предприятий Гомельского региона. Полученные результаты могут быть использованы также для поддержки принятия решения по управлению кризисными процессами других сложных систем.

В ходе выполненного автором исследования построена математическая модель системы антикризисной устойчивости, описывающая взаимосвязь между уровнем развития отдельных механизмов противостояния кризисным изменениям и достигнутым уровнем устойчивости объекта управления. Модель реализована в качестве искусственной нейронной сети, имеющей следующие характеристики:

1) входной слой содержит 20 параметров, характеризующих уровень интенсивности применения отдельных механизмов, направленных на противостояние кризисным изменениям объекта управления, по которым зафиксирована значительная частота проявления и/или уровень ущерба;

2) выходной слой содержит один нейрон, принимающий одно из трех значений, соответствующий степени тяжести кризиса. Каждое состояние нейрона соответствует отдельному комплексу управленческих воздействий;

3) скрытый слой содержит два нейрона, один из которых отражает положительное влияние параметра на степень тяжести кризиса, другой отражает отрицательные побочные эффекты;

4) выбор активационной функции выполняется в процессе обучения нейронной сети.

С целью оптимизации выбора активационной функции нейронов был проведен ряд вычислительных экспериментов, в ходе которых были обучены более 250 нейронных сетей. В ходе исследования рассмотрены логарифмическая активационная функция, арктангенс, а также гауссиан, сигмоидальная функция и гиперболический тангенс с параметром крутизны от 0,1 до 2,0 с шагом 0,1. Обучение нейронных сетей было выполнено при помощи алгоритма обратного распространения ошибки.

Затем рассчитан средний процент распознавания состояния исследуемых объектов с использованием каждой функции. Также установлено, что результаты обучения нейронных сетей, основанных на каждой из десяти наилучших в рамках данной задачи функций, не являются стабильно высокими. Ряд экспериментов показывает хорошие результаты, другие значительно хуже. Поэтому ни одна из рассмотренных активационных функций не может быть окончательно рекомендована для решения задачи распознавания состояний потенциально кризисных объектов.

С целью повышения точности распознавания разработан алгоритм выбора активационной функции (рис. 1), основанный на последовательном обучении сетей, построенных на предварительно отобранных десяти активационных функциях, начиная с наиболее точной и заканчивая показавшей самые низкие результаты распознавания. Алгоритмом предусмотрена возможность выбора наиболее точной модели из числа построенных, а также прерывания цикла в случае достижения установленного лицом, принимающим решение, нижнего порога качества распознавания.

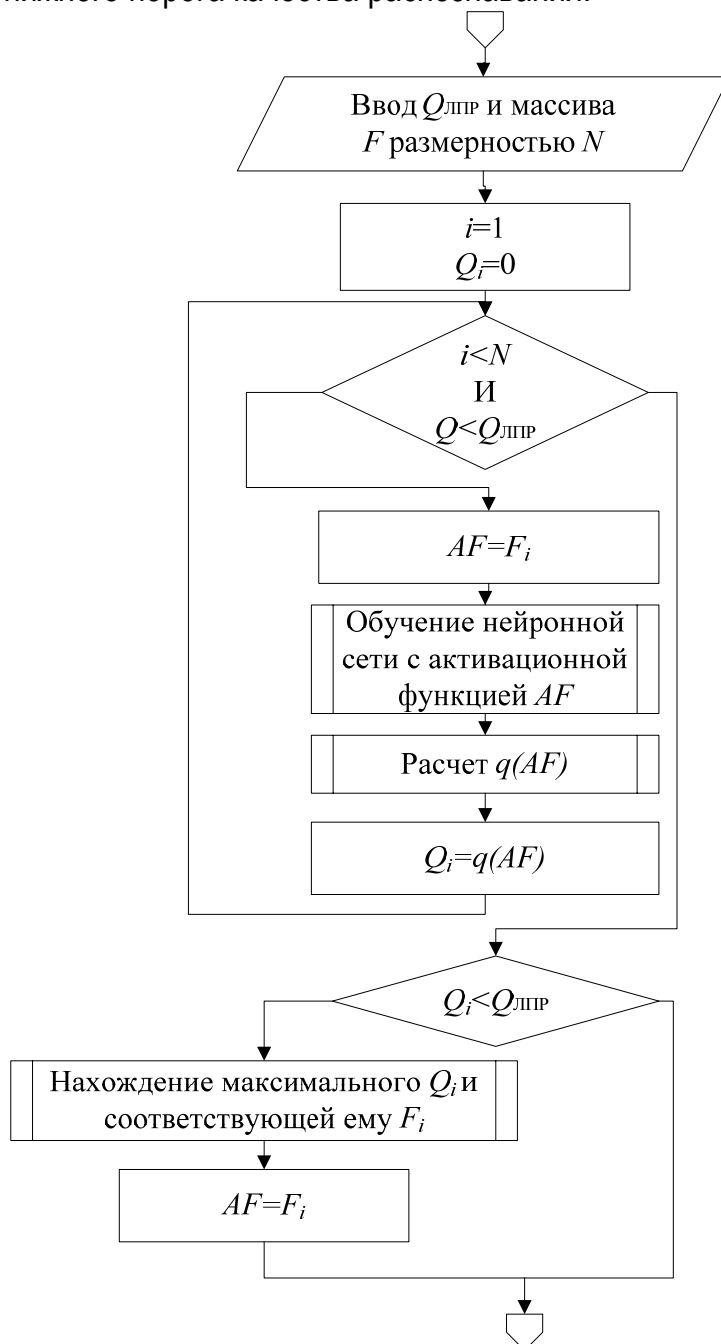


Рисунок 1 – Графическая схема алгоритма выбора активационной функции нейронной сети

Предложенный алгоритм позволил получить математические модели, верно распознающие в среднем 93,3% исследуемых объектов. При этом аналогичный показатель для математических моделей, построенных на основе случайного выбора активационной функции, составляет 53,2%. В случае использования функции, по которой зафиксированы наиболее высокие показатели в ходе выполненного исследования (гиперболический тангенс с параметром крутизны 0,5) данный показатель составляет 90,9%. Выбор наиболее точной из десяти моделей со случайно выбранной активационной функцией позволяет достичь в среднем 87% распознавания.

Примечание. На рисунке использованы следующие обозначения:

$Q_{\text{лпр}}$ – установленный лицом, принимающим решение нижний порог распознавания состояний исследуемых объектов, отн. ед.;

Q_i – доля верно распознанных состояний i -й нейронной сетью, отн. ед.;

F – массив из N функций, позволяющих получить наиболее точные модели, упорядоченный по убыванию точности;

AF – выбранная активационная функция нейронной сети;

$q(AF)$ – доля верно распознанных состояний нейронной сетью, построенной на основе выбранной на данном этапе активационной функции.

УДК 004.0896

СОЗДАНИЕ АВТОНОМНОГО РОБОТА НА БАЗЕ РАДИОУПРАВЛЯЕМОЙ МОДЕЛИ

Мешко Е.Э., Богуш А.С.

УО «Брестский государственный технический университет», г. Брест

Научный руководитель – Дунец А.П., доцент

Введение

Проделанная работа на базе радиоуправляемой игрушечной модели затрагивает инженерные и программные решения проблем, указанных ниже, которые впоследствии могут быть применены в роботизированных средствах передвижения.

Надо отметить, что роботы могут быть манипулируемые, управляемые человеком, и автономные, способные двигаться самостоятельно. В данной работе рассматривается автономный робот. Для движения автономных роботов характерны следующие проблемы:

- низкая маневренность;
- недостаточное ориентирование на маршруте (проблема логики движения);
- частые потери контроля маршрута следования (нечёткое движение по маршруту, выезд за его пределы или непредвиденные алгоритмом ситуации).

Ключевыми причинами являются:

- слабость алгоритмов управления;
- выбор ходовой части ;
- погрешность электроники и механики.

Описание робота

В качестве решения проблемы был сконструирован робот на базе игрушечного танка. Его дифференциальная кинематическая схема движения в значительной мере превосходит платформы с велосипедной схемой в том, что не возникает проблем с поворотом, так как робот может разворачиваться фактически на месте, это сразу же решает проблему с маневренностью. Для осуществления движения были установлены 2 мотора.

Они подключены к специально изготовленному драйверу моторов. Далее были изготовлены 3 инфракрасных датчика черной линии, которые подключены к аппаратно-вычислительной управляющей платформе Arduino на основе микроконтроллера ATmega168. С её помощью осуществляется управление роботом. Эта популярная платформа выбрана из-за простоты прошивки и кодирования. Питание у робота раздельное: 4 батарейки стандарта AA питают моторы, девятивольтовая батарея питает электронику.

Алгоритм

Описание логики движения робота по черной линии можно представить в виде графа состояний (рисунок 1).

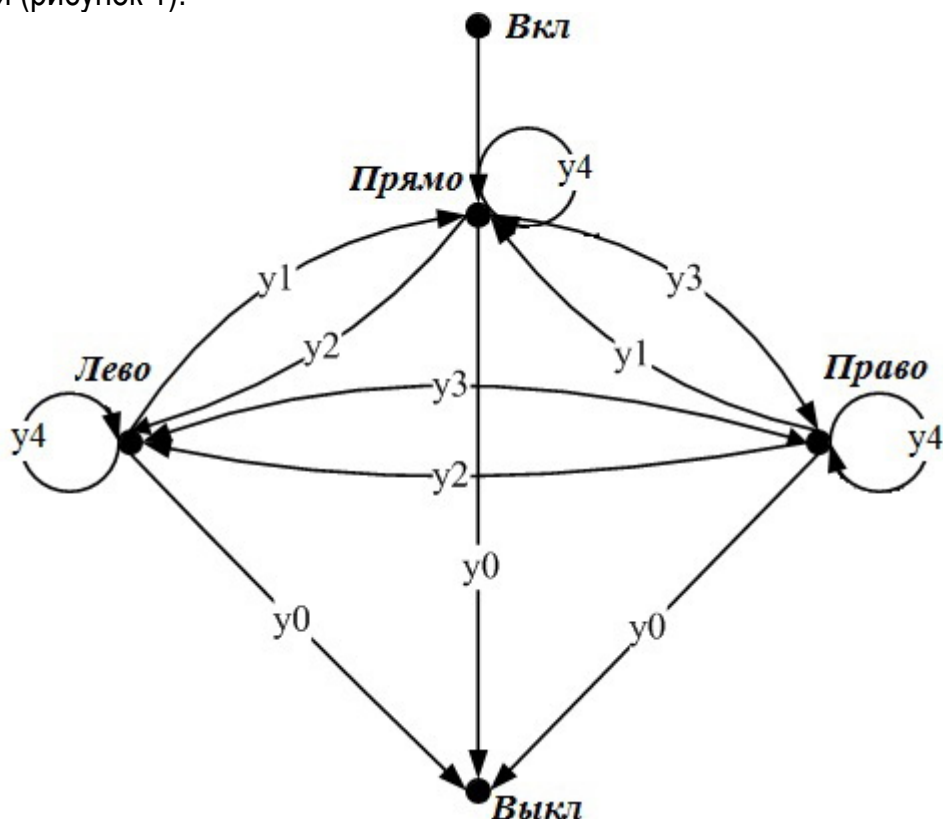


Рисунок 1 – Граф состояний

Состояния:

- | | |
|--|---|
| x_1 – левый датчик на черном | x_1 – левый датчик на белом |
| x_2 – центральный датчик на черном | x_2 – центральный датчик на белом |
| x_3 – правый датчик на черном | x_3 – правый датчик на белом |
| y_0 – переключатель выключен | $y_1 = x_1^{-1}x_2x_3$ – прямо |
| $y_2 = x_1^{-1}x_2x_3 \vee x_1^{-1}x_2x_3$ – влево | $y_3 = x_1^{-1}x_2x_3 \vee x_1^{-1}x_2x_3$ – вправо |
| $y_4 = x_1^{-1}x_2x_3 \vee x_1^{-1}x_2x_3 \vee x_1^{-1}x_2x_3$ – выполнение последнего состояния | |

Словесное описание алгоритма следующее: после включения робот движется до тех пор, пока датчики на белой линии. Если правый датчик или средний и правый находятся на черной линии, следует выполнить поворот, до тех пор, пока средний датчик вновь не окажется на черной линии. Аналогично для состояния левого датчика или левого и среднего одновременно. В случае, когда все датчики на черном, все на белом или крайние на черном, а средний на белом, роботу дается команда выполнять последнее состояние.

Простота алгоритма позволяет с легкостью использовать его для управления роботом. Данный алгоритм позволяет нивелировать неточности, вызванные шасси, возвращая робота на линию. Например, если робот ехал прямо по линии, и левую гусеницу заклинило. Из-за различного вращения гусениц робот начнет поворачивать влево, что приведет к потере линии. Центральный датчик потеряет линию. При этом последним ее увидит правый боковой датчик. Произойдет поворот робота в другую сторону, что вернет его на линию. Алгоритм также позволяет двигаться по незамкнутой линии, за счет запоминания и выполнения последнего состояния.

Результаты экспериментов

В результате моделирования получился робот с именем «RobT-34». На этапе подготовки все тесты проводились в университете на импровизированной трассе. После ряда испытаний было принято решение протестировать робота в более сложных условиях и принять участие в международных соревнованиях «ROBORACE», где робот показал себя очень достойно, заняв 5 место. Из соревнований было получено много интересных впечатлений, большое число идей, а также ценнейший опыт. Конечно же, возник ряд трудностей, которые не были предусмотрены. Не было учтено, что на трассе могут присутствовать несколько роботов, что приводило к столкновениям, создававшим трудности, из-за того, что инфракрасные датчики были установлены клином. В резкие, угловатые повороты менее 45 градусов робот не входил или тратил слишком много времени на их преодоление.

Заключение

В ближайшем будущем планируется внесение изменений в конструкцию робота: установка новой управляющей платы, увеличение скорости движения, замена датчиков на инфракрасные датчики расстояния, модификация всех составляющих и подготовка к следующему этапу соревнований ROBORACE.

Полученный опыт в конструировании и программировании робота на базе игрушечной модели танка является очень ценным. При помощи таких проектов студенты могут на практике применять и совершенствовать свои знания в области программирования, механики и электроники. Это делает процесс обучения более интересным и захватывающим, а также стимулирует студентов вести научную деятельность.

УДК 004.896

АВТОНОМНЫЙ МОБИЛЬНЫЙ РОБОТ ДЛЯ УЧАСТИЯ В КОЛЬЦЕВЫХ ГОНКАХ

Пучик А.А., Хомиченко Д.В.

*УО «Брестский государственный технический университет», г. Брест
Научный руководитель – Дунец А.П., доцент*

Введение

В данной работе описано создание автономного робота для участия в соревнованиях, таких как RoboRace. Робот создавался на основе микроконтроллера и датчиков, позволяющих ему ориентироваться в пространстве. Были использованы две модели датчиков, которые были установлены на платформу, – инфракрасные и механические. Инфракрасные использовались для отслеживания черной линии, механические, в роли которых выступали кнопки, использовались для определения столкновения робота.

Правила гонок

RoboRace - это соревнования автономных мобильных роботов, организованное по принципу Формулы 1. Для участников соревнований нет возрастных, территориальных, национальных и других ограничений. В любом из этапов соревнований могут принять участие как студенты и школьники, так и специалисты в области робототехники. В соревнованиях участвуют как простые узконаправленные конструкции, созданные начинающими робототехниками, так и комплексные робототехнические устройства со сложной программной частью. Соревнования идут в несколько заездов на вылет. Побеждает тот, кто проехал требуемое количество кругов первым. Роботы должны обгонять друг друга без столкновений, за столкновения начисляются штрафные очки. Движение роботов осуществляется по трассе с заранее нанесенными на нее черными линиями и установленными бортами. Допускается движение робота по любой из линий или любым иным способом, который подразумевает автономную ориентацию. Размеры робота не более 50 см x 25 см, вес не должен превышать 3 кг.

Механика модели робота

Робот представляет собой четырёхколёсную платформу игрушечной машинки на радиоуправлении, к которой прикреплено 6 инфракрасных датчиков, определяющих черную линию, и два механических, определяющих столкновение робота с препятствием (Рисунок 1).



Рисунок 1 – Внешний вид робота

Датчики находятся на передней части робота, с их помощью которых он принимает сигнал о положении линии. Также на машинку установлен микроконтроллер для обработки данных, передаваемых с датчиков, драйвер двигателя, позволяющий управлять скоростью и направлением движения робота, сервопривод, отвечающий за поворот передних колес. На робота было установлено восемь аккумуляторов типа АА (1,3 вольта) для питания двигателей, датчиков и микроконтроллера. Также был установлен защитный корпус, позволяющий спрятать всю электронику и предотвратить её повреждение.

Платформа робота была взята с игрушечной модели джипа.

Эта модель была выбрана по некоторым факторам:

- высокая подвеска;
- легкая конструкция;
- небольшие размеры;
- прочный корпус.

Датчики прикреплены к ПВХ в два ряда. Этот материал был выбран, так как он лёгкий и достаточно прочный для крепления и защиты датчиков от столкновений. Датчики расположены в два ряда: в первом стоит один средний, а во втором расположено пять датчиков так, чтобы третий был на одном уровне с передним центральным датчиком (Рис. 2).

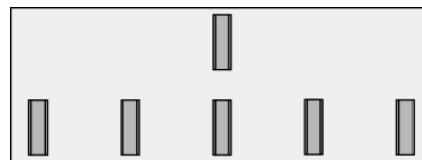


Рисунок 2 – Расположение инфракрасных датчиков

Такое расположение датчиков позволило нам решить задачу отслеживания линии.

Так же на робота было установлено две кнопки, которые служили сигналом о том, что он ударился в стену или в робота соперника. На кнопки был прикреплен кусок железной пластины, который позволил увеличить рычаг срабатывания датчика.

Тестирование

После установки всей электроники на платформу и прошивки микроконтроллера были проведены первые испытания алгоритма на реальном роботе. Изначально был установлен алгоритм простейших движений, таких как ехать вперёд, назад и поворачивать в одну из сторон. На начальном этапе испытания датчики не были задействованы, проверялась работа двигателя мотора и сервопривода. Так была проверена электроника и простейшие алгоритмы поведения робота. На следующем этапе были установлены датчики и загружен алгоритм, позволяющий роботу реагировать на их показания. После установки всей электроники возник ряд недостатков, которые будут описаны далее.

Испытания датчиков проводились на специальной трассе кольцевого типа с нанесёнными на неё черными полосами.

Перед установкой датчиков робот был достаточно быстрый и лёгкий. После установки датчиков и дополнительных элементов питания был обнаружен ряд значительных недостатков:

- уменьшение скорости;
- увеличение габаритов;
- смещение центра тяжести;
- увеличение веса.

Из-за большого веса изначально робот не мог тронуться с места, так как двигатель и редуктор были предназначены для более легкой платформы. Было принято решение: найти более мощный двигатель и собрать новый редуктор, который давал бы больше мощности. Это сразу же привело к тому, что робот потерял значительную часть скорости.

Увеличение габаритов привело к увеличению радиуса поворота робота. Эта задача была решена с помощью логики и программного кода.

Смещение центра тяжести привело к тому, что робот начал терять сцепление с поверхностью трассы. Робот стоял на месте и буксовал ведущими колёсами. Для устранения этого недостатка было найдено решение. Заключалось оно в том, что на колёса, которые стоят на платформе, были одеты резиновые кольца, что привело к увеличению сцепления с трассой, но результат не был оправдан. Тогда было решено увеличить вес на заднюю (ведущую) ось, но с увеличением веса снизилась скорость робота. В итоге робот получился мощным.

Тестирование робота на трассе было успешным. Выбранная модель расположения датчиков оправдала себя. Робот уверенно ехал по черной линии, не теряя её. Входил в острые повороты, подымался на горку без особых проблем. Единственной проблемой робота являлась его низкая скорость. Однако именно мощность, а не скорость давала преимущество на подъеме и спуске 3D-трассы на соревнованиях в Львове, где робот занял второе место.

Заключение

В дальнейшем планируется:

- взять гусеничную платформу и оборудовать её ультразвуковыми датчиками, с помощью которых использовать логику “принцип туннеля”. Этот робот будет использоваться для участия в соревнованиях RoboRace;

- создать двух гоночных роботов, которые могли бы взаимодействовать между собой и принимать общие решения.

На данном этапе наша команда работает над роботом, который сможет ориентироваться по установленной на нем видеокамере.

УДК 621.391.(075.8)

МЕТОДЫ РАЗДЕЛЬНОГО КОДИРОВАНИЯ ИНФОРМАЦИИ

Смолякова О.Г.

УО «Белорусский государственный университет информатики и радиоэлектроники», г. Минск

Блинов И.Н.

УО «Белорусский государственный университет», г. Минск

Для защиты данных от помех широко применяются методы кодирования информации. Основным методом декодирования многих кодов является синдромный метод. При использовании синдромного декодирования все символы защищаемой последовательности рассматриваются как одинаково важные и защищается вся информационная последовательность целиком. При необходимости же коррекции многократных ошибок этим методом возникает проблема селектора [1].

Для случая, когда передаваемая информация имеет более значимые и менее значимые участки, разработаны методы раздельного кодирования. Под раздельным кодированием понимается подход к кодированию информации, когда защищаемая информация делится на блоки, каждый из которых кодируется отдельно. Закодированные раздельно данные можно представить в виде последовательности информационных и проверочных символов каждого блока (рисунок 1). В случае неравномерного раздельного кодирования каждый блок кодируется собственным кодом $C_{(k)}$.

информационные символы блока 1	проверочные символы блока 1	...	информационные символы блока N	проверочные символы блока N
--------------------------------	-----------------------------	-----	--------------------------------	-----------------------------

Рисунок 1 – Представление информации при раздельном кодировании информации

Основное преимущество методов табличного раздельного кодирования и табличного приоритетного кодирования заключается в возможности использования кодов с небольшой длиной, а, следовательно, упрощается процедура синдромного декодирования.

Сущность метода табличного раздельного кодирования заключается в делении информационной последовательности на блоки, которые располагаются друг под другом, причем каждый блок кодируется кодом, кодовое расстояние которого удовлетворяет требуемым корректирующим способностям по защите текущего блока данных.

Метод табличного приоритетного кодирования разбивает информационные данные на блоки, которые располагаются друг под другом согласно убыванию степени требуемой защиты ($t_1 < t_2 < t_3 < \dots$, t – кратность корректируемой ошибки). Каждый блок кодируется кодом, кодовое расстояние которого удовлетворяет требуемым корректирующим способностям по защите текущего блока данных.

На рисунке показано применение метода табличного приоритетного кодирования при использовании четырех различных информационных блоков.

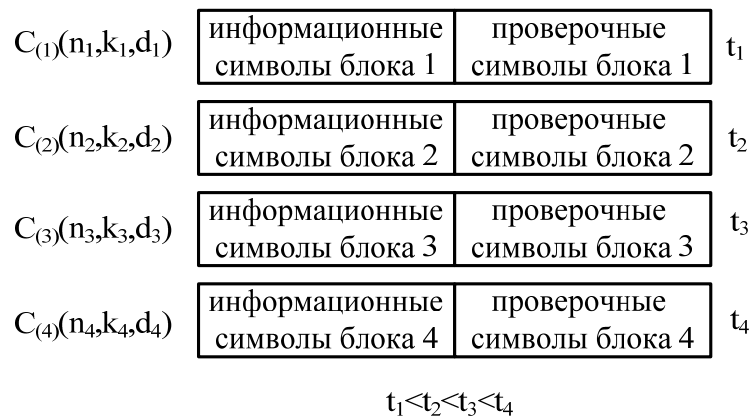


Рисунок 2 – Представление информации при использовании метода табличного приоритетного кодирования

Использование методов табличного отдельного кодирования и табличного приоритетного кодирования позволяет повысить информационную надежность передаваемых данных за счет использования кодов с различными длинами и кодовыми расстояниями и упростить процедуру декодирования. Разбиение информации на блоки допускает распараллеливание как процедуры декодирования, так и процедуры кодирования. Количество синдромов, используемых в процедуре селектирования в случаях одномерного кодирования [2] и табличного отдельного кодирования, приведены в таблице. Анализ данных таблицы показывает, что разбиение информационной последовательности только на два блока позволяет сократить число селектируемых синдромов минимум с 66 раз для $t=6$.

Таблица – Число селектируемых комбинаций при одномерном и табличном отдельном кодировании

Одномерное кодирование	Табличное отдельное кодирование		
$t=6, n=511$	t	разбиение на	
		2 блока, $n=255$	4 блока, $n=127$
24 010 353 303 937	6	359 895 314 625	5 169 379 425
	5	8 637 487 551	254 231 775
	4		10 334 625
	3		333 375

Применение методов отдельного кодирования возможно как при заранее определенном числе блоков, так и при определении этого значения автоматически. Жесткий способ отдельного кодирования информации (жесткий кодек) заранее определяет количество блоков и используемые коды, которые известны кодирователю и декодирователю. При адаптивном способе отдельного кодирования (адаптивный кодек) количество блоков и используемые коды заранее неизвестны и определяются процедурой кодирования. Адаптивный кодек добавляет к данным информацию о числе блоков, используемых кодах и порядке их применения (рисунок 3).

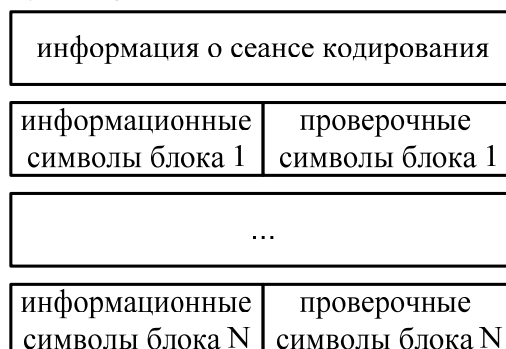


Рисунок 3 – Представление информации при адаптивном отдельном кодировании

Использование методов отдельного кодирования при защите данных позволяет варьировать степень помехоустойчивой защиты информации, кодируя более значимую часть сообщения кодом с большей корректирующей способностью.

Список цитированных источников

1. Теория прикладного кодирования: Учеб. пособие для студентов инженер.-техн. специальностей вузов: в 2-х т. / В. К. Конопелько, В.А. Липницкий, В.Д. Дворников [и др.]; под ред. проф. В.К. Конопелько. – Мн.: БГУИР, 2004. – Том 1. – 285 с.
2. Вернер, М. Основы кодирования: учебник для вузов / М. Вернер. – М.: Техносфера, 2006. – 288 с.

УДК 004.4'242

ПОСТРОЕНИЕ КОРПОРАТИВНЫХ БИЗНЕС-ПРОЦЕССОВ НА IBM WEBSHERE PROCESS SERVER

Суворов В.В., Лещёв А.Е.

*УО «Белорусский государственный университет информатики
и радиоэлектроники», г. Минск*

Научный руководитель – Пилецкий И.И., к. ф-м. наук, доцент

Управление бизнес-процессами (BPM) нацелено на внедрение деловых инноваций и оптимизацию с помощью внедрения стратегии бизнеса на основе моделирования разработки и управления бизнес-процессами на протяжении всего их жизненного цикла. BPM дает возможность бизнесу определить и реализовать стратегические цели бизнеса, а затем оценивать и управлять финансовой и оперативной эффективностью компании для достижения этих целей[1]. Мощь оптимальных результатов, входящих в жизненный цикл BPM, вытекает из интегрированного набора устойчивых технологических инфраструктур и инструментов.

Автоматизация бизнес-процессов обеспечивает сближение технологии, устраняющей ограничения бизнеса и ИТ с помощью интеграции, и усовершенствованной технологии, чтобы помочь облегчить преобразование бизнеса. Эти возможности обеспечивают прочную связь операционной и аналитической среды, бизнес и ИТ-среды, стратегии и ежедневных действий.

Построение бизнес-процессов соединяет информацию и ресурсы ИТ, согласуя основные ресурсы организации – людей, информацию, технологию и процессы, чтобы создать единое интегрированное представление, позволяющее получать в реальном времени оценки показателей бизнеса и производительности ИТ-систем. Эта интеграция ресурсов позволяет предприятию быстрее получать бизнес-информацию, скорее реагировать на тенденции рынка и угрозы со стороны конкурентов и повысить операционную производительность и бизнес-результаты.

Платформа IBM WebSphere позволяет строить качественные информационные системы для сферы бизнеса. Позволяя строить многоуровневые ИТ-решения, платформа сама по себе является достаточно сложной. Трудности при ее освоении во многом связаны с тем, что платформа образована широким набором программных продуктов IBM (включая семейства продуктов Rational, Tivoli, Lotus и др.), которые необходимо надлежащим образом установить, сконфигурировать и интегрировать. Более того, сложность освоения существенно возрастает за счет того, что платформа IBM WebSphere является одной из новых технологий разработки ПО мирового уровня и опыта работы с ней в РБ недостаточно.

В основе IBM WebSphere лежит сервис-компонентная архитектура (SCA) и сервисные объекты данных (SDO).

Сервис-компонентная архитектура – это новая технология, упрощающая разработку и ввод в действие приложений в сервис-ориентированной архитектуре 2. С помощью SCA клиенты могут проще разрабатывать новые и трансформировать существующие активы ИТ в пригодные для повторного использования сервисы, которые могут быть быстро приспособлены к изменяющимся бизнес-требованиям. Более того, эта новая технология существенно снижает сложность разработки, предоставляя возможность унифицировать сервисы независимо от того, на каком языке программирования они написаны и на какой платформе должны быть запущены.

Сервисные объекты-данных – это технология, которая изначально развивалась совместно BEA и IBM, а теперь развивается BEA, IBM, Oracle, SAP, Siebel, Sybase и XCalia. SDO спроектированы для того, чтобы упростить и унифицировать способ, которым приложения работают с данными. Используя SDO, прикладные программисты могут единообразно получать доступ и манипулировать данными из разнотипных источников, включая реляционные базы данных, источники данных XML, Web-сервисы и информационные системы предприятия.

Таким образом архитектура IBM WebSphere обеспечивает модульную, масштабируемую, переносимую среду для поддержки бизнеса.

Портфель процессной интеграции IBM обеспечивает возможности, необходимые для обеспечения стратегии и решений для всеобъемлющего управления бизнес-процессами в масштабе предприятия. Он предлагает единый подход для преобразования и управления бизнесом, уравнивая стратегию и оперативные цели с бизнес-действиями и поддерживающими ИТ-службами.

Решения IBM для BPM включают инструменты для разработки, используемые для реализации заказных артефактов, усиливающих возможности среды, и инструменты для управления эффективностью бизнеса, используемые для мониторинга и управления реализацией во время исполнения на уровнях ИТ и бизнеса[3].

Входящие в состав решения IBM для BPM инструменты поддерживают следующие наиболее важные виды деятельности:

- моделирование – захват, имитация, анализ и оптимизация бизнес-моделей с целью снижения рисков и увеличения гибкости;
- сборку – разработку, сборку и тестирование интегрированных решений;
- размещение – прямое размещение моделей и политик для реализации намерений бизнеса;
- управление размещенной моделью;
- мониторинг и установление зависимости метрик и предупреждений, получаемых в реальном времени от внутренних и внешних источников, чтобы обеспечить визуализацию эффективности бизнеса и ИТ;
- анализ показателей эффективности с целью понимания сути бизнес-метрик и информации для принятия решений на основании контекста;
- ответные действия на основе своевременного понимания происходящего с помощью взаимодействия, оптимизации и автоматизации для получения превосходства.

Одной из ключевых особенностей управления бизнес-процессами от IBM является связь между платформой разработки и службами управления эффективностью бизнеса.

Список цитированных источников

1. Wahli U., Leybovich L., Prevost E., Scher R., Venancio A., Wiederkom S., MacKinnon N. Business Process Management: Modeling through Monitoring Using WebSphere V6 Products, 2006
2. Service Component Architecture [Электронный ресурс] - Электронные данные. - Режим доступа: <http://www.ibm.com/developerworks/webservices/library/specification/ws-sca/>
3. WebSphere Process Server [Электронный ресурс] - Электронные данные. - Режим доступа: <http://www.ibm.com/software/integration/wps/>

УДК 004.896

АЛГОРИТМ ДВИЖЕНИЯ АВТОНОМНОГО МОБИЛЬНОГО РОБОТА ДЛЯ УЧАСТИЯ В КОЛЬЦЕВЫХ ГОНКАХ

Хомиченко Д.В., Пучик А.А.

*УО «Брестский государственный технический университет», г. Брест
Научный руководитель – Дунец А.П., доцент*

В данной работе описан алгоритм ранее созданного автономного робота для участия в соревнованиях, таких как RoboRace. Алгоритм разрабатывался на основе технических характеристик робота. Алгоритм обеспечивает роботу максимально точное движение по линии и минимальную вероятность ее потери.

Правила гонок

RoboRace - это соревнование автономных мобильных роботов, организованное по принципу Формулы 1. Движение роботов осуществляется по трассе с заранее нанесенными на неё черными линиями и установленными бортами. Допускается движение робота по любой из линий или любым иным способом, который подразумевает автономную ориентацию.

Алгоритм движения робота разрабатывается каждой командой для своего робота.

Технические данные робота

Характеристики робота: четырёхколёсная платформа, 6 инфракрасных датчиков, определяющих линию, 2 механических датчика, воспринимающих столкновения, микроконтроллер и драйвер двигателя, позволяющий управлять скоростью и направлением движения робота, сервопривод, отвечающий за поворот передних колес.

Инфракрасные датчики находятся на передней части робота и расположены в два ряда: в первом стоит один средний (далее будем называть его “передним центральным”), а во втором (датчики второго ряда, далее будем называть “левый”, “левый центральный”, “центральный”, “правый центральный” и “правый” соответственно) расположено пять датчиков так, чтобы третий был на одном уровне с передним центральным датчиком (Рис. 1).

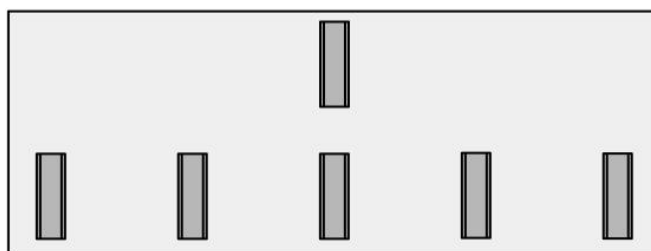


Рисунок 1 – Расположение инфракрасных датчиков

Логика движения робота

Движение робота осуществляется посредством поворота его передних колёс, угол поворота которых определяется положением черной линии относительно датчиков. Робот движется прямо на полной скорости, если линию обнаруживают оба центральных датчика. При пропадании линии с одного из датчиков - робот снижает скорость на 40% для подготовки к повороту (после нескольких экспериментов привело к выводу, что при меньшей скорости радиус поворота у робота меньше) и ожидает сигнал от “левого центрального” или “правого центрального” датчика, получив который, робот поворачивает влево или вправо соответственно. Если поворот очень резкий и робот не может отследить линию центральными датчиками, то линию фиксируют “левый” или “правый” датчики. Следует заметить то, что при появлении линии на одном «левом» или «правом» датчике робот ожидает сигнала на «левом центральном» или «правом центральном» соответственно. В случае потери линии и этими датчиками, то робот продолжает поворот в ту же сторону до появления сигнала на одном из центральных датчиков, после чего выравнивает движение робота по линии.

Так же был разработан метод отслеживания линии. Этот метод позволяет ехать роботу подобной конструкции вдоль одной линии, не теряя её. Метод основывается на том, что определенные датчики, получив сигнал о черной линии, влияют на поведение робота только при некоторых условиях. «Центральные боковые» датчики активируются в любых ситуациях, т.к. это означает, что линия отклонилась в ту или иную сторону от центра; или же робот пытается центрировать линию относительно себя. Крайние боковые датчики учитываются только после того как линия появится на соответствующем “боковом центральном” датчике - это позволяет нам при появлении другой линии ее не учитывать. Центральные датчики активируются после любого варианта событий. Благодаря этому робот может выбрать более плавную линию для следующего круга.

Алгоритм работы робота описан функциями, и тело основной исполняемой функции выглядит как приведено на рисунке 2.

```
void loop()    // Основной цикл управления
{
  if ( !ButtonRead() ) // Проверяем механические кнопки
  {
    Control();    // Сбор информации с датчиков и анализ ситуации
    Move();      // Движение робота
  }
}
```

Рисунок 3 – Алгоритм работы основной исполняемой функции

При срабатывании кнопок на препятствие робот поворачивает передние колеса в одну или другую сторону и отъезжает назад, немного едет прямо (это позволяет роботу иметь некоторую вероятность прохождения трассы в случае поломки датчиков). Скорость обработки запросов очень велика, поэтому нажатие сразу двух кнопок «словить» практически нереально, для этого после нажатия на одну кнопку было вставлено небольшое ожидание (0.1 сек) и снова делался запрос на нажатие кнопок. Если сработали сразу обе кнопки, то робот просто отъезжает назад и дальше пытается ориентироваться по линии. Вероятность того, что кнопки зациклят робота на одном промежутке трассы очень мала, и за все время работы робота таких ситуаций не было. Для считывания кнопок была разработана отдельная функция (рисунок 3).

```
boolean ButtonRead()
{
  if (digitalRead(Butt_L) == 0) // проверка нажатия левой кнопки
  { delay(100); // ожидание нажатия 2-й кнопки
    if (digitalRead(Butt_R) == 0) // проверка нажатия правой кнопки
      CenterBack(); // движение при нажатии 2-х кнопок
    else LeftBack(); // движение при нажатии левой кнопки
  }
  if (digitalRead(Butt_R) == 0)
  { delay(100);
    if (digitalRead(Butt_L) == 0)
      CenterBack();
    else RightBack(); // движение при нажатии левой кнопки
  }
  if (digitalRead(Butt_L) == 0 || digitalRead(Butt_R) == 0)
    return 1; // было срабатывание кнопок
  else return 0; // не было срабатывания кнопок
}
```

Рисунок 3 - Алгоритм выполнения считывания кнопок

Тестирование

После прошивки микроконтроллера были проведены первые испытания алгоритма на реальном роботе. С первого раза не достигли лучшего результата, т.к. для всякого алгоритма существуют коэффициенты, которые должны быть откорректированы на опытах.

Тестирование робота на трассе было успешным. Разработанная логика оправдала себя. Робот уверенно ехал по черной линии и не терял её, столкновений практически не было. Робот на соревнованиях во Львове на 3D-трассе занял второе место.

Заключение

В дальнейшем планируется:

- использовать подобную логику для гусеничной платформы,
- разработать логику для робота, воспринимающего информацию с камеры.

Сейчас идет активная работа над разработкой логики, основанной на показаниях датчиков.

АНАЛИЗ И МОДЕЛИРОВАНИЕ СЛОЖНЫХ СИСТЕМ

УДК 656.2

ПОИСК КРАТЧАЙШИХ ПУТЕЙ НА СЕТЯХ СО СТРУКТУРНЫМИ ОГРАНИЧЕНИЯМИ

Батура П.М., Шешко Е.В.

УО «Белорусский государственный университет информатики
и радиоэлектроники», г. Минск

Научный руководитель – Ревотюк М.П., к. т.н, доцент

Пусть транспортная сеть представлена нагруженным ориентированным графом $G(M, N)$, где N и M – множества вершин и дуг графа, а каждой дуге $(i, j) \in M$, $i, j \in N$, соответствует неотрицательное вещественное число $w(i, j) < \infty$, называемое длиной дуги. Если между вершинами s и t , $s, t \in N$ существует ориентированный путь, то длиной пути будем называть сумму длин дуг, образующих путь.

Обозначим для любой дуги $(i, j) \in M$ графа $G(M, N)$ множество допустимых вершин для развития путей из вершины $j \in N$ через $cont(i, j)$, а для любой вершины $x \in N$ множество ее смежных вершин – x' : $x' = \{k | w(x, k) \geq 0\}$.

Очевидно, что $cont(i, j) \subseteq j'$, а в случае отсутствия ограничений на выбор пути после прохождения дуги (i, j) – $cont(i, j) \equiv j'$, $i, j \in N$, $(i, j) \in M$.

Решаемая задача: найти, если существует, на множестве вершин $\{s_0 = s, s_1 \in cont(s, s), s_2 \in cont(s_0, s_1), \dots, s_i \in cont(s_{i-2}, s_{i-1}), \dots, t\}$ кратчайший путь от вершины s к вершине t , где $s, t \in N$.

Содержательный смысл функции $cont(i, j)$ может иметь отношение, например, к правилам дорожного движения, определяющих альтернативы выбора дуг в узле сети с учетом параметров транспортного средства. Другой пример – задачи управления подвижными единицами промышленного транспорта, когда необходимо учесть занятость участков сети на интервалах уже реализуемых маршрутов перевозки. Введение функции $cont(i, j)$ отражает намерение разделения модели транспортной сети и модели перемещения по ее дугам.

Модель графа $G(M, N)$ обычно задана структурой смежности в форме представления *FSF* (*Forward Star Form*) [1,2]:

$$FSF = \begin{cases} H_0 = 0, \\ H_{i+1} = H_i + |i'|, \quad i = \overline{1, n-1}; \\ Y(H_i + j) = i'(j), \\ W(H_i + j) = w(i, i'(j)), \quad j = \overline{0, |i'|-1}, \quad i = \overline{0, n-1}. \end{cases} \quad (1)$$

Модель перемещений – отображение дополнительных параметров дуг на множество функций $cont(i, j)$, может быть определена системой продукционных правил, например, соответствующих правилам дорожного движения. Структура (1) при этом остается без изменений.

Ограничения на выбор пути после прохождения некоторой дуги $(i, j) \in M$ формально можно рассматривать как изменение структуры графа. Отсюда следует, что искомый алгоритм может быть построен путем целенаправленной модификации алгоритма Дейкстры [2]. Общая схема такого алгоритма соответствует жадной волновой схеме построения дерева кратчайших маршрутов от заданной исходной вершины $s \in N$ до всех остальных. Случай поиска кратчайшего пути до заданной конечной вершины $f \in N$ легко учитывается введением проверки необходимости развития волны после прохождения очередной вершины.

Однако при неудачном выборе метода расстановки пометок вершин в реализации алгоритма Дейкстры [2] можно получить неверный результат [3]. Действительно, пусть дерево кратчайших путей характеризуется множеством предшествующих вершин $\{P(x), x \in N\}$. Значение расстояния до вершин дерева обозначим через $\{R(x), x \in N\}$. Состояние процесса построения дерева при отсутствии ограничений пусть отражается в очереди вершин $\{Q(x), x \in N\}$. Элементы очереди $Q = \{q_1, q_2, \dots, q_i\}$ в этом случае должны быть упорядочены по значениям расстояний так, что $R(q_1) \leq R(q_2) \leq \dots \leq R(q_i)$.

Известный алгоритм поиска пути $s \rightarrow f$ на основе очереди вершин [2]:

```

for  $x \in N$  { // Инициализация результата
   $R(x) = \infty$ ;  $P(x) = x$ ;
}
 $R(s) = 0$ ;  $Q = s$ ; // Фиксация корня дерева
do { // Организация ветвления
   $x = Q--$ ;
  if ( $x == f$ ) break; // Прерывание цикла при достижении цели
   $r = R(x)$ ;
  for ( $y \in x'$ )
    if ( $R(y) < r + w(x, y)$ ) {
      if ( $R(y) < \infty$ )  $Q -= y$ ;
       $R(y) = r + w(x, y)$ ,  $P(y) = x$ ;
       $Q += y$ ;
    }
} while ( $Q \neq \emptyset$ );

```

На первый взгляд, в таком алгоритме достаточно заменить выражение $y \in x'$ на $y \in \text{cont}(P(x), x)$, и задача поиска кратчайших путей с ограничениями решена. Но характерная особенность рассматриваемой задачи – зависимость процесса расширения дерева путей от предыстории. Например, граф с ограничениями может включать цикл, когда пропущенные дуги некоторой вершины могут стать продолжением пути из последующих вершин пути. Другой пример – возврат в начальную вершину дуги после разрешенного правилами движения разворота. Как итог, решение о включении вершины в дерево кратчайших путей после ее выборки из очереди не является окончательным, хотя итерации поиска будут завершены.

Альтернативы развития путей, отражающие реальные ограничения, можно учесть перестройкой исходного графа [3]. Однако переход к поиску кратчайших путей на новом графе без учета ограничений требует не только расширения исходного графа, но и

уточнения интерпретации получаемых результатов. Неудобства реализации подобного приема становятся практически неприемлемыми в многозадачной среде сервера обслуживания запросов, параметризованных относительно вида ограничений.

Вместе с тем можно исключить потребность перестройки исходного графа, если отображать процесс построения дерева кратчайших путей не на очередь вершин, а на очередь дуг $\{Q(x, y), (x, y) \in M\}$. Позиция дуги (x, y) в очереди дуг пусть, подобно очереди вершин, соответствует потенциалу ее конечной вершины $L(x, y), (x, y) \in M$. Просматриваемая дуга, помещенная в очередь, изменит свое состояние лишь один раз в момент выборки из очереди. Это не только упрощает структуру представления очереди (достаточно использования линейного списка), но и позволяет учесть различные условия развития дерева из любой вершины или дуги.

Алгоритм волновой схемы поиска пути $s \rightarrow f$ на основе очереди дуг [3]:

```

for (x ∈ N) { // Инициализация области данных результата
  R(x) = ∞, P(x) = x;
}
R(s) = 0;
for ((x, y) ∈ A) { // Инициализация представления дерева дуг
  L(x, y) = ∞, P(x, y) = location(x, y);
}
for (y ∈ s') { // Начальный узел дерева дуг
  L(s, y) = w(s, y), Q += (s, y);
}
while (Q ≠ ∅) { // Расширение дерева дуг
  (x, y) = Q--; r = L(x, y), R(y) = r, P(y) = x;
  if (y == f) break; // Прерывание цикла при достижении цели
  for ((z ∈ y') && (y ∈ cont(y, z)))
    if (L(y, z) < ∞) {
      L(y, z) = r + w(y, z), P(y, z) = location(x, y);
      Q += (y, z);
    }
}

```

Недостаток алгоритма поиска на основе очереди дуг – дополнительная память порядка $O(|M|) - O(|N|)$. Однако свобода назначения ограничений на единственном экземпляре графа позволяет решать задачи поиска оптимальных путей перемещения разнородных подвижных единиц в реальном времени.

Список цитированных источников

1. Ревотюк, М.П. Реляционные модели задач оптимизации управления на интерпретируемых сетях/М.П.Ревотюк, З.А.Чан//Проблемы проектирования и производства радиоэлектронных средств: Сб. Материалов III Межд. НТК: в 2-х томах. – Новополюк: ПГУ, 2004. – Том 2. – С. 139-141.
2. Кормен, Т. Алгоритмы: построение и анализ: [пер. с англ.] / Кормен Т., Лейзерсон Ч., Ривест Р. – М.: МЦМНО, 2002. – 960 с.
3. Ревотюк, М.П. Поиск кратчайших путей на графах полиморфных сетей с ограничениями / М.П.Ревотюк, И.Ю.Дарадкех, В.А. Кирейчук // Известия Белорусской инженерной академии. – 2004. – № 1(17)/1. – С. 126-129.

УДК 519.872

О НАХОЖДЕНИИ ОЖИДАЕМЫХ ДОХОДОВ В СИСТЕМАХ НМ-СЕТИ С ПРИОРИТЕТНЫМИ ЗАЯВКАМИ

Бекиш Н.В., Кутурко О.М.

*УО «Гродненский государственный университет имени Я.Купалы», г. Гродно
Научный руководитель: Матальцкий М.А., доктор физ.-мат. наук, профессор*

Рассмотрим замкнутую сеть, в которой циркулируют K_1 заявок первого типа и K_2 заявок второго типа, причем заявки не могут менять свой тип. Система S_i содержит m_i параллельных линий обслуживания. Однотипные заявки, стоящие в очереди некоторой системы массового обслуживания (СМО), выбираются на обслуживание в произвольном порядке, например, FIFO. Заявки первого типа имеют абсолютный приоритет по отношению к заявкам второго типа. В данном случае это будет означать выполнение двух условий: а) если в момент освобождения линии некоторой СМО после обслуживания заявки в ее очереди имеются приоритетные заявки, то любая из них занимает освободившуюся линию; б) если в систему обслуживания, все линии которой заняты обслуживанием, но не только приоритетных заявок, поступает приоритетная заявка, то она вытесняет неприоритетную заявку с одной из линий и начинает обслуживаться этой линией; вытесненная заявка становится в очередь рассматриваемой СМО.

Состояние сети в данном случае характеризуется вектором

$$(k, t) = (k_{11}, k_{12}; k_{21}, k_{22}; \dots; k_{n1}, k_{n2}; t),$$

где k_{ic} – число заявок типа c в i -й СМО, $\sum_{i=1}^n k_{i1} = K_1$, $\sum_{i=1}^n k_{i2} = K_2$.

Пусть интенсивности обслуживания заявок в момент времени t $\mu_{ic}(k_{ic}(t))$ в системе S_i зависят от числа заявок в этой системе, $i = \overline{1, n}$, $c = 1, 2$. И будем предполагать, что в момент времени t выполняется следующее условие:

$$k_{i1}(t) < m_i, \quad i = \overline{1, n}, \quad c = 1, 2. \quad (1)$$

В этом случае имеем:

$$\varepsilon_{i1}(k_{i1}(t)) = \min\{k_{i1}(t), m_i\} = k_{i1}(t), \quad i = \overline{1, n},$$

$$\varepsilon_{i2}(k_{i1}(t), k_{i2}(t)) = \begin{cases} k_{i2}(t), & k_{i1}(t) + k_{i2}(t) < m_i \\ m_i - k_{i1}(t), & k_{i1}(t) + k_{i2}(t) \geq m_i \end{cases} = \min\{k_{i2}(t), m_i - k_{i1}(t)\}, \quad i = \overline{1, n}.$$

Рассмотрим динамику изменения доходов некоторой системы S_i сети. Обозначим через $V_i(t)$ ее доход в момент времени t . Пусть в начальный момент времени доход системы равен $V_i(0) = v_{i0}$. Доход этой СМО в момент времени $t + \Delta t$ можно представить в виде $V_i(t + \Delta t) = V_i(t) + \Delta V_i(t, \Delta t)$, где $\Delta V_i(t, \Delta t)$ – изменение дохода системы S_i на интервале времени $[t, t + \Delta t)$. Условные вероятности событий, которые могут произойти за время Δt и изменения доходов системы S_i , имеют вид:

1) с вероятностью $\mu_{j1} k_{j1}(t) p_{ji} \Delta t + o(\Delta t)$ заявка первого типа перейдет после обслуживания из системы S_j в систему S_i , при этом доход системы S_i возрастет на величину $r_{ji}^{(1)}$, а доход системы S_j уменьшится на эту величину, $j \neq i$, где $r_{ji}^{(1)}$ СВ с математическим ожиданием (м.о.) $M\{r_{ji}^{(1)}\} = d_{ji}$, $i, j = \overline{1, n}$;

2) с вероятностью $\mu_{i1}k_{i1}(t)p_{ij}\Delta t + o(\Delta t)$ заявка первого типа перейдет после обслуживания из системы S_i в систему S_j , при этом доход системы S_i уменьшится на величину $R_{ij}^{(1)}$, а доход системы S_j увеличится на эту величину, $j \neq i$, где $R_{ij}^{(1)}$ СВ с м.о. $M\{R_{ji}^{(1)}\} = d_{ij}$, $i, j = \overline{1, n}$;

3) с вероятностью $\mu_{j2} \min\{k_{j2}(t), m_j - k_{j1}(t)\}p_{ji}\Delta t + o(\Delta t)$ заявка второго типа перейдет после обслуживания из системы S_j в систему S_i , при этом доход системы S_i возрастет на величину $r_{ji}^{(2)}$, а доход системы S_j уменьшится на эту величину, $i, j = \overline{1, n}$, $j \neq i$, где $r_{ji}^{(2)}$ СВ с м.о. $M\{r_{ji}^{(2)}\} = b_{ji}$, $i, j = \overline{1, n}$;

4) с вероятностью $\mu_{i2} \min\{k_{i2}(t), m_i - k_{i1}(t)\}p_{ij}\Delta t + o(\Delta t)$ заявка второго типа перейдет после обслуживания из системы S_i в систему S_j , при этом доход системы S_i уменьшится на величину $R_{ij}^{(2)}$, а доход системы S_j увеличится на эту величину, $i, j = \overline{1, n}$, $j \neq i$, где $R_{ij}^{(2)}$ СВ с м.о. $M\{R_{ji}^{(2)}\} = b_{ij}$, $i, j = \overline{1, n}$;

5) с вероятностью $1 - \sum_{\substack{i,j=1, \\ j \neq i}}^n [\mu_{i1}k_{i1}(t) + \mu_{i2} \min\{k_{i2}(t), m_i - k_{i1}(t)\}]p_{ij}\Delta t + o(\Delta t)$ на отрезке времени величиной Δt изменение состояния системы S_i не произойдет. Кроме того, за каждый малый промежуток времени Δt система S_i увеличивает свой доход на величину $r_i \Delta t$, где r_i – СВ с м.о. $M\{r_i\} = c_i$, $i = \overline{1, n}$.

Будем также считать, что вышеперечисленные СВ попарно независимы.

При фиксированной реализации процесса $k(t)$, усредняя по $k(t)$ с учетом условия нормировки $\sum_k P(k(t) = k) = 1$ и вводя обозначение $v_i(t) = M\{V_i(t)\}$, $i = \overline{1, n}$, для изменения ожидаемого дохода системы S_i получаем

$$v_i(t + \Delta t) = v_i(t) + M\{\Delta V_i(t, \Delta t)\} = v_i(t) + \left[\sum_{\substack{j=1 \\ j \neq i}}^n \mu_{j1} N_{j1}(t) p_{ji} d_{ji} - \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{i1} N_{i1}(t) p_{ij} d_{ij} - \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{i2} M \min\{k_{i2}(t), m_i - k_{i1}(t)\} p_{ij} b_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{j2} M \min\{k_{j2}(t), m_j - k_{j1}(t)\} p_{ji} b_{ji} + c_i \right] \Delta t + o(\Delta t).$$

где $N_{i1}(t)$ – среднее число заявок, первого типа соответственно (ожидающих и обслуживающихся) в системе S_i в момент времени t , $i = \overline{1, n}$. Далее, переходя к пределу при $\Delta t \rightarrow 0$, получим неоднородные линейные ОДУ первого порядка

$$\frac{dv_i(t)}{dt} = \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{j1} N_{j1}(t) p_{ji} d_{ji} - \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{i1} N_{i1}(t) p_{ij} d_{ij} - \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{i2} M \min\{k_{i2}(t), m_i - k_{i1}(t)\} p_{ij} b_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{j2} M \min\{k_{j2}(t), m_j - k_{j1}(t)\} p_{ji} b_{ji} + c_i.$$

Задав начальные условия $v_i(0) = v_{i0}$, $i = \overline{1, n}$, можно найти ожидаемые доходы систем сети.

Полагая, что $M \min\{k_{i2}(t), m_i - k_{i1}(t)\} = \min(N_{i2}(t), m_i - N_{i1}(t))$, получаем следующую систему уравнений:

$$\begin{aligned} \frac{dv_i(t)}{dt} = & \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{j1} N_{j1}(t) p_{ji} d_{ji} - \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{i1} N_{i1}(t) p_{ij} d_{ij} + \\ & - \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{i2} \min(N_{i2}(t), m_i - N_{i1}(t)) p_{ij} b_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{j2} \min(N_{j2}(t), m_j - N_{j1}(t)) p_{ji} b_{ji} + c_i. \end{aligned} \quad (2)$$

Можно показать, что если сеть функционирует так, что в среднем в ее системах не наблюдается очередей, то $N_{i1}(t)$ и $N_{i2}(t)$ удовлетворяют системам ОДУ:

$$\frac{dN_{i1}(t)}{dt} = \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{j1} N_{j1}(t) p_{ji} - \mu_{i1} N_{i1}(t), \quad (3)$$

$$\frac{dN_{i2}(t)}{dt} = \sum_{\substack{j=1 \\ j \neq i}}^n \mu_{j2} N_{j2}(t) p_{ji} - \mu_{i2} N_{i2}(t), \quad i = \overline{1, n}. \quad (4)$$

Решив системы уравнений (3), (4), (2), можно найти ожидаемые доходы систем сети, зависящие от времени.

Список цитированных источников

1. Матальцкий, М.А. Системы и сети массового обслуживания: анализ и применение / М.А. Матальцкий, О.М. Тихоненко, Е.В. Колузаева. – Гродно: ГрГУ, 2011. – 817 с.

УДК 51-77

О МОДЕЛИРОВАНИИ АДАПТИВНОЙ СЕРВИСНОЙ СИСТЕМЫ СЛУЖБЫ ТЕЛЕКОММУНИКАЦИИ

Болтromeюк А.И.

УО «Белорусский государственный университет информатики и радиоэлектроники», г. Минск

Научный руководитель – Поттосина С.А., доцент, кандидат физ.-мат. наук

Сегодня перед телекоммуникационными предприятиями стоит задача поддержки конкурентоспособности, в первую очередь, за счет снижения издержек и сохранения мощности. Сервисная служба на таком предприятии играет одну из основных ролей, следовательно, её работа должна быть максимально эффективной. Для обеспечения эффективности сервиса необходимо учитывать непредсказуемость потоков заявок клиентов, что, с точки зрения проектирования сервиса, достаточно трудоёмко. Поэтому проектировщики сервиса нуждаются в удобном и мощном инструментарии, позволяющем экспериментальным способом проверить оптимальность конфигурации сервисной системы.

Одним из способов проведения тестирования конфигурации оборудования является использование программы, имитирующей поведение сервисной системы в различных ситуациях. В результате задача сводится к построению математической модели сервиса, проектированию имитационной модели и реализации её с помощью программных средств.

В данной статье рассматривается двухканальная адаптивная СМО с дополнительным прибором. Итак, пусть имеются два простейших потока заявок с интенсивностями λ_1 и λ_2 . Для обслуживания поступающих заявок имеются три прибора. Длительность обслуживания каждым прибором распределена по экспоненциальному закону: первый прибор обслуживает заявки только первого типа с интенсивностью μ_1 , второй – заявки только второго типа с интенсивностью μ_2 , третий, вспомогательный прибор, может обслуживать как заявки первого типа с интенсивностью μ'_1 , так и заявки второго типа с интенсивностью μ'_2 .

В системе можно выделить четыре различные структуры. В структуре S_0 вспомогательный прибор отключен, заявки первого типа обслуживаются первым прибором, заявки второго типа – вторым. В структуре S_1 вспомогательный прибор подключен к обслуживанию заявок первого типа вместе с первым прибором. Заявки второго типа обслуживаются только вторым прибором. В структуре S_2 , наоборот, вспомогательный прибор подключен к обслуживанию заявок второго типа вместе со вторым прибором, заявки первого типа обслуживаются только первым прибором. Наконец, в структуре S_3 вспомогательный прибор долю времени подключен к обслуживанию заявок первого типа и долю времени – к обслуживанию второго типа.

В зависимости от параметров λ_i , μ_i , μ'_i ($i=1,2$) та или иная структура (в смысле некоторого критерия оптимальности) будет лучше или хуже другой структуры, поэтому очевидна и цель адаптации – с течением времени «выйти» на лучшую структуру. В качестве критерия оптимальности выберем средние суммарные потери от простоя заявок в очереди и потери в связи с амортизацией постоянного прибора.

С самого начала построения имитационной модели нужно определить, какие свойства реальной системы должны быть зафиксированы в виде параметров, а какие могут изменяться в ходе моделирования, то есть служить переменными. В случае сервисного обслуживания на предприятии телекоммуникации, переменными будут мощность потока заявок и время обслуживания, а параметрами — стоимость амортизации приборов и стоимость простоя заявки в очереди. При моделировании основное внимание, как правило, уделяется состоянию переменных в различные моменты времени.

Параметры модели:

- стоимость амортизации приборов – C_1 ;
- стоимость простоя в очереди – C_2 .

Управляемые переменные:

- мощность первого прибора, μ_1 ;
- мощность второго прибора, μ_2 ;
- мощность третьего прибора при обработке заявки первого типа, μ^*_1 ;
- мощность третьего прибора при обработке заявки второго типа, μ^*_2 ;
- интенсивность потока заявок первого типа, λ_1 ;
- интенсивность потока заявок второго типа, λ_2 .

Неуправляемые параметры:

- время обслуживания t_j ;
- промежуток времени между поступлением заявок в систему T_j .

Для разработки программной модели исходная система должна быть представлена как стохастическая система массового обслуживания. Это можно объяснить следующим: информация от внешней среды поступает в случайные моменты времени, длительность

обработки различных типов информации может быть в общем случае различна. Таким образом, внешняя среда является генератором сообщений. А комплекс вычислительных устройств – обслуживающими устройствами.

В имитационной модели источник заявок подает на вход буферной памяти независимые друг от друга сообщения. Закон появления сообщений – пуассоновский поток. Поток сообщений обычно имитируется моментами времени, отображающими появление очередного сообщения в потоке.

В буферной памяти сообщения записываются «в навал» и выбираются по одному в обслуживающий аппарат по принципу FIFO/LIFO, где распределяются, в зависимости от вида и указаний блока синхронизации, по обслуживающим приборам. Длительность обработки одного сообщения в ОП распределена по экспоненциальному закону.

Блок сбора статистики собирает статистику по каждому из объектов модели. Блок синхронизации показывает, когда и в какое время будут активизированы те или иные фрагменты модели.

Буферная память – блок программы, предназначенный для хранения данных, ожидающих обработки в других компонентах. В данной модели она применяется для хранения информации об очередях и временных характеристиках модели и реализована в виде списка объектов различных заявок.

Обслуживающий аппарат отвечает за распределение потока заявок, входящих в СМО, по конкретным приборам. Его функционирование зависит от данных, полученных из блока синхронизации. Для имитации работы обслуживающего прибора необходимо реализовать выработку длительности обслуживания требования. Эти интервалы могут моделироваться с помощью генератора, функционирующего по принципу работы источника заявок.

Задача блока статистики заключается в накоплении численных значений, необходимых для вычисления статистических оценок, заданных параметров работы моделируемой системы, показателях её эффективности: среднее время ожидания в очереди, среднее значение длины очереди, коэффициент загрузки, вероятность выбора структуры, издержки на обслуживание. В приведенной модели блок статистики также отвечает за определение интенсивности потоков заявок.

Если компонент-имитатор работы источника или буферной памяти обслуживающего аппарата имитируют работу отдельных устройств, то блок синхронизации имитирует алгоритм взаимодействия отдельных устройств системы. В данной модели он реализуется по событийному принципу. При событийном принципе состояние всех блоков имитационной модели анализируется лишь в момент появления какого-либо события. Момент поступления следующего события определяется минимальным значением из списка будущих событий, представляющего собой совокупность моментов ближайшего изменения состояния каждого из блоков системы.

Проверка адекватности данной модели проводилась на результатах 1000 экспериментальных запусков, которые продолжались по 1 минуте, и показала, что в 83,4% случаев результаты моделирования обладают благоприятными статистическими характеристиками. Таким образом, можно сделать вывод, что существует возможность использования данной модели для проектирования сервисных систем с высоким уровнем достоверности прогноза.

УДК 004.94: 669.162.275.2

О МОДЕЛИРОВАНИИ МЕТАЛЛУРГИЧЕСКИХ РАСПЛАВОВ

Валеева Я.Е.

*ФГБОУ ВПО «Магнитогорский государственный технический университет
им. Г.И. Носова», г. Магнитогорск, Россия*

Научный руководитель – Кочержинская Ю.В., к.т.н.

В условиях рыночной экономики одной из ключевых целей для металлургического предприятия является получение продукции наилучшего качества с наименьшими затратами. Для ее достижения необходимо определить оптимальный химический состав сырья. Однако во многих случаях экспериментальное решение задачи определения оптимального состава является либо чересчур затратным, либо невозможным при нынешнем уровне развития науки и технологии. Одним из таких случаев является, например, определение химического состава металлургического расплава. Важнейшие свойства металлургических расплавов, такие как прочность и ковкость, зависят от состава примесей. Изучение влияния конкретных видов примесей при заданных условиях позволит получать более качественный металл при наименьших издержках производства. Эмпирические способы исследования состава и свойств расплава не позволяют получить полностью достоверные сведения о расплаве, находящемся под воздействием высоких температур, кроме того, являются довольно дорогостоящими.

В то же время текущий уровень развития информационных технологий, в том числе и персональных компьютеров, позволяет производить моделирование разнообразных физических и химических процессов с достаточно хорошей точностью, причем при сравнительно небольших затратах. В связи с этим компьютерное моделирование реальных металлургических расплавов оказывается весьма актуальным. Такое исследование позволит понять и оценить на ионном уровне механизм и кинетику взаимодействия компонентов расплава, что в свою очередь сделает возможным проведение оперативного прогноза свойств расплава, что необходимо для своевременной корректировки его состава.

Объектами исследования при компьютерном имитационном моделировании металлургических расплавов являются неметаллические и металлические расплавы. К неметаллическим относятся шлаки, основу которых составляют соединения кремния, алюминия, марганца и кальция. К металлическим расплавам относятся чугун и сталь, основу которых составляют соединения железа. Между этими соединениями действуют силы ионной, ковалентной, металлической химической связи, а также ван-дер-ваальсовы силы межмолекулярного притяжения.

Для моделирования процессов в металлургических расплавах применяют близкие, но различные методы. Двумя основными методами компьютерного моделирования являются метод молекулярной динамики (МД) и метод Монте-Карло (МК).

Метод молекулярной динамики является теоретическим детерминистическим методом, что означает исключение стохастических взаимодействий. Отправной точкой метода является хорошо определенное микроскопическое описание системы. Причем приготовление «хороших» начальных данных (координат и скоростей) для сложных молекулярных структур является проблемой не менее важной и, как правило, более трудоемкой, чем получение продуктивных траекторий, на которых осуществляется анализ поведения

моделируемой системы. Сущность метода молекулярной динамики состоит в численном решении уравнений классической механики для получения траекторий движения некоторого ограниченного числа частиц, по которым могут быть определены дальнейшие характеристики расплавов, как статические, так и динамические, как равновесные, так и неравновесные.

Весь процесс моделирования методом МД можно разделить на 3 этапа:

- Инициализация;
- Достижение равновесия;
- Расчет характеристик.

1 Инициализация

В качестве входных данных для моделирования выступают состав расплава, его температура и плотность. На основе этих данных проводят инициализацию системы взаимодействующих частиц (определяют начальные положения частиц и их скорости).

2 Достижение равновесия

Основой метода молекулярной динамики является численное интегрирование уравнений движения системы, представляющих собой законы классической механики Ньютона:

$$\frac{d\vec{r}_i}{dt} = \vec{v}_i(t), \quad m_i \cdot \frac{d\vec{v}_i(t)}{dt} = \vec{F}_i,$$

где \vec{r}_i , \vec{v}_i , m_i – радиус-вектор, скорость и масса i -частицы соответственно, \vec{F}_i – равнодействующая сил, действующих на частицу со стороны всех остальных.

При интегрировании уравнений движения система релаксирует в равновесное состояние.

Для исключения влияния граничных эффектов используются периодические граничные условия Борна-Кармана, при которых система является псевдобесконечной за счет окружения расчетного куба бесконечным числом копий. Тогда имитируется взаимодействие каждой конкретной частицы со всеми остальными частицами бесконечной периодической системы.

На каждом шаге по времени вычисляются силы, действующие на частицу. Для описания сил межчастичного взаимодействия выбирают аппроксимирующий потенциал взаимодействия из соображений соответствия теоретических данных экспериментальным. Суммируются все силы, действующие на заданную частицу.

Также на каждом временном шаге происходит пересчет ускорений частиц по формулам, определяемым вторым законом Ньютона:

$$a_x = \frac{F_x}{m}, a_y = \frac{F_y}{m}, a_z = \frac{F_z}{m}.$$

На основе полученных ускорений производится пересчет координат частиц.

Пересчет сил, ускорений и расчет траекторий частиц продолжается до тех пор, пока система не достигнет равновесия. Система достигает равновесного состояния, когда равнодействующие сил, действующих на каждую частицу, так же как и ускорения частиц, становятся равными нулю.

3 Расчет характеристик

После достижения равновесного состояния производится расчет парных парциальных корреляционных функций (ППКФ). ППКФ $\rho(r_{ij})$ – функция, определяющая плотность вероятности расположения атомов j на расстоянии r от атомов i (т. е. характеризующая распределение атомов j -го сорта относительно атомов i -го сорта). То есть, для построения

данной зависимости определяется количество пар частиц заданных сортов i и j , встречающихся на каждом конкретном расстоянии друг от друга. Дальнейший анализ ППКФ позволит определять различные характеристики расплавов.

Моделированием бинарных, тернарных и многокомпонентных расплавов методом молекулярной динамики занимаются: Д.К. Белашенко [2, 3], А.А. Мирзоев, И.В. Мальцев, Б.Р. Гельчинский, Л.В. Скворцов, О.И. Бухтояров.

Метод Монте-Карло является основным стохастическим методом моделирования молекулярных систем на ЭВМ. Этот метод основывается на статистической оценке сгенерированной с помощью последовательности случайных чисел выборки.

Алгоритм моделирования методом Монте-Карло можно описать следующим образом:

1. Задается начальная точка x_0 в фазовом пространстве. Данный шаг аналогичен инициализации в методе МД.

2. Случайным образом выбирается новое состояние или конфигурация системы. Например, частица случайным образом перемещается с позиции x_0 на новую позицию x' внутри модельного куба.

3. Вычисляется вероятность перехода $W(x_0, x')$.

4. Генерируется равномерно распределенное случайное число от 0 до 1.

5. Если вероятность оказывается меньше сгенерированного в четвертом пункте числа, то за новое состояние принимают x_0 , иначе – x' .

6. Осуществляется переход к пункту 2.

Работы по моделированию систем методом Монте-Карло имеются у Д.К. Белашенко [2], О.И. Островского.

Основным преимуществом методов МК и МД является то, что по полученным траекториям частиц можно определить практически любые характеристики – как термодинамические (энергия, давление, энтропия), так и кинетические (коэффициенты диффузии, частоты колебаний атомов). Особенностью метода молекулярной динамики является то, что он позволяет получать динамические характеристики системы, тогда как метод Монте-Карло позволяет рассчитывать только равновесные характеристики системы. Точность вычислений при этом ограничивается, в основном, вычислительной мощностью ЭВМ, что является и преимуществом, и недостатком, в связи с тем, что текущий уровень развития вычислительной техники все еще не позволяет производить сверхточные расчеты, однако в ближайшее время это положение может исправиться.

Недостатком методов является сложность определения межчастичных потенциалов, проблема нахождения которых остается актуальной и по сей день. Функции, аппроксимирующие потенциалы межчастичного взаимодействия, являются полуэмпирическими, соответственно даже при применении теоретических методов компьютерного моделирования остается значительная зависимость от экспериментальных данных, что, очевидно, ограничивает точность расчетов.

Таким образом, область моделирования многокомпонентных металлургических расплавов представляется актуальной, в частности для производства чугуна, существенной альтернативы которому на сегодняшний день не существует.

Список цитированных источников

1. Хеерман, Д.В. Методы компьютерного эксперимента в теоретической физике. – М.: Наука, 1990.
2. Белашенко, Д.К. Компьютерное моделирование структуры и свойств некристаллических оксидов // Успехи химии. – 1997.
3. Белашенко, Д.К. Компьютерное моделирование некристаллических веществ методом молекулярной динамики/ Соросовский образовательный журнал. – 2001.

УДК 004.4

ПАРАМЕТРИЧЕСКАЯ МОДЕЛЬ ДЛЯ РАСЧЕТА НАПРЯЖЕННО-ДЕФОРМИРОВАННОГО СОСТОЯНИЯ КАРКАСА ИНФОРМАЦИОННОГО ЭКРАНА В СРЕДЕ «ANSYS»

Вашкевич Д.И.

*УО «Белорусский национальный технический университет», г. Минск
Научный руководитель – Напрасников В.В., к.т.н., доцент*

При проектировании конструкции информационных экранов выдвигаются противоречивые требования. С одной стороны, конструкция должна удовлетворять условиям прочности при воздействии на нее нагрузок со стороны оборудования, собственного веса, ветровых нагрузок и сейсмических воздействий. С другой стороны, современное состояние экономики выдвигает существенные требования по минимизации материалоемкости таких конструкций. Решение таких задач предполагает оптимизационную постановку, ядром которой должна выступать параметрическая модель конструкции. В работе описываются особенности построения такой модели на примере одного из информационных экранов (рис. 1 слева).

Построение ведется в рамках системы многодисциплинарных расчётов ANSYS. Конечно-элементная расчетная схема конструкции представлена на рис. 1 справа.

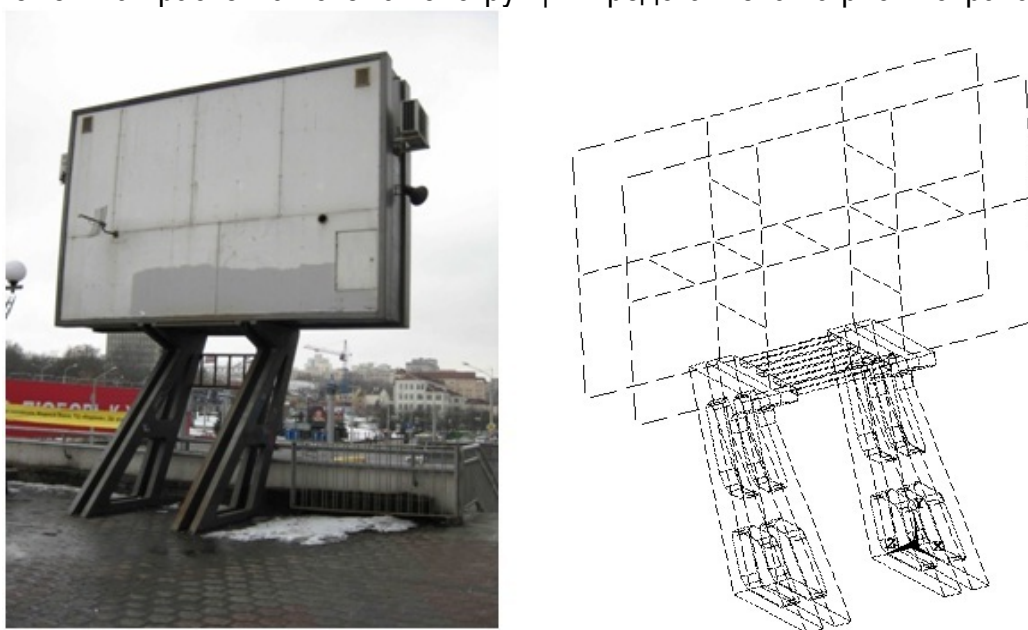


Рисунок 1 – Вид и конечно-элементная расчетная схема конструкции

Основными параметрами являются: толщина стенок опорных элементов, длины ребер жесткости, высоты опорных элементов, радиусы скруглений.

Конструкция состоит из двух существенно различных частей: массивное основание, для моделирования которого использовались конечные элементы типа shell 63, с толщиной стенки заданной величины delta; каркасная часть для размещения оборудования, для моделирования которой использовались конечные элементы типа beam 188 с сечением типа «полый» квадрат.

В качестве примера приведем изменение во времени координаты узла № 11343 по оси OX при тестовой динамической нагрузке (см. рис.2).

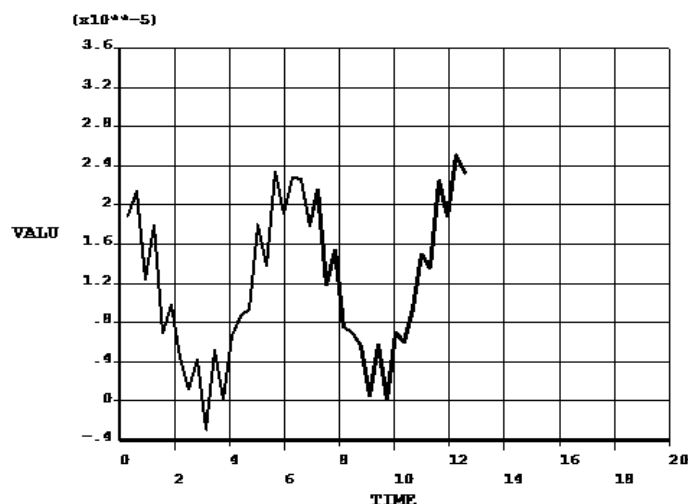


Рисунок 2 – Изменение во времени координаты узла № 11343 по оси OX

На основе разработанной модели была реализована задача по минимизации объема этой конструкции за выбором оптимальных параметров основания. В исходном варианте объем составлял 4,4745 м³, при этом эквивалентное напряжение по Мизесу достигало наибольшего значения 6,2 МПа. В оптимальном варианте объем составлял 0,16355 м³, при этом эквивалентное напряжение по Мизесу достигало наибольшего значения 2,67 МПа.

Выводы: в результате исследований выполнено построение геометрической модели кронштейна и каркаса информационного экрана, создана их параметрическая модель, позволяющая рассчитать конструкцию на прочность и жесткость в период эксплуатации. На основе этой модели построена оптимизационная модель для снижения материалоемкости конструкции при сохранении ее прочности и устойчивости. Материалоемкость конструкции удалось снизить более чем в 5 раз. При выполнении моделирования использовалась учебная версия комплекса, предоставленная компанией ANSYS.

Список цитированных источников

1. Напрасников, В.В. Конечно-элементное моделирование в ANSYS в режиме удаленного доступа к суперкомпьютеру «СКИФ»: учебно-методическое пособие / В.В. Напрасников, А.В. Бородуля, В.А. Кочуров. – Мн.: БНТУ, 2008. – 65 с.
2. Информационный программно-технический комплекс для дистанционного решения сложных прикладных задач на основе использования суперкомпьютерных систем / Напрасников В. В. [и др.] // Весці Нацыянальнай акадэміі навук Беларусі. Сер. фізіка-тэхнічных навук. – 2010. – № 2. – С.86-96.

УДК 519.872

АНАЛИЗ И ОПТИМИЗАЦИЯ МОДЕЛЕЙ ПРОГНОЗИРОВАНИЯ ДОХОДОВ ПРЕДПРИЯТИЙ С ИСПОЛЬЗОВАНИЕМ НМ-СЕТЕЙ

Ивановская Т.К.

*УО «Гродненский государственный университет им. Я.Купалы», г. Гродно
Научный руководитель: Матальцкий М.А., доктор физ.-мат. наук, профессор*

Сети массового обслуживания (МО) часто применяются в качестве моделей на предприятиях и заводах, которые реализуют свою продукцию покупателям и получают от этого некоторые ожидаемые доходы.

В данной работе предлагается использовать НМ (Howard-Matalytski)-сеть обслуживания при прогнозировании доходов предприятия от реализации продукции. Для этого надо решить следующие задачи:

- 1) разработать математическую модель прогнозирования доходов предприятия с помощью НМ-сетей;
- 2) провести сбор статических данных, необходимых для исследования модели;
- 3) разработать программное обеспечение для нахождения ожидаемых доходов.

Прогнозирование и нахождение ожидаемых доходов проводилось на основании данных двух предприятий: РУП «Гродненский завод медицинских препаратов» и ОАО «Скидельский сахарный комбинат».

Предложены следующие модели функционирования предприятий в виде сетей МО с центральной системой массового обслуживания (СМО). Для РУП «Гродненский завод медицинских препаратов» в качестве центральной СМО S_4 выступает непосредственно сам завод медицинских препаратов, а периферийными СМО являются крупнейшие покупатели – государственные склады разных городов: S_1 – УП «Фармация» г. Витебск; S_2 – УП «Фармация» г. Брест; S_3 – УП «Фармация» г. Гродно. Для ОАО «Скидельский сахарный комбинат» модель аналогична предыдущей, здесь в качестве центральной СМО S_6 выступает комбинат, а периферийными СМО являются крупнейшие покупатели: S_1 – УП «Бакалея» Брестского района; S_2 – УП «Бакалея» Минского района; S_3 – УП «Бакалея» Гомельского района; S_4 – УП «Бакалея» Гродненского района; S_5 – УП «Бакалея» Витебского района. При этом под заявкой понимается грузовой автомобиль, осуществляющий перевозку товара.

Для нахождения ожидаемых доходов необходимо решить следующую систему разностно-дифференциальных уравнений:

$$\begin{aligned} \frac{dv_n(k,t)}{dt} = & r_n(k) - \sum_{i=1}^n \mu_i u(k_i) v_n(k,t) + \\ & + \sum_{i=1}^{n-1} [\mu_i u(k_i) r(k + I_i - I_n, t) - \mu_n u(k_n) p_{ni} R(k - I_i + I_n, t)] + \\ & + \sum_{i=1}^{n-1} [\mu_i u(k_i) v_n(k + I_i - I_n, t) + \mu_n u(k_n) p_{ni} v_n(k - I_i + I_n, t)], \end{aligned} \quad (1)$$

где $v_n(k,t)$ – полный ожидаемый доход, который получает система S_n за время t , если в начальный момент времени сеть находится в состоянии k ; $u(x) = \begin{cases} 1, x > 0, \\ 0, x \leq 0, \end{cases}$ – функция Хевисайда; p_{ni} – вероятность поступления заявок из системы S_n в систему S_i , $i = \overline{1, n-1}$; $r(k + I_i - I_n, t)$ – размер дохода, который приносит система S_n , за время Δt когда сеть совершает переход из состояния (k, t) в состояние $(k + I_i - I_n, t + \Delta t)$ с вероятностью $\mu_n p_{ni} u(k_n) \Delta t + o(\Delta t)$ (то есть заявка переходит из системы S_n в систему S_i , $i = \overline{1, n-1}$); $R(k - I_i + I_n, t)$ – размер убытка, который получает система S_n , когда сеть совершает переход из состояния (k, t) в состояние $(k - I_i + I_n, t + \Delta t)$ с вероятностью $\mu_i u(k_i) \Delta t + o(\Delta t)$ (заявка осуществляет переход из системы S_i , $i = \overline{1, n-1}$ в систему S_n); $r_n(k)$ – доход, который получает система S_n за единицу времени в течение всего периода пребывания сети в состоянии k .

Система (1) может быть сведена к системе конечного числа линейных неоднородных ДУ с постоянными коэффициентами, которая в матричной форме может быть записана в виде:

$$\frac{dV_i(t)}{dt} = Q_i(t) + AV_i(t), \quad i = \overline{1, n-1}, \quad (2)$$

где $V_i^T(t) = (v_i(1, t), \dots, v_i(l, t))$ – вектор доходов системы S_i , $i = \overline{1, n-1}$, l – число состояний сети.

Решение системы (2) можно найти, используя прямой метод:

$$V_i(t) = e^{At} V_i(0) + \int_0^t e^{A(t-\tau)} Q_i(\tau) d\tau,$$

необходимым условием которого является вычисление матричной экспоненты e^{At} .

Для более быстрого и эффективного вычисления матричной экспоненты можно использовать специальный алгоритм «быстрого» вычисления, основанный на формуле:

$$e^{tA^{(q)}} = (e^{A^{(q)} \frac{t}{2^M}})^{2^M} \approx \left(\sum_{s=0}^N \frac{A^{(q)S}}{S!} \left(\frac{t}{2^M} \right)^S \right)^{2^M},$$

где $N \geq 1$, $M \geq 0$ – некоторые целые числа; $A^{(q)}$ – матрица чисел на q -ом интервале времени, $q = \overline{1, r}$.

Применяя данный метод к модели функционирования СМО (предприятия) и решая систему разностно-дифференциальных уравнений с помощью пакета математических вычислений Mathematica, можно получить графики ожидаемых доходов центральной СМО (предприятия).

УДК 510.647: 338.22.021.4

ИССЛЕДОВАНИЕ РАБОТОСПОСОБНОСТИ СОЦИАЛЬНО-ЭКОНОМИЧЕСКИХ СИСТЕМ

Коваленко И.В.

*УО «Белорусский государственный университет информатики
и радиоэлектроники», г. Минск*

Научный руководитель – Поттосина С.А., канд. физ.-мат. наук, доцент

Любой объект или процесс можно рассматривать как комплекс взаимосвязанных элементов. Банк, компания, государство, университет – все это социально-экономические системы, основные особенности которых – многокомпонентность и сложность организационной структуры, неопределенность внутреннего состояния. Данная неопределенность обусловлена тем, что основу любой социально-экономической организации составляют люди, действия которых не всегда поддаются управлению и контролю. Кроме того, каждая организация вынуждена действовать в условиях быстро изменяющейся внешней среды. Именно поэтому необходимо обратить пристальное внимание на теорию надежности, ориентированную на исследование условий работоспособности различного рода систем.

Для оценки работоспособности системы в теории надежности используются понятия работоспособности, надежности.

Работоспособность (availability, operability) – это состояние системы, при котором она способна реализовывать заданную функцию с заранее установленными требованиями в течение расчётного периода времени.

Под *безотказностью, надежностью (reliability)* понимается способность системы выполнять свои функции (полностью или частично) до некоторого момента времени при определенных условиях и при условии, что в начальный момент времени система была работоспособна.

Обычно при анализе надежности или безопасности объекта и его элементов исследуются два возможных состояния: корректная работа системы или отказ. На практике же таких состояний намного больше, они характеризуются особенностями функционирования системы. Для анализа таких промежуточных состояний работоспособности используются специальные математические модели - «системы с несколькими уровнями работоспособности».

Одно из направлений анализа надежности систем с несколькими уровнями работоспособности состоит в исследовании значимости или влияния отдельных элементов системы на различные уровни ее работоспособности (в англоязычной литературе Importance Analysis).

Проведенный анализ публикаций показал, что в настоящее время данное научное направление находится на этапе интенсивного развития, что объясняется перспективностью таких исследований. Для разработки методов анализа систем с несколькими уровнями работоспособности предлагаются различные математические подходы. Один из возможных подходов заключается в использовании математического аппарата логического дифференциального исчисления.

Системы с несколькими уровнями работоспособности являются математическим описанием функционирования исходного объекта. Представление объекта такой математической моделью позволяет исследовать его надежность на разных уровнях работоспособности.

Одним из перспективных, по мнению Э.Зио (E.Zio), подходов является использование при описании математической модели анализируемого объекта или процесса нескольких уровней работоспособности [1]. В англоязычной литературе такая математическая модель получила название *Multi-State System (MSS)*. Эта модель позволяет представить исходный объект как систему с несколькими уровнями работоспособности.

Использование MSS в качестве математической модели для анализа надежности впервые было предложено в 70-е годы [2]. В этой работе были определены основные теоретические понятия для анализа исходного объекта с помощью MSS. В частности, для MSS было введено понятие структурной функции (*Structure Function*). Эта функция устанавливает соответствие между состояниями отдельных элементов системы и ее надежностью. В этом случае исходные показатели объекта интерпретируются как исходные атрибуты структурной функции, а результирующие – как ее результирующие атрибуты.

Система, описывающая показатели реального экономического объекта, достаточно громоздка, в результате структурная функция обладает значительной вычислительной сложностью. Одним из подходов к снижению этой сложности является декомпозиция структурной функции. Суть декомпозиции заключается в разбиении функции на подфункции с меньшим числом атрибутов (на рис. 1 представлена структурная функция для системы оценки кредитоспособности физических лиц).

Аргументы таких подфункций будем интерпретировать как промежуточные атрибуты, а сами подфункции – как компоненты структурной функции. В свою очередь компоненты этой функции так же являются структурной функцией меньшей сложности и в свою очередь могут быть декомпозируемы. Компонент структурной функции реализует функцию *m*-значной логики. Математические методы логического дифференциального исчисления позволяют проанализировать изменение состояний работоспособности системы в зависимости от изменения состояний ее элементов. В рамках этого анализа вычисляется вероятность отказа или нахождения системы на различных уровнях работоспособности в зависимости от состояний работоспособности исследуемого элемента.

Актуальность этого анализа объясняется тем, что его результаты позволяют:

- определить влияние значений отдельных показателей на поведение объекта;
- вычислить устойчивость к изменениям отдельных показателей;
- обнаружить начальные условия, обеспечивающие развитие или ухудшение результирующих показателей экономического объекта и ориентированных на получения оптимальной стратегии развития;
- определить при проектировании оптимальную структуру объекта с учетом разнообразных критериев.

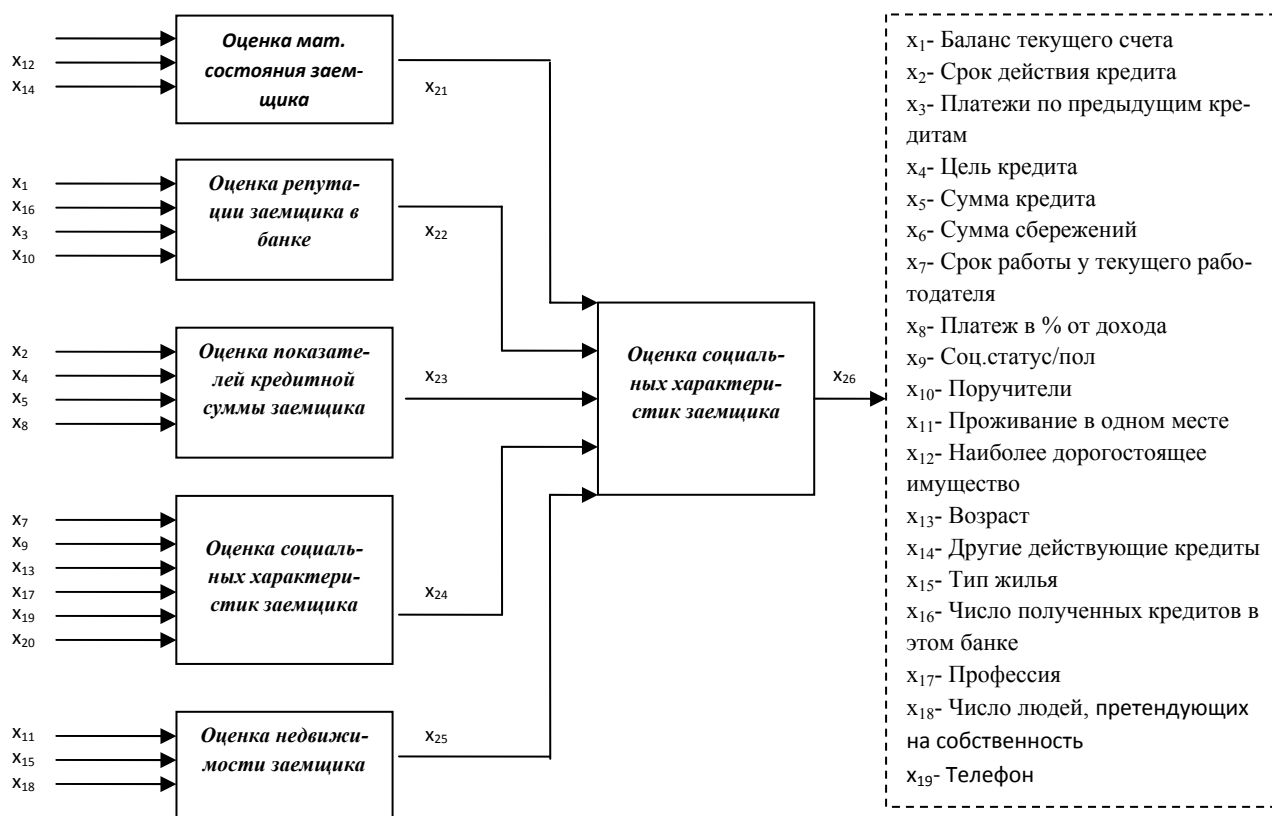


Рисунок 1 – Схематическое представление структурной функции, описывающей систему оценки кредитоспособности заемщика

Практическое использование математической модели MSS имеет ряд сложностей. К этим сложностям, во-первых, относят большую вычислительную нагрузку, связанную с размерностью структурной функции. Снятие этого затруднения авторы видят в использовании новых алгоритмов обработки структурной функции, ориентированных на регулярные и параллельно-конвейерные вычисления.

В заключение отметим, что в настоящее время математические модели MSS и алгоритмы их анализа успешно используются при оценке надежности и решении прикладных задач в области энергетики, обеспечения работы морского порта. Известны примеры использования MSS при оценке надежности медицинских и логистических систем, газо- и нефтепроводов, программного обеспечения и анализе систем сложной структуры.

Список цитированных источников

1. Zio, E. Reliability engineering: Old problems and new challenges // Reliability Engineering and System Safety. – 2009. – Vol.94. – № 2. – P. 125-141.
2. Andrews, J.D. Importance Measures for Non-Coherent-System Analysis / J.D. Andrews, S. Beeson // IEEE Trans. on Reliability. – 2003. – Vol.52. – №3. – P.301-310.

AB-INITIO МОДЕЛИРОВАНИЕ ЭЛЕКТРОННЫХ СВОЙСТВ НАНОСТРУКТУРНЫХ МАТЕРИАЛОВ С ИСПОЛЬЗОВАНИЕМ ПРОГРАММНОГО КОМПЛЕКСА VASP

Козлова О.А.

УО «Белорусский государственный университет информатики и радиоэлектроники», г. Минск

Левченко Н.В.

УО «Белорусский национальный технический университет», г. Минск
Научный руководитель – Нелаев В.В., доктор физ.-мат. наук, профессор

Проведено исследование применимости программного комплекса VASP для *ab-initio* моделирования электронных свойств оксидов редкоземельного оксида иттрия и дисульфида молибдена как активных материалов сенсорных наносистем. Исследована электронная плотность и рассчитана зонная структура.

ВВЕДЕНИЕ

Развитие технологии наносистем требует использования принципиально новых материалов, свойства которых определяются процессами, протекающими на атомном и молекулярном уровне, в нанослоях и нанобъемах. Расчет и оптимизация электронных свойств наносистем возможны только на основе компьютерных, требующих колоссальных ресурсов, вычислений на основе первопринципных *ab-initio* физических приближений.

В качестве инструмента моделирования в данной работе использовался программный пакет VASP, который предназначен для моделирования эволюции атомно-молекулярных и электронно-ядерных систем методами квантовой механики и молекулярной динамики. Взаимодействие между ионами и электронами моделируемой системы описывается с помощью псевдопотенциального подхода или метода присоединенных плоских волн. В программном комплексе VASP решается уравнение Шредингера для электронно-ядерной системы и на основе полученного решения проводится оценка ее полной энергии, сил и других параметров и других физических величин, определяющих электронные свойства. Релаксация атомов осуществляется согласно градиентно-сопряженному алгоритму, основанному на необходимой настройке гамильтониана системы посредством варьирования плотности заряда в системе [1].

Работа выполнялась посредством высокопроизводительной кластерной вычислительной системы СКИФ К-1000. Исследованы электронные характеристики двухкомпонентных соединений Y_2O_3 и MoS_2 , используемые в качестве материалов структурных элементов приборов сенсорики и наноэлектроники.

КРИСТАЛЛОГРАФИЧЕСКИЕ СВОЙСТВА Y_2O_3 И MoS_2

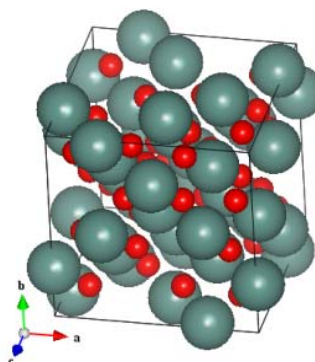
В кристаллической структуре положение атомов определяется кристаллофизическими параметрами, характеристикой и типом связи.

Для представления ячейки Y_2O_3 использовали кубическую кристаллографическую структуру пространственной группы – Ia-3 (№206). [2].

Элементарной ячейка Y_2O_3 представлена на рисунке:

Рисунок 1 – Кубическая ячейка Y_2O_3

Для описания ячейки MoS_2 использовалась тригональная кристаллографическая структура пространственной группы $R\bar{3}m$ (№160) [3].



Элементарная ячейка MoS_2 представлена на рис. 2:

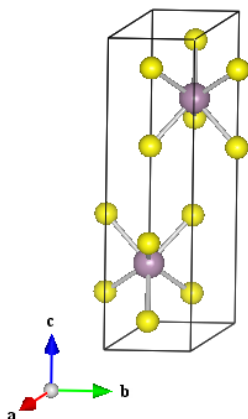


Рисунок 2 – Тригональная ячейка MoS_2

РЕЛАКСАЦИЯ Y_2O_3 И MoS_2

Для определения электронных свойств материалов Y_2O_3 и MoS_2 необходимо провести оптимизацию структуры.

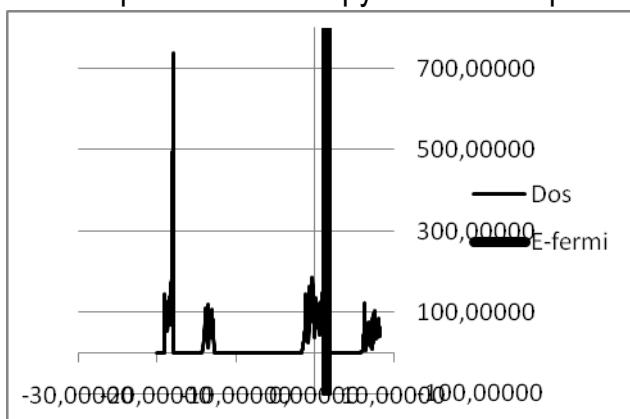
При проведении релаксации устанавливаются параметры для расчета сил, значений изменения объема и формы ячейки.

ЭЛЕКТРОННЫЕ СВОЙСТВА Y_2O_3 И MoS_2

В результате расчета электронных свойств используется метод присоединенных плоских волн (PAW-метода). PAW-метод состоит из набора основных функций и набора огибающих функций. Пространство разделено соответственно на две атомные сферы – атомную область и промежуточную область для связей. Частичные взаимопроникновения этих областей соответствуют границе раздела.

Провели расчет полной энергии и электронной плотности. Установили количество точек сетки для Y_2O_3 и MoS_2 , соответствующее 999, также определили минимальное и максимальное значение энергии. Для Y_2O_3 эти значения равны -20 и $8,3$ эВ соответственно, а для MoS_2 $-9,5$ и 9 эВ.

При моделировании электронных свойств получили электронную плотность — плотность вероятности обнаружения электронов в определенной точке пространства.



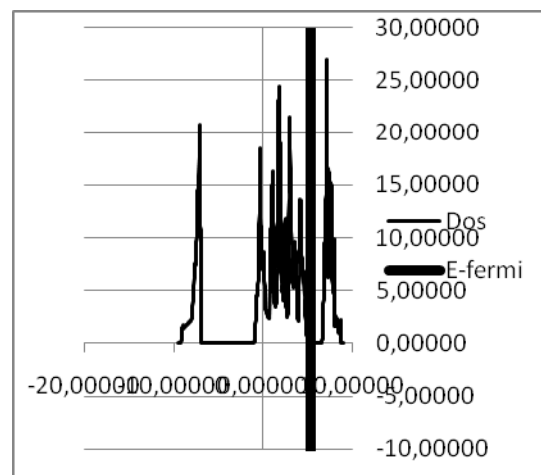
На рис. 4 представлена электронная плотность соединения MoS_2 :

Рисунок 4 – Электронная плотность MoS_2

При рассмотрении результатов электронной плотности определили значение ширины запрещенной зоны, для Y_2O_3 это значение составляет $4,31$ эВ, а для MoS_2 – $1,23$ эВ при уровне Ферми, равном для Y_2O_3 – $1,5298$ эВ, а для MoS_2 – $5,36$ эВ.

На рис. 3 представлена электронная плотность соединения Y_2O_3 :

Рисунок 3 – Электронная плотность Y_2O_3



ЗАКЛЮЧЕНИЕ

В процессе проведения работы рассмотрены используемые кристаллографические структуры, проведена релаксация и получены электронные плотности Y_2O_3 и MoS_2 . На основании моделирования электронных свойств построена электронная плотность элементарных ячеек Y_2O_3 и MoS_2 , определено значение ширины запрещенной зоны, которое для Y_2O_3 составляет 4,31 эВ, а для MoS_2 – 1,23 эВ. Можно отметить, что в исследуемых материалах отсутствует спиновая поляризация носителей, следовательно, они не обладают магнитными свойствами.

Список цитированных источников

1. Kresse, G. From ultrasoft pseudopotentials to the projector augmented-wave method: *Phys. Rev. B* / G. Kresse, J. Joubert. – 1999. – Vol. 62, No. 3. P. 1758-1765,
2. Baldinozzi, G. Rietveld refinement of two-phase Zr-doped Y_2O_3 : *Materials Science Forum* G. Baldinozzi, J.-F. Berar, G.P. Calvarin. – 1998. – 278. P. 680-685.
3. Schoenfeld, B. Anisotropic mean-square displacement (MSD) in single crystals of 2H- and 3R- MoS_2 : *Acta Crystallographica B*. – 1983. – 39. –P 404-407.

УДК: 51-77

МОДЕЛЬ КОНКУРЕНЦИИ КАФЕДР ЗА ТРУДОВОЙ РЕСУРС

Лысюк А.Н., Васильев Д.И.

УО «Брестский государственный технический университет», г. Брест
 Научный руководитель – Дереченник С.С., к.т.н., доцент

Основным типом деятельности вуза является образовательный процесс, для эффективного управления которым приходится решать ряд частных задач, среди которых формирование численности профессорско-преподавательского состава (ППС) кафедр является одной из основных [1]. Данную задачу можно отнести к планово-экономическим задачам, так как она заключается в эффективном распределении трудовых ресурсов вуза. Неэффективное и необоснованное распределение ставок ППС зачастую приводит к существенным проблемам, как на уровне отдельных кафедр, так и для вуза в целом, что негативно сказывается на образовательном процессе и, как следствие, на уровне подготовки конечных специалистов.

К числу существенных недостатков следует отнести рост среднегодовой нагрузки преподавателя и студентов из-за вполне естественного стремления кафедр в получении дополнительных ставок ППС.

Традиционная методика расчета, основанная на суммарной нагрузке вуза¹, может быть записана в виде следующей системы уравнений:

$$\begin{cases} H_{BVZ} = \sum_{i=1}^N H_i \\ \langle H_{II} \rangle = H_{BVZ} / \Pi_{BVZ} \\ \Pi_i = \left[\frac{H_i}{\langle H_{II} \rangle} \right]_{0,25} \end{cases}, \quad (1)$$

¹ В частности, данная методика расчета используется в Брестском государственном техническом университете.

где N – количество кафедр в вузе;

$H_i, i=1 \dots N$ – суммарная учебная нагрузка i -й кафедры;

$P_{\text{Вуз}}$ – общее количество ставок ППС, выделяемое на осуществление образовательного процесса и зависящее от количества студентов, обучаемых в вузе;

$H_{\text{Вуз}}$ – общевузовская суммарная учебная нагрузка;

$\langle H_{\Pi} \rangle$ – среднегодовая нагрузка преподавателя;

P_i – количество ставок, выделяемое кафедре, округленное до 0,25.

Существует ряд «общечеловеческих» причин, по которым кафедра может быть заинтересована в получении дополнительных ставок. Для того чтобы этого достичь, кафедра может пойти двумя путями: попросить дополнительную ставку у ректората или попытаться увеличить свою учебную нагрузку за счет внесения изменений в рабочие учебные планы специальностей – добавление зачетов, контрольных работ, рефератов по непрофилирующим и факультативным дисциплинам.

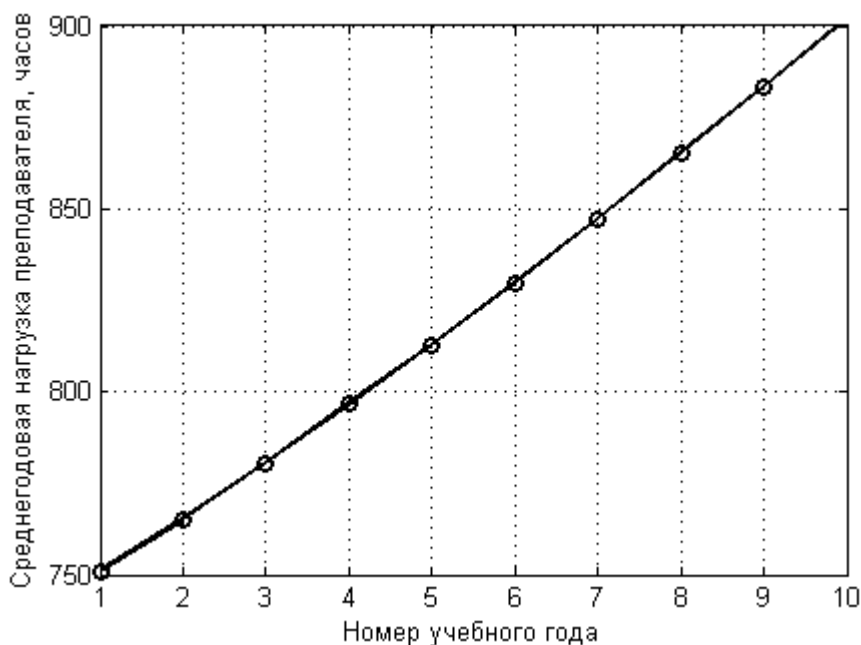
Таким образом, если кафедра стремится заполучить γ ставок, то ее учебная нагрузка возрастет по формуле 2.

$$H_i^j = H_i^{j-1} + \gamma_i^j \cdot \langle H_{\Pi}^{j-1} \rangle, \quad (2)$$

где верхний индекс обозначает год проведения расчета.

Однако в связи с тем, что общее количество ставок является фиксированным значением, зависящим от количества обучаемых студентов[2], то, с одной стороны это приводит к росту среднегодовой нагрузки преподавателя и студента (рисунок 1), а с другой стороны, между кафедрами происходит некая борьба за ставки (рисунок 2), в которой одна кафедра выступает в качестве «хищника», а другая в качестве «жертвы» [3].

Рисунок 1 – Динамика изменения среднегодовой нагрузки преподавателя по годам



Видно, что для одной кафедры наблюдается прирост численности ППС, а для другой кафедры наоборот наблюдается снижение численности ППС. Таким образом, если одна кафедра увеличивает свою учебную нагрузку, то выделение ставок на покрытие данной нагрузки производится за счет других кафедр, которые могут и не пересекаться с кафедрами-«хищниками» в рамках образовательного процесса.

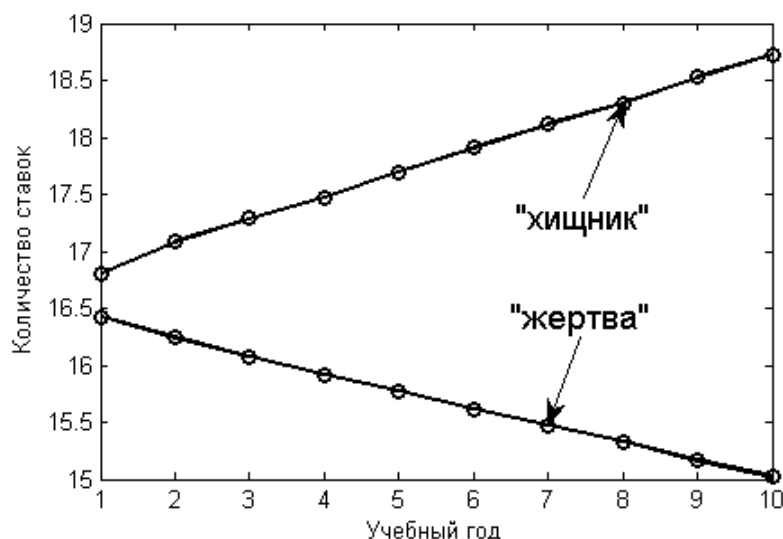


Рисунок 2 – Динамика изменения численного состава кафедр

Следовательно, традиционной методике присущи два серьезных недостатка: увеличение среднегодовой нагрузки преподавателя и

несправедливое перераспределение численности ППС по кафедрам, когда в выигрыше оказываются кафедры с «хищнической» стратегией.

Возможны следующие методы решения сформулированных проблем:

- в качестве основы для расчета использовать типовые, а не рабочие учебные планы;
- зафиксировать нормы времени;
- совершенствовать методику формирования численности ППС таким образом, чтобы необоснованные попытки завышения учебной нагрузки не приносили ожидаемых результатов.

На наш взгляд, самым прогрессивным является последний вариант. Очевидно, что такая методика должна учитывать все существенные параметры, такие как трудоемкость учебных планов специальностей, контингент студентов, обучаемых по специальности, существующие нормы времени и предоставлять объективную оценку сформированной нагрузке.

Список цитированных источников

1. Васильев, Ю.С. Экономика и организация управления вузом: учебник / Ю.С. Васильев, В.В. Глухов, М.П. Федоров, А.В. Федотов. – СПб.: Издательство «Лань», 1999. – 448 с.
2. Резник, С.Д. Управление кафедрой: учебник 2-е изд. / С.Д. Резник. – М.: ИНФА-М, 2005. – 635 с.
3. Шикин, Е.В. Математические методы и модели в управлении: 2-е изд., испр. / Е.В. Шикин, А.Г. Чхартишвили. – М.: Издательство «Дело», 2002. – 440 с.

УДК 519.872

О РЕШЕНИИ НЕКОТОРЫХ ЗАДАЧ ОПТИМИЗАЦИИ СМО, ПРИМЕНЯЕМЫХ ПРИ МОДЕЛИРОВАНИИ СИСТЕМ МЕЖБАНКОВСКИХ РАСЧЕТОВ

Монько В.Д., Матальцкий М.А.

УО «Гродненский государственный университет им. Я. Купалы», г. Гродно

При проектировании систем межбанковских расчетов (СМБР) требуется выбрать технические и программные средства, которые обеспечат требуемую производительность, надежность и безопасность системы с минимальными затратами на приобретение или разработку этих средств. Одним из подходов к решению этой задачи является построение математической модели СМБР на основе элементов теории массового обслуживания и выбор оптимального варианта на основе этой модели [1].

Часто необходимо знать, как будет вести себя оборудование в критических ситуациях. Рассмотрим несложную ситуацию, когда поток электронных сообщений (ЭС) поступает в систему, называемую «обработчик». При этом возникают следующие задачи:

а) какова должна быть интенсивность обработки ЭС μ^{onm} , чтобы среднее число ЭС в «обработчике» $N(t)$ уменьшилось до минимальных размеров за установленное время θ ?

б) найти оптимальную интенсивность обработки ЭС μ^{onm} , чтобы среднее число ЭС в «обработчике» уменьшалось до определенного значения N_n за установленный промежуток времени θ .

В качестве модели нашей системы можно применить систему массового обслуживания (СМО) типа M/M/1 с дисциплиной обслуживания FIFO и провести ее исследование и оптимизацию в переходном (нестационарном) режиме.

Среднее число ЭС (заявок) в системе в момент времени t определяется как

$$N(t) = \sum_{k=1}^{\infty} kP_k(t), \quad (1)$$

где $P_k(t)$ – вероятность того, что в момент времени t в системе находится k ЭС (заявок). Она имеет вид [2]:

$$P_k(t) = e^{-(\lambda+\mu)t} \left[\rho^{\frac{k-i}{2}} I_{k-i}(at) + \rho^{\frac{k-i-1}{2}} I_{k+i+1}(at) + (1-\rho)\rho^k \sum_{j=k+i+2}^{\infty} \rho^{\frac{j}{2}} I_j(at) \right], \quad (2)$$

где $\rho = \frac{\lambda}{\mu} < 1$, $a = 2\mu\rho^{\frac{1}{2}}$, $I_k(x)$ – обобщенная функция Бесселя первого рода порядка k , i – число ЭС в системе в момент времени $t = 0$. Значение функции $I_k(x)$ можно рассчитать по формуле [3]:

$$I_k(x) = \sum_{m=0}^{\infty} \frac{\left(\frac{x}{2}\right)^{k+2m}}{(k+m)!m!}, k \geq 0, I_k(x) = I_{-k}(x)$$

Тогда задача а) выглядит следующим образом:

$$\left\{ \begin{aligned} N(\theta, \mu) &= e^{-(\lambda+\mu)\theta} \sum_{k=0}^{\infty} k \left[\left(\frac{\lambda}{\mu}\right)^{\frac{k-i}{2}} I_{k-i}(2\theta\sqrt{\lambda\mu}) + \left(\frac{\lambda}{\mu}\right)^{\frac{k-i-1}{2}} I_{k+i+1}(2\theta\sqrt{\lambda\mu}) + \right. \\ &\quad \left. + \left(1 - \frac{\lambda}{\mu}\right) \left(\frac{\lambda}{\mu}\right)^k \sum_{j=k+i+2}^{\infty} \left(\frac{\lambda}{\mu}\right)^{\frac{j}{2}} I_j(2\theta\sqrt{\lambda\mu}) \right] \rightarrow \min_{\mu} \quad (3) \\ \lambda &< \mu \leq \mu_{\max}, \end{aligned} \right.$$

где μ_{\max} – заданное значение.

Ясно, что число заявок в любой однолинейной СМО уменьшается с ростом интенсивности их обслуживания. Поэтому функция $N(\theta, \mu)$ является убывающей по μ и решением задачи (3) является $\mu^{onm} = \mu_{max}$.

На рисунке 1 изображен график зависимости среднего числа ЭС в системе $N(\theta, \mu)$ при различных значениях числа ЭС в шлюзе в начальный момент времени.

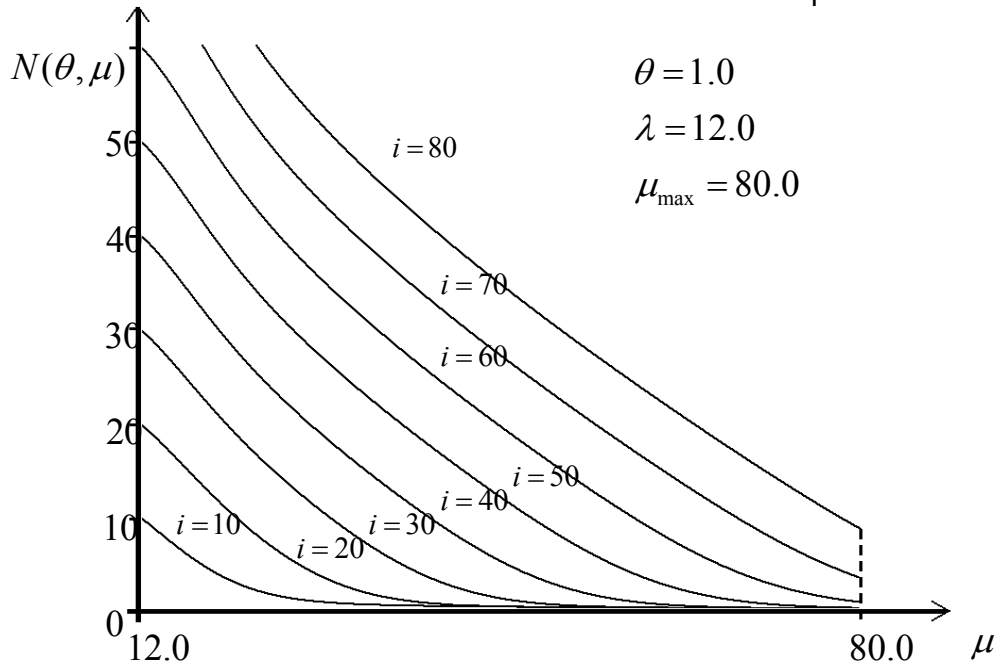


Рисунок 1 – График зависимости $N(\theta, \mu)$ от μ

Для решения задачи б) также воспользуемся соотношениями (1), (2). В данном случае μ^{onm} находится как решение нелинейного уравнения

$$N(\theta, \mu^{onm}) = e^{-(\lambda + \mu^{onm})\theta} \sum_{k=0}^{\infty} k \left[\left(\frac{\lambda}{\mu^{onm}} \right)^{\frac{k-i}{2}} I_{k-i} \left(2\theta \sqrt{\lambda \mu^{onm}} \right) + \left(\frac{\lambda}{\mu^{onm}} \right)^{\frac{k-i-1}{2}} I_{k+i+1} \left(2\theta \sqrt{\lambda \mu^{onm}} \right) + \left(1 - \frac{\lambda}{\mu^{onm}} \right) \left(\frac{\lambda}{\mu^{onm}} \right)^k \sum_{j=k+i+2}^{\infty} \left(\frac{\lambda}{\mu^{onm}} \right)^{\frac{j}{2}} I_j \left(2\theta \sqrt{\lambda \mu^{onm}} \right) \right] = N_n$$

Примеры найденных μ^{onm} при различных значениях параметров приведены в таблице 1.

Таблица 1 – Примеры найденных μ^{onm}

θ	λ	i	N_n	μ^{onm}
2.1	16.7	33	27.1	19.39
2.1	16.7	33	18.5	23.15
1.1	40.7	53	18.5	69.91

Список цитированных источников

- Карпук, А.А. О математическом моделировании системы межбанковских расчетов / А.А. Карпук, М.А. Маталыцкий // Устойчивое развитие экономики: состояние, проблемы, перспективы: материалы IV Международной научно-практической конференции. – Пинск: ПГУ, 2010. – С.21-25.
- Саати, Т. Элементы теории массового обслуживания и ее применения –М.: Советское радио, 1965. – 520 с.
- Википедия [Электронный ресурс] / Wikimedia Foundation. – Режим доступа: <http://www.wikipedia.org>. – Дата доступа: 10.10.2011.

УДК 004.942:51-74

МОДЕЛИРОВАНИЕ ПРОЦЕССА РАЗБРАСЫВАНИЯ УДОБРЕНИЙ С УЧЕТОМ УГЛА СХОДА С ДИСКА СРЕДСТВАМИ "МАТЕМАТИКА"

Пешков И.А.

УО «Гродненский государственный университет им. Я. Купалы», г. Гродно
Научный руководитель – Цехан О.Б., к. физ.-мат. наук, доцент

Ставится задача создания математической модели работы дискового разбрасывателя удобрений для нахождения таких оптимальных параметров его конструкции, при которых неравномерность распределения различных доз минеральных удобрений будет минимальной.

Процесс работы дискового разбрасывателя разбит на несколько фаз, и каждая фаза описывается соответственной моделью. Построена модель движения материальной частицы по плоскому диску с одной лопастью, которая описывает процесс. Её результатами являются начальная скорость и угол схода частицы с диска [1].

Математическая модель, описывающая сход материальной частицы под определенным углом с крутящегося диска, позволяет определить оптимальную скорость вращения диска.

Для завершающей фазы процесса работы диска построены модели свободного падения частицы после схода с разбрасывателя, а также вычисления плотности распределения.

Дифференциальное уравнение движения частиц при полете в указанном направлении имеет следующий вид:

$$x'' = -\frac{ky^E}{g}(x')^2 = -k_n(x')^2 \quad (1)$$

Для решения этого уравнения применялся пакет *Mathematica* версии 7. Получены зависимости дальности полета L от начальной скорости v_p и угла схода ϕ .

Для вычисления плотности рассева удобрений смоделирован процесс схода частицы с диска радиуса R при движении техники с поступательной скоростью с учетом угла схода ϕ и действия случайных возмущений.

Программа моделирует полет частицы в зависимости от начальной скорости, угла схода частиц с диска и рисует картину результата падения частиц с диска, а также график плотности рассева частиц удобрения (см. рисунок 2-3). Опишем более детально подсчет плотности рассева удобрений с учетом угла схода ϕ с диска (см. рисунок 1).

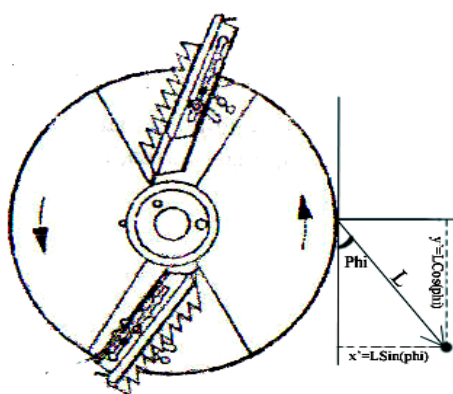


Рисунок 1 – Сход частицы под углом ϕ к касательной в точке схода частицы с диска

Дальность полета частицы, слетающей со скоростью v_p с диска, расположенного на высоте H над землей, определяется уравнением:

$$L = \frac{\log[km + \Gamma + \frac{1}{v_p}] \log[\frac{1}{v_p}]}{km}, \text{ где } \Gamma = \sqrt{2} \sqrt{\frac{H}{g}} \quad (2)$$

Рассчитываем дальность полета частицы L , затем находим координаты точки падения (x, y) , если бы частица сходила под углом 90 градусов (как на рис. 1.15), и поворачиваем ее на заданный угол относительно точки схода частицы с диска.

Чтобы повернуть частицу на заданный угол, будем использовать следующую систему уравнений:

$$\begin{cases} x' = x_0 + (x - x_0) \cos(\Phi) - (y - y_0) \sin(\Phi) \\ y' = y_0 + (x - x_0) \sin(\Phi) - (y - y_0) \cos(\Phi) \end{cases} \quad (3)$$

Данная система описывает поворот точки относительно заданной точки (x_0, y_0) . В нашем случае эта точка будет постоянно изменяться и высчитываться на каждом шаге при поступательном движении машины. Применив эту систему уравнений, мы получим новые координаты частицы.

Далее разбиваем область на сектора и подсчитываем количество частиц в каждом, после чего строим графики плотности в виде гистограммы и графика.

Примеры расчета координаты точки падения частицы удобрения, слетающей со скоростью v с диска, расположенного на высоте H над землей.

Пример 1:

Задано: радиус диска $R = 0.4$, скорость $v = 40$ схода частицы с диска, вращающегося со скоростью $w = 7$ и расположенного на высоте $H = 2$ над землей при поступательной скорости машины $V = 10$, угол схода $\phi = 10^\circ$.

Результат работы программы отображен на рисунках 2-3.

Рисунок 2 – Результаты движения точки на краю диска относительно оси x и y с углом схода $\phi = 10^\circ$

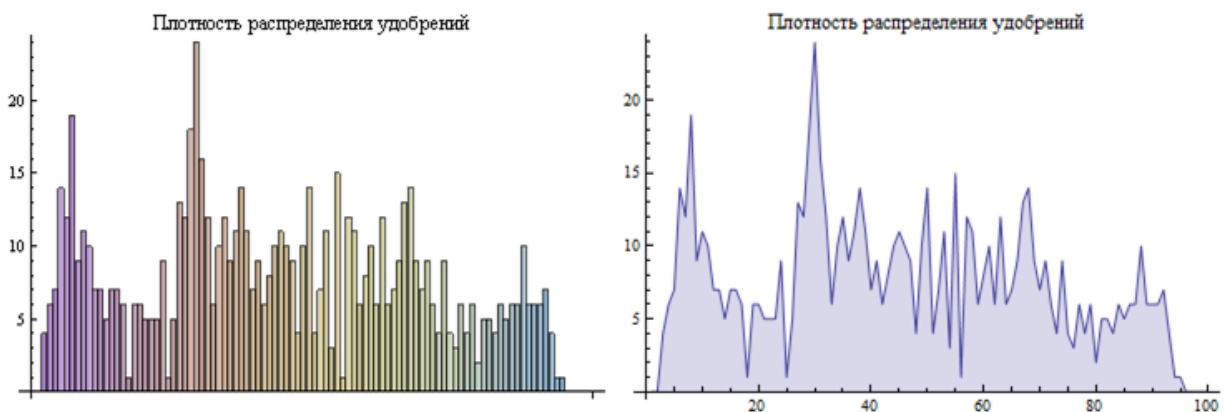
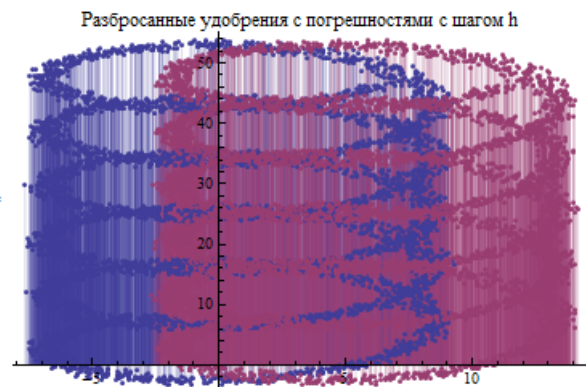


Рисунок 3 – Плотность рассева частиц удобрения в виде гистограммы и графика

Разработанное программное обеспечение позволяет моделировать весь процесс движения частиц при работе диска с одной лопастью и находить оптимальные параметры его работы. В дальнейшем эти результаты будут использоваться для определения оптимальных параметров конструкции диска.

На основе полученных в программе данных и в сравнении их с натурным экспериментом выявлено, что все модели адекватно отражают реальные результаты.

Список цитированных источников

1. Пешков, И.А. Моделирование работы дискового разбрасывателя в системе компьютерной алгебры MATHEMATICA / И.А. Пешков, О.Б.Цехан // Современные информационные компьютерные технологии mclT-2010: материалы II Международной научно-практической конференции / УО <<Гр. ун-т им. Я. Купалы>>. – Гродно, 2010. – 1 электр. компакт диск (CD-R). – 995 с. – Рус. – Деп. в ГУ <<БелИСА>> 24.05.2010 г., No. Д201019.

УДК 004.413

ГИБКАЯ МЕТОДОЛОГИЯ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Самойленко А.П.

*УО «Белорусский государственный экономический университет», г. Минск
Научный руководитель – Литвинец В.И., доцент, к.т.н, ст. научн. сотрудник*

Summary. To reduce labor costs, time and risks of software development, nowadays is increasingly being applied agile methodologies, in particular Scrum. It allows not only to solve all the problems mentioned above, but also improve team spirit and reduce the complexity of documentation.

Цель работы: обоснование целесообразности внедрения гибкой методологии разработки программного обеспечения, характеризуемой невысокими затратами и малой степенью риска для разработчика ПО.

Задача работы: создать комплексную характеристику гибкой методологии разработки ПО и на примере водопадной выявить преимущества предложенной методологии применительно к проекту по созданию сайта.

Существующая проблема: высокая стоимость разработки сайта, большие трудозатраты, высокая степень риска, сложность внесения изменений в техническое задание, длительность реализации проекта во времени.

Сведения о проекте: разработка сайта ведется по каскадной методологии. На проекте трудятся: 1 Project manager, 2 бизнес-аналитика, 1 дизайнер, 3 тестировщика, 8 разработчиков. Планируется сократить трудозатраты, время на разработку сайта, максимально удовлетворить клиента, решить проблему трудоёмкости ведения документации.

Каскадная модель (англ. waterfall model) — модель разработки программного обеспечения, в которой процесс выглядит как поток, последовательно проходящий фазы анализа требований, проектирования, реализации, тестирования, интеграции и поддержки.

Следуя *каскадной модели*, разработчик переходит от одной стадии к другой строго последовательно. Сначала полностью завершается этап «определение требований». Далее происходит переход к проектированию, в ходе которого создаются документы, подробно описывающие для программистов способ и план реализации указанных требований. На этапе реализации интегрируются отдельные компоненты, разрабатываемые различными командами программистов. После того как реализация и интеграция завершены, производится тестирование и отладка продукта, устраняются все недочёты,

появившиеся на предыдущих стадиях разработки. Далее программный продукт внедряется, обеспечивается его поддержка — внесение новой функциональности и устранение ошибок. Процесс разработки затянут, на любое изменение затрачивается колоссальное количество времени. Поскольку заказчик сможет оценить сайт только по завершении проекта, существует риск дополнительного согласования и, соответственно, роста затрат на исправление уже готового продукта. Изменение архитектуры в проекте тем сложнее, чем больше создано документов, а функциональная структура компании (отдел разработки, тестирования...) избыточна, ранее принятые решения и планы устарели.

Рассмотрим преимущества предлагаемой гибкой методологии (agile) и в частности Скрама (Scrum). Скрам — это каркасный подход к управлению проектом, в рамках которого вы можете строить свой agile- процесс, адаптируя его под ваши нужды.

Правила реализации:

1. Один человек в проекте (Product owner), уполномоченный принимать решения об очередности разработки частей функционала. Данная роль отводится бизнес-аналитикам.

2. Product Owner поддерживает список требований-пожеланий к продукту. Этот список сортируют по принципу «ценное сверху, менее ценное снизу». Такой список называется «Product Backlog».

3. При планировании короткой фазы проекта (спринта), команда — архитекторы, дизайнеры, разработчики, тестировщики – выбирает из Product Backlog ту верхнюю часть, которую реально начать и закончить за выделенный период. Заказчик и команда принимают удовлетворяющие всех решения: заказчик ожидает получить ценнейший набор функционала за короткое время, команда получает достоверный план.

4. Просмотр и обсуждение результатов - демонстрация. На выходе такой встречи мы имеем: уменьшенные риски за счёт доказательства работоспособности команды, технологий и требований; идеи по развитию сайта; доверие заказчика и команды.

5. Далее команда обсуждает серию экспериментов и задач, которые помогут более эффективно работать в течение следующего спринта. Это называется ретроспектива.

6. Для более слаженной работы членам команды придётся синхронизировать свои усилия и помогать друг другу. Делается это раз в день в течение 15 минут в присутствии всей команды. В ходе обсуждения участники могут просить помощи в решении проблем, которые они не могут устранить самостоятельно (падает сервер, пропадает интернет...) Эти проблемы помогает устранить так называемый ScrumMaster.

Таким образом, преимущества от внедрения данной методологии на проекте:

- удовлетворение клиента за счёт ранней и бесперебойной поставки ценного ПО;
- проведение изменений требований, даже в конце разработки (это повышает конкурентоспособность полученного продукта);
- частая поставка рабочего ПО;
- тесное, ежедневное общение заказчика с разработчиками на протяжении проекта;
- проектом занимаются мотивированные личности, которые обеспечены нужными условиями работы, поддержкой и доверием;
- постоянная адаптация к изменяющимся обстоятельствам;
- минимизация рисков;
- сокращение объемов документации.

В результате внедрения данной методологии планируется сократить команду до 1 Project manager'a, 1 Скрам мастера, 1 дизайнера, 2-х бизнес-аналитиков, 1 тестировщика и пяти разработчиков (т.е. сократить команду на двух тестировщиков и трёх разработчиков). При построении стратегического плана выявилось сокращение времени на разработку сайта на полгода (30%). Скрам внедрен на проект 2 месяца назад. Отставаний от плана не наблюдается, хотя объем функциональностей сайта заметно возрос с начала проекта.

УДК 004.4

ПАРАМЕТРИЧЕСКАЯ МОДЕЛЬ ДЛЯ РАСЧЕТА НАПРЯЖЕННО-ДЕФОРМИРОВАННОГО СОСТОЯНИЯ КОЛЕСА ОБОЗРЕНИЯ В СРЕДЕ «ANSYS»

Семашко Д.В., Плышевская О.В.

*УО «Белорусский национальный технический университет», г. Минск
Научный руководитель – Напрасников В.В., к.т.н., доцент*

При проектировании конструкций, функционирование которых связано с безопасностью людей, особое внимание уделяется адекватности соответствующих математических моделей. Одной из таких конструкций является колесо обозрения, которое в процессе своей эксплуатации может требовать выполнения текущих ремонтов.

В представленной работе описывается использование конечно-элементной системы ANSYS, которая в мире признана наиболее надежной и эффективной для решения задачи расчета напряженно-деформированного состояния при различных классах внешних воздействий, применительно к описываемому объекту.

Особое внимание при построении конечно-элементной модели этой конструкции уделялось возможности вариантных расчетов на этапе проекта, что достигается за счет введения параметризации. Вид объекта и конечно-элементная расчетная схема представлены на рисунке 1.



Рисунок 1 – Вид объекта и конечно-элементная схема

Основными параметрами являются толщина стенок опорных элементов, радиус колеса, толщина стенок стержней колеса, длина стержней.

На основе этой модели решены следующие задачи:

– расчет прочности и жесткости конструкции при статических взаимодействиях под действием собственного веса и веса кабинок с пассажирами;

– расчет прочности и устойчивости конструкции в период предполагаемого ремонта одной из опор без разбора конструкции;

– расчет динамической реакции колеса обозрения в момент предполагаемого землетрясения с магнитудами различных величин. В качестве примера приведем изменение во времени координаты узла № 24 по оси OX при землетрясении магнитудой 7.8 балла за период наблюдения 5 секунд, что соответствует участку с максимальной амплитудой ускорения на акселерограмме (см. рисунок 2).

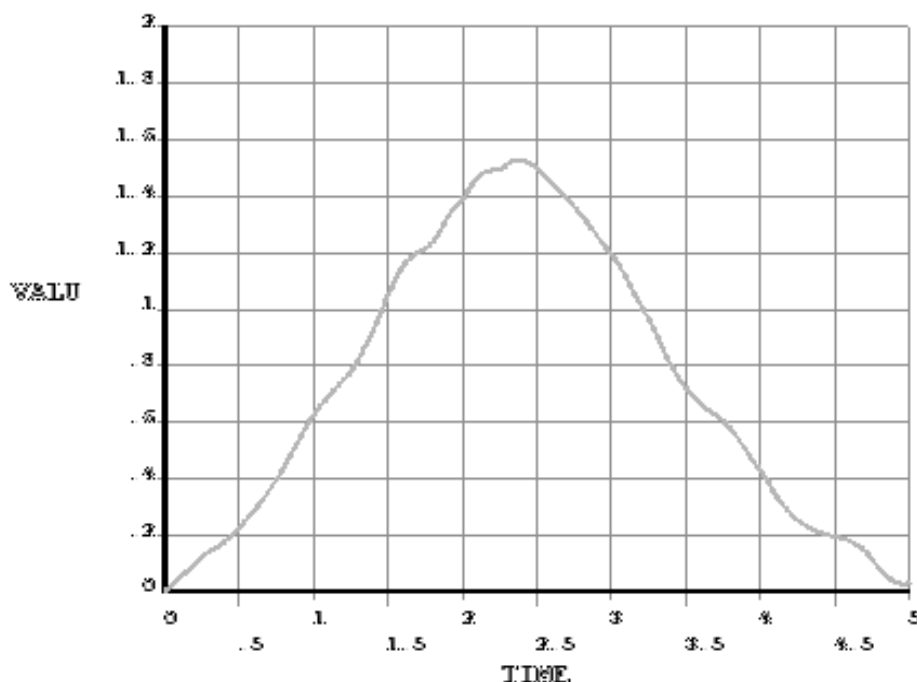


Рисунок 2 – Изменение во времени координаты узла № 24 по оси OX

Выводы: в результате исследований получена параметрическая модель колеса обозрения, позволяющая рассчитывать его на прочность и жесткость в период эксплуатации и ремонта; на основе этой модели может быть построена оптимизационная модель для снижения материалоемкости конструкции при сохранении ее прочности и устойчивости. При выполнении моделирования использовалась учебная версия комплекса, предоставленная компанией ANSYS.

Список цитированных источников

1. Напрасников, В. В. Конечно-элементное моделирование в ANSYS в режиме удаленного доступа к суперкомпьютеру «СКИФ»: учебно-методическое пособие / В.В. Напрасников, А.В. Бородуля, В.А. Кочуров – Мн.: БНТУ, 2008. – 65 с.
2. Информационный программно-технический комплекс для дистанционного решения сложных прикладных задач на основе использования суперкомпьютерных систем / Напрасников В. В. [и др.] // Весці Нацыянальнай акадэміі навук Беларусі. Сер. фізіка-тэхнічных наук. – 2010. – № 2. – С.86-96.

УДК 519.67

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ОЦЕНКИ КУМУЛЯНТНЫХ ХАРАКТЕРИСТИК СИГНАЛОВ

Середенко Е.И., Середенко Р.С.

УО «Белорусский государственный университет информатики и радиоэлектроники», г. Минск

Научный руководитель – Хоменок М.Ю., к.т.н., доцент

При решении практических задач возникает необходимость в оценивании спектров высших порядков некоторого процесса по имеющемуся конечному набору данных измерений. Наиболее рациональный метод для решения данного вопроса – это создание программного модуля для оценки кумулянтных и полиспектральных характеристик сигналов.

Существуют два основных класса методов оценивания полиспектров, а именно – обычные Фурье-методы и параметрические методы, которые базируются на той или иной априорной информации об исследуемом сигнале – его моделях (например, моделях авторегрессии, скользящего усреднения).

В данной работе рассматриваются обычные методы полиспектрального оценивания и их свойства. Их можно разбить на следующие два класса:

1) *класс косвенных (или непрямых) методов*, основанных на аппроксимации определения полиспектров:

$$S_n(f_2, \dots, f_n) = \int_{-\infty}^{+\infty} \dots \int \langle x(t), x(t + \tau_2), \dots, x(t + \tau_n) \rangle e^{-i2\pi \sum_{k=2}^n f_k \tau_k} d\tau_2 \dots d\tau_n,$$

и 2) *класс прямых, методов*, аппроксимирующего эквивалентное определение биспектра

$$S_n(f_2, \dots, f_n) = \lim_{T \rightarrow \infty} S_n(T, f_2, \dots, f_n) = \lim_{T \rightarrow \infty} \frac{1}{T} \left\langle X_T(f_2; t), \dots, X_T(f_n; t), X_T^* \left(\sum_{k=2}^n f_k; t \right) \right\rangle$$

где $X_T(f; t) = \int_{t-T}^t x(\tau) e^{-i2\pi f \tau} d\tau$ – T -финитное преобразование Фурье.

Методы оценивания спектральных кумулянтов предполагают оценивание сначала моментов (в спектральной или временной областях) всех порядков от первого до данного, а затем вычисление по ним оценок требуемых спектральных кумулянтов.

Поэтому практическая реализация и теоретический анализ таких оценок, в общем случае, представляют собой довольно сложные задачи.

В качестве среды разработки программной модели была выбрана среда MatLab.

Разработанное программное обеспечение состоит из двух частей. В первой части анализируемый сигнал задается математическим выражением. Вторая часть представляет собой вычислительный блок пакета Simulink, входящий в состав среды разработки MatLab.

Блок вычисления биспектра в пакете Simulink имеет ряд параметров: интервал дискретизации, время начала анализа, время окончания анализа, ширина окна, сдвиг окна.

Блок-схема алгоритма программной модели с вводом математического выражения исследуемого сигнала изображена на рисунке 1.

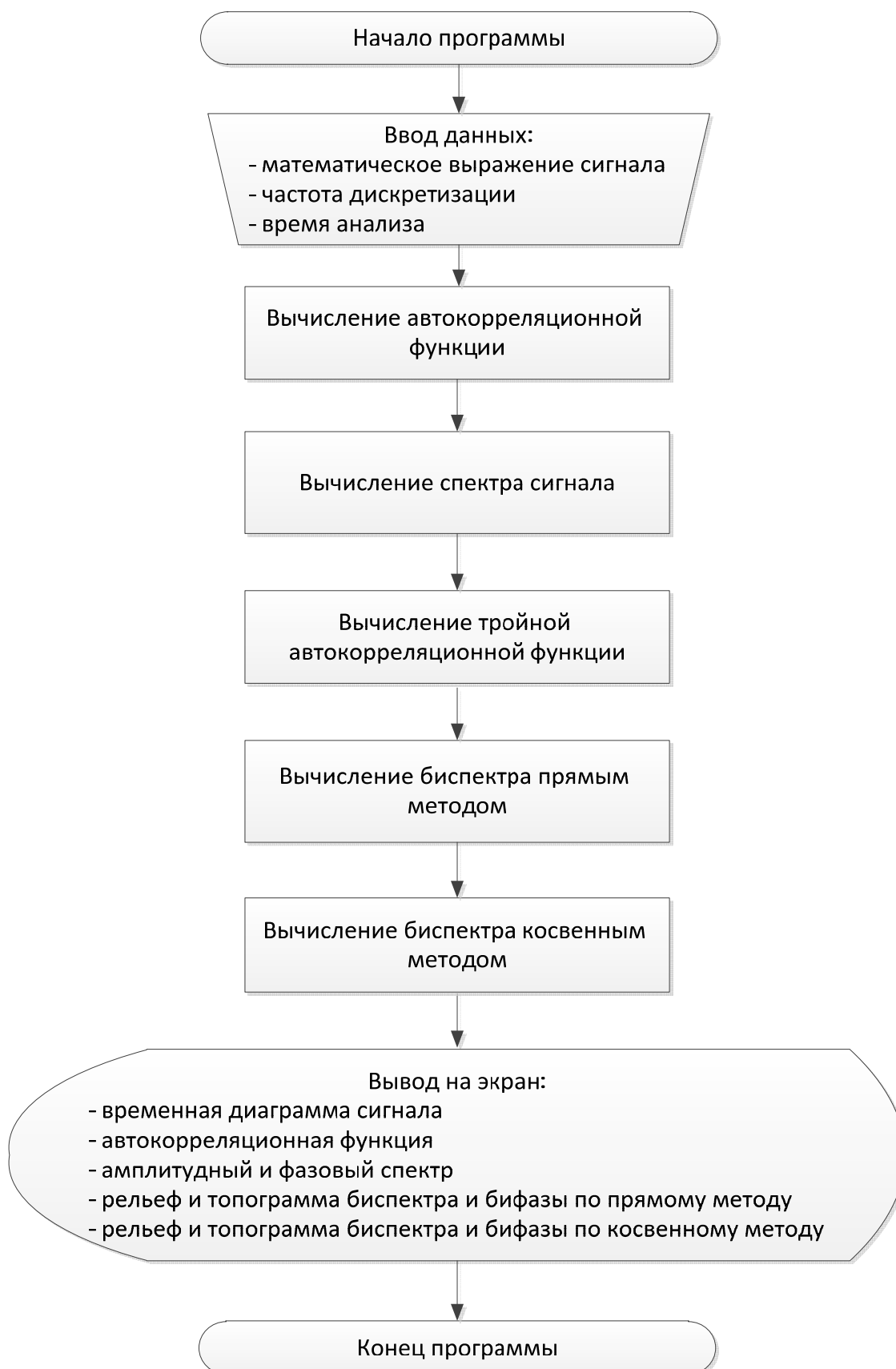


Рисунок 1 – Блок схема алгоритма программной модели

Блок-схема алгоритма функционального блока пакета Simulink изображена на рисунке 2.

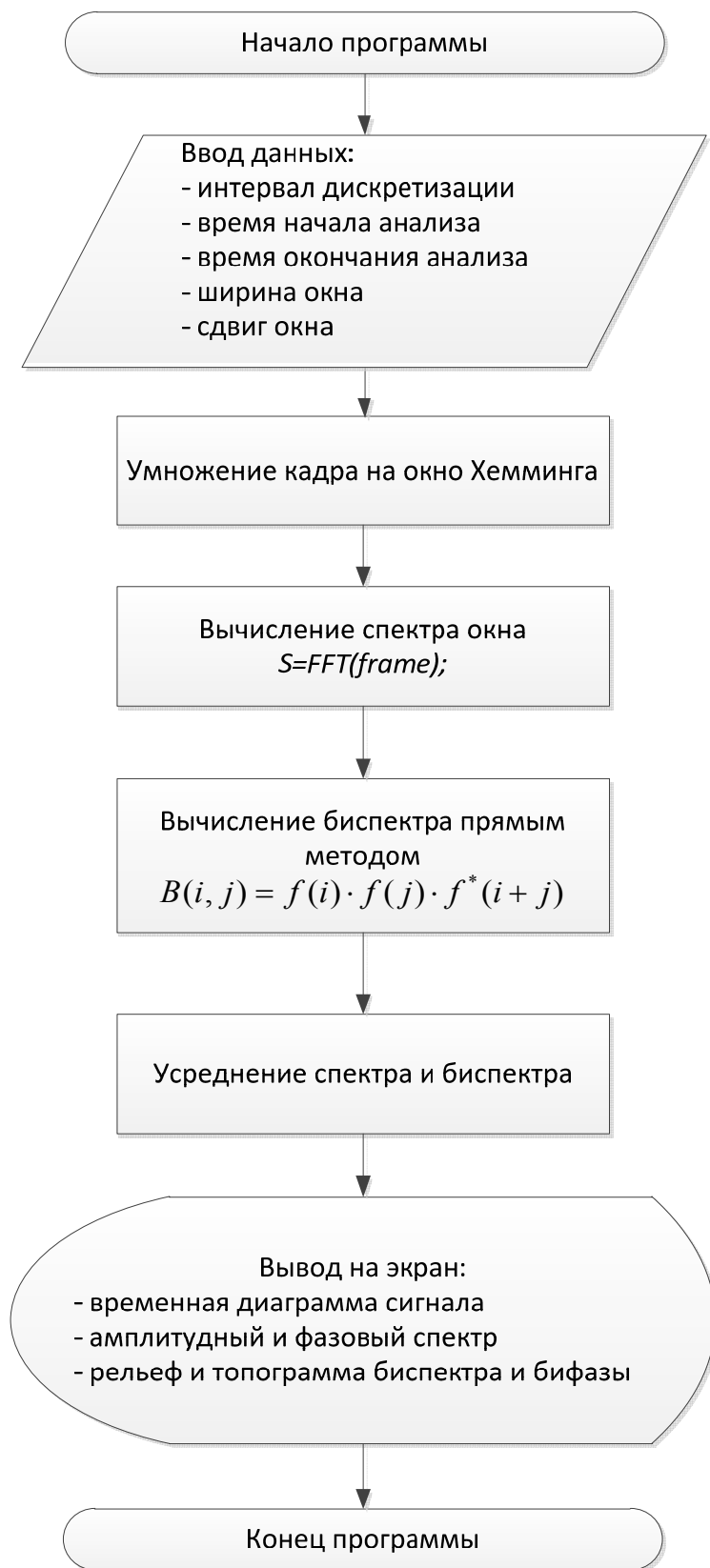


Рисунок 2 – Блок схема алгоритма функционального блока

Список цитированных источников

1. Бочков, Г.Н. Полиспектральный анализ и синтез сигналов: учебно-методический материал по программе повышения квалификации «Новые подходы к проблемам генерации, обработки, передачи, хранения, защиты информации и их применения» / Г.Н. Бочков, К.В. Горохов. – Нижний Новгород, 2007. – 113 с.

УДК 517.953

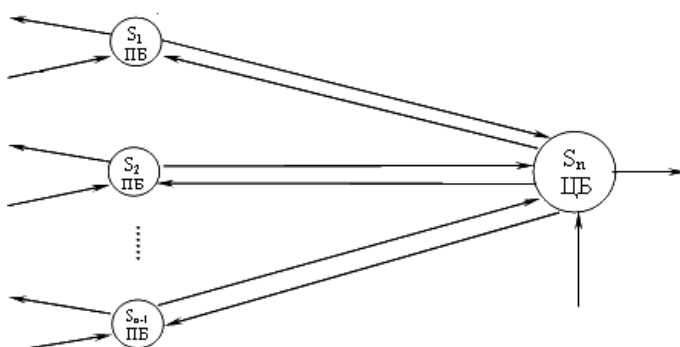
О ПРИМЕНЕНИИ НМ-СЕТЕЙ С ОГРАНИЧЕННЫМИ ВРЕМЕНАМИ ЗАЯВОК ПРИ ПРОГНОЗИРОВАНИИ ДОХОДОВ

Статкевич С.Э.

УО «Гродненский государственный университет имени Янки Купалы», г. Гродно

Сеть массового обслуживания (МО) представляет собой совокупность систем массового обслуживания (СМО), между которыми циркулируют заявки, переходя из одной СМО в другую. НМ (Howard-Matalytski)-сети являются расширением понятия сетей МО и позволяют находить ожидаемые доходы СМО сети. Заявка при переходе из одной СМО в другую приносит ей некоторый доход, а доход первой СМО уменьшается соответственно на эту величину. Они применяются в качестве моделей прогнозирования доходов различных объектов в экономике, транспортной логистике, страховании и др.

Рассмотрим банковскую сеть, изображенную на рис.1. На верхнем уровне банковской сети находится Центральный банк (ЦБ), ниже – крупные периферийные банки (ПБ) с их филиалами. Все межбанковские платежи проводятся Расчетным центром (РЦ) ЦБ с помощью банковской компьютерной сети через корреспондентские счета (КС), открываемые на балансе каждого банка. Каждый платеж оформляется в виде одного электронного платежного документа (ЭПД). Прием и обработка ЭПД осуществляются по мере их поступления.



Каждый платеж оформляется в виде одного электронного платежного документа (ЭПД). Прием и обработка ЭПД осуществляются по мере их поступления.

**Рисунок 1 - Открытая модель
банковской сети**

Каждый банк в течение дня видит состояния очередей своих платежей и может в случайный момент времени отозвать ЭПД из очереди ожидания средств по денежным переводам при наличии ошибочных реквизитов в ЭПД либо в случае, когда для его проведения нет средств. В этом случае при моделировании доходов целесообразно применить НМ-сети с ограниченными временами ожидания заявок в очередях, которые позволяют найти ожидаемые доходы от переходов между состояниями банковской сети, соответствующих поступлению платежей (заявок) между банками, а также оптимальные резервы банков [1].

Для ожидаемых доходов систем сети получена система разностно-дифференциальных уравнений

$$\frac{dv_i(k,t)}{dt} = - \left[\lambda + \sum_{j=1}^n (\mu_j \min(k_j, m_j) u(k_j) + \theta_j (k_j - m_j) u(k_j - m_j)) \right] v_i(k,t) + \sum_{j=1}^n \left\{ \lambda p_{0j} v_i(k + I_j, t) + [\mu_j \min(k_j, m_j) u(k_j) p_{j0} + \theta_j (k_j - m_j) u(k_j - m_j) q_{j0}] v_i(k, t) \right\} +$$

$$\begin{aligned}
 & + \sum_{j=1}^n \left\{ \left[\mu_i \min(k_i, m_i) u(k_i) p_{ij} + \theta_i (k_i - m_i) u(k_i - m_i) q_{ij} \right] v_i(k - I_i + I_j, t) + \right. \\
 & \left. + \left[\mu_j \min(k_j, m_j) u(k_j) p_{ji} + \theta_j (k_j - m_j) u(k_j - m_j) q_{ji} \right] v_i(k + I_i - I_j, t) \right\} + \\
 & + \sum_{\substack{c,s=1 \\ c,s \neq i}}^n \left[\mu_s \min(k_s, m_s) u(k_s) p_{sc} + \theta_s (k_s - m_s) u(k_s - m_s) q_{sc} \right] v_i(k + I_c - I_s, t) + \\
 & + \sum_{j=1}^n \left[\mu_j \min(k_j, m_j) u(k_j) p_{ji} r_{ij} (k + I_i - I_j, t) - \right. \\
 & \quad - \mu_i \min(k_i, m_i) u(k_i) p_{ij} r_{ji} (k - I_i + I_j, t) + \\
 & \quad + \theta_j (k_j - m_j) u(k_j - m_j) q_{ji} h_{ij} (k + I_i - I_j, t) \\
 & \quad \left. + \theta_i (k_i - m_i) u(k_i - m_i) q_{ij} h_{ji} (k - I_i + I_j, t) \right] + \\
 & + \lambda p_{0i} r_{0i} (k + I_i, t) - \mu_i \min(k_i, m_i) u(k_i) p_{i0} R_{i0} (k - I_i, t) - \\
 & - \theta_i (k_i - m_i) u(k_i - m_i) q_{i0} H_{i0} (k - I_i, t) + r_i(k), \quad i = \overline{1, n},
 \end{aligned}$$

где $k = (k, t) = (k_1, \dots, k_n, t)$ – вектор состояний сети, $k_i(t)$ – число заявок в системе S_i в момент времени t , $i = \overline{1, n}$; $v_i(k, t)$ – полный ожидаемый доход, который получает система S_i за время t , если в начальный момент времени сеть находится в состоянии $(k, 0)$; λ – интенсивность поступления простейшего потока заявок в сеть; m_i – количество линий обслуживания в системе S_i ; μ_i^{-1} , θ_i^{-1} – соответственно среднее время обслуживания в каждой линии и среднее время ожидания заявки в очереди в системе S_i ; p_{ij} – вероятность перехода заявки после обслуживания в системе S_i в систему S_j ; q_{ij} – вероятность перехода заявки в очередь системы S_j , если время ожидания в очереди системы S_i истекло; I_i – n -вектор с нулевыми компонентами, за исключением компоненты с номером i , которая равна 1; $r_i(k)$ – доход системы S_i в у.е. в единицу времени в течение времени пребывания сети в состоянии k ; $R_{i0}(k - I_i, t)$, $H_{i0}(k - I_i, t)$ – расходы системы S_i , когда сеть меняет свое состояние из (k, t) на $(k - I_i, t + \Delta t)$ соответственно из-за ухода заявки после обслуживания в ней во внешнюю среду и из-за ухода заявки из очереди этой СМО во внешнюю среду, которая не дождалась обслуживания в ней; $r_{0i}(k + I_i, t)$ – доход системы S_i , когда сеть меняет свое состояние с (k, t) на $(k + I_i, t + \Delta t)$ из-за прихода заявки из внешней среды в эту СМО; $r_{ij}(k + I_i - I_j, t)$, $h_{ij}(k + I_i - I_j, t)$ – доходы системы S_i , когда сеть меняет свое состояние с (k, t) на $(k + I_i - I_j, t + \Delta t)$ соответственно из-за перехода заявки после обслуживания в системе S_j в эту систему и из-за перехода заявки, не дождавшейся обслуживания в системе S_j , из очереди этой системы в систему S_i .

В докладе рассматривается применение прямого метода и метода преобразований Лапласа для решения данной системы уравнений.

Список цитированных источников

1. Матальцкий, М.А. Вероятностный анализ доходов в банковских сетях / М.А. Матальцкий, А.В. Паньков // Вестник БГУ. Сер.1. Физика, математика, информатика. – 2004. - №2. – С. 86-91.

УДК 519.85

О ДВУХ УСЛОВИЯХ ДЛЯ ВЫЯВЛЕНИЯ НЕСУЩЕСТВЕННЫХ ОГРАНИЧЕНИЙ В ЗАДАЧЕ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

Степанова Т.В.

УО «Гродненский государственный университет имени Янки Купалы», г. Гродно
Научный руководитель – Цехан О.Б., доцент, канд. физ.-мат. наук

Модели линейного программирования часто используются для решения производственных задач. Для реальных объектов размерность модели может быть очень велика, что усложняет решение задачи. В связи с этим актуальны обоснованные методы и удобные процедуры, позволяющие сокращать размерность модели [1, с. 58].

Рассмотрим задачу линейного программирования:

$$\varphi(x) = C_1 X_1 + C_2 X_2 \rightarrow \max \quad (1)$$

$$\begin{cases} a_{11}x_1 + a_{12}x_2 \leq b_1 & (2) \\ a_{21}x_1 + a_{22}x_2 \leq b_2 & (3) \\ x_1 \geq 0, x_2 \geq 0 & (4) \end{cases}$$

$$\begin{cases} a_{21}x_1 + a_{22}x_2 \leq b_2 & (3) \\ x_1 \geq 0, x_2 \geq 0 & (4) \end{cases}$$

$$\begin{cases} x_1 \geq 0, x_2 \geq 0 & (4) \end{cases}$$

Определение. Ограничение (3) является следствием системы неравенств (2) и (4), если для любых переменных, для которых выполнено ограничение (2), выполняется также и ограничение (3).

Неравенство-следствие является несущественным для задачи оптимизации (1)-(4), поскольку все допустимые (а значит, и оптимальные) точки определяются без учета неравенства-следствия.

Очевидно, что для системы (2)-(4) возможны только 3 случая: неравенство является следствием, неравенство не является следствием, система неравенств не имеет решения. В результате анализа всех возможных ситуаций для неравенств (2)-(4) выявлено [2], при каких условиях на параметры неравенств (2),(3) имеет место каждый из случаев. Доказано, в частности, следующее условие, которое в [3, с.69] приведено без доказательства.

Утверждение 1: Пусть параметры системы (2)-(4) удовлетворяют условиям $a_{ij} \geq 0, b_i > 0, x_j \geq 0, i = \overline{1,2}, j = \overline{1,2}$ и связаны соотношениями:

$$\frac{a_{1j}}{b_1} \geq \frac{a_{2j}}{b_2}, \forall j = \overline{1,2}. \quad (5)$$

Тогда ограничение (3) является следствием системы неравенств (2) и (4).

Применение к системе (1)-(4) условия из [3, с.69] дает утверждение 2.

Утверждение 2: Пусть

$$\overline{\varphi_i} \triangleq \max \{c_1 x_1 + c_2 x_2 / a_{i1} x_1 + a_{i2} x_2 \leq b_i, x \geq 0\}, i = \overline{1,2} \quad (6)$$

$$\underline{\varphi_i} \triangleq \min \{c_1 x_1 + c_2 x_2 / a_{i1} x_1 + a_{i2} x_2 \leq b_i, x \geq 0\}, i = \overline{1,2} \quad (7)$$

Если

$$\underline{\varphi}_2 \geq \min_k \bar{\varphi}^k, \tag{8}$$

то неравенство (3) является несущественным для задачи (1)-(4).

Утверждение 3. Условие (8) выполнено всегда, если выполнено условие (5).

Доказательство: Пусть выполнены условия утверждения 1. Рассмотрим условия (6)-(8) и разделим каждое неравенство на $b_i, i = \overline{1,2}$:

$$\bar{\varphi}^i = \max \left\{ c_1 x_1 + c_2 x_2 / \frac{a_{i1} x_1}{b_i} + \frac{a_{i2} x_2}{b_i} \leq 1, x \geq 0 \right\}, i = \overline{1,2}, \tag{9}$$

$$\underline{\varphi}^i = \min \left\{ c_1 x_1 + c_2 x_2 / \frac{a_{i1} x_1}{b_i} + \frac{a_{i2} x_2}{b_i} \geq 1, x \geq 0 \right\}, i = \overline{1,2}, \tag{10}$$

Пусть выполнено (5), но не выполнено (8), т.е. $\underline{\varphi}_2 < \min_k \bar{\varphi}^k$, что равносильно:

$$\min \left\{ c_1 x_1 + c_2 x_2 / \frac{a_{21} x_1}{b_2} + \frac{a_{22} x_2}{b_2} \geq 1, x \geq 0 \right\} < \max \left\{ c_1 x_1 + c_2 x_2 / \frac{a_{11} x_1}{b_1} + \frac{a_{12} x_2}{b_1} \leq 1, x \geq 0 \right\} \tag{11}$$

В условиях утверждения 1 экстремумы в неравенстве (11) достигаются в угловых точках допустимого множества, в которых одна из координат нулевая, а другая – ненулевая. Пусть в обоих случаях это $x_2 = 0$. Тогда $c_1 \geq 0$. Рассмотрим ограничения из неравенства (11) на точках с координатой $x_2 = 0$ Из (11) следует:

$$\min \left\{ c_1 x_1 / \frac{a_{21} x_1}{b_2} = 1, x \geq 0 \right\} < \max \left\{ c_1 x_1 / \frac{a_{11} x_1}{b_1} = 1, x \geq 0 \right\} \tag{12}$$

откуда с учетом $c_1 \geq 0$ имеем : $\frac{a_{21}}{b_2} > \frac{a_{12}}{b_1}$, что противоречит (5), а значит, доказывает результат.

Пример, в котором ограничение-следствие можно выявить по любому из условий утверждений 1 или 2, проиллюстрирован на рисунке 1:

$$\begin{cases} x_1 + x_2 \rightarrow \max \\ 2x_1 + 3x_2 \leq 6 \\ x_1 + 2x_2 \leq 8 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

$$\bar{\varphi}^1 = 3; \bar{\varphi}^2 = 8; \underline{\varphi}^1 = 2; \underline{\varphi}^2 = 4$$

$$\underline{\varphi}_2 = 4 > \min_k \bar{\varphi}^k = 3; \frac{a_{1j}}{b_1} \geq \frac{a_{2j}}{b_2}; \frac{2}{6} \geq \frac{1}{8}; \frac{3}{6} \geq \frac{2}{8}$$

Легко понять, что утверждение, обратное утверждению 3, верно не всегда: выполнение условий утверждения 2 не всегда обеспечивает выполнение условия (5), что демонстрирует следующий пример, проиллюстрированный на рисунке 2:

$$\begin{cases} x_1 + 2x_2 \rightarrow \max \\ -2x_1 + 3x_2 \leq 6 \\ x_1 + 2x_2 \leq 8 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

$$\bar{\varphi}^1 = \infty; \bar{\varphi}^2 = 4; \underline{\varphi}^1 = 0; \underline{\varphi}^2 = 8$$

$$8 = \underline{\varphi}_2 > \min_k \bar{\varphi}^k = 4$$

$$\frac{a_{11}}{b_1} \geq \frac{a_{21}}{b_2}; \text{ не выполнено так как } \frac{-2}{6} \leq \frac{1}{8}$$

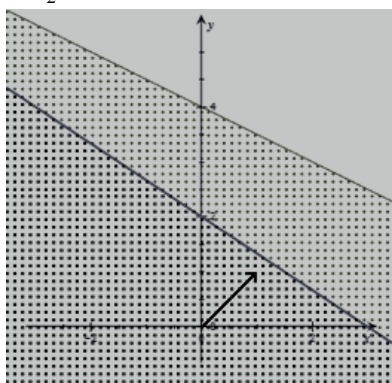


Рисунок 1- Пример выполнения условий 1 и 2

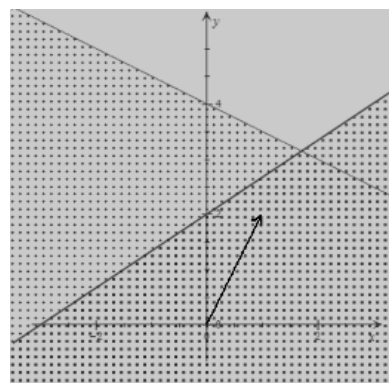


Рисунок 2- Пример выполнения условия 2

Сравнение утверждений 1 и 2 позволяет сделать выводы. Существуют ситуации, когда выполняется утверждение 1, но не выполняется утверждение 2, и наоборот. Рассмотрим пример как на рис.1, но изменим целевую функцию:

$$\begin{cases} -x_1 - x_2 \rightarrow \max \\ 2x_1 + 3x_2 \leq 6 \\ x_1 + 2x_2 \leq 8 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

$$\frac{a_{1j}}{b_1} \geq \frac{a_{2j}}{b_2} \quad \frac{2}{6} \geq \frac{1}{8} \quad \frac{3}{6} \geq \frac{2}{8}; \quad \bar{\varphi}_1 = \infty; \bar{\varphi}_2 = \infty; \underline{\varphi}_1 = 3; \underline{\varphi}_2 = 8; \text{ утверждение 2 не выполнено т.к.:}$$

$$8 = \underline{\varphi}_2 < \min_k \bar{\varphi}_k = \infty$$

Утверждение 1 параметрическое и позволяет несложными вычислениями выявить несущественные ограничения, а утверждение 2 требует оптимизационной задачи, что делает выявление несущественные ограничения сложнее.

Список цитированных источников

1. Цехан, О.Б. Условия для выявления избыточных ограничений в моделях оптимизации // Вестник ГрГУ, сер. 2. – 2011. – № 1. – Стр.58-66.
2. Воробьев, И.Ю. Выделение и анализ ситуаций наличия излишних ограничений в задачах линейного программирования / Воробьев И.Ю., Степанова Т.В. // Молодежь и наука: реальность и будущее: материалы IV Международной научно-практической конференции / Редкол.: О.А. Мазур, Т.Н. Рябченко, А.А. Шатохин: в 4 томах. – Невинномысск: НИЭУП, 2011. – Том IV: Естественные и прикладные науки. – С.423-426
3. Первозванский, А.А. Декомпозиция, агрегирование и приближенная оптимизация / А.А. Первозванский, В.Г. Гайцгори. – НАУКА, 1979. – 344.

УДК 681.84/89:534.647

РЕЖИМЫ ЦИФРОВОГО СПЕКТРАЛЬНОГО АНАЛИЗА ДЛЯ ВИБРОМОНИТОРИНГА ТРУБЧАТЫХ ЭЛЕМЕНТОВ БОЛЬШЕПРОЛЕТНЫХ СООРУЖЕНИЙ

Стойко В.С., Дереченник С.С. - мл.

УО «Брестский государственный технический университет», г. Брест
Научный руководитель – Дереченник С.С., к.т.н., доцент

Решение проблемы конструктивной безопасности сооружений, в особенности большепролетных, обуславливает необходимость системной организации контроля их текущего тех-

нического состояния. В ее основе продолжает оставаться периодическое проведение обследований напряженно-деформированного состояния элементов несущих конструкций. Эффективным неразрушающим методом контроля состояния стержневых (трубчатых) элементов является вибродиагностика – измерение собственных частот поперечных колебаний элемента, зависящей от внутреннего осевого сжимающего либо растягивающего усилия.

Подобный подход, применительно к структурной системе «БрГТУ», положен в основу комплексных мероприятий по мониторингу технического состояния ряда уникальных большепролетных сооружений, в том числе Летнего амфитеатра в г. Витебске. В частности, для определения первой собственной частоты колебаний используется компьютерная измерительная система анализа характеристик вибрации ВИКМА-1, базирующаяся на переносной персональной ЭВМ [1].

Организация регулярных обследований с использованием данной измерительной системы заметно усложнена повышенной трудоемкостью их проведения, что связано с необходимостью присутствия специалистов в точках контроля (т.е. непосредственно на самой конструкции). Возможным вариантом решения указанной проблемы представляется создание стационарно расположенной на сооружении и работающей в автоматическом режиме многоканальной (многоточечной) системы мониторинга. Такая система должна включать в себя: первичные преобразователи (источники вибрации и датчики) и модули обработки вибросигнала (цифрового спектрального анализа), а также центральный блок, обеспечивающий управление работой модулей, сбор, хранение и индикацию (либо передачу по каналам связи) полученных результатов.

Для обработки информации от первичного датчика необходимо: преобразовать электрический сигнал датчика вибрации в цифровой код, накопить отсчеты сигнала за заданный промежуток времени в буферной памяти, выполнить цифровую обработку сигнала (например, быстрое преобразование Фурье) для выявления собственной частоты колебаний. Мониторинг большинства сооружений рассматриваемого типа выполняется не менее чем в 10-15 точках контроля, соответственно число модулей обработки вибросигнала в системе может достигать нескольких десятков. Экономические соображения диктуют построение модуля на недорогих элементах, в том числе микроконтроллере с ограниченной вычислительной мощностью. Поэтому используемые режимы цифровой обработки сигнала должны обеспечить наименьшую вычислительную сложность алгоритма обработки и, в то же время, достаточную для целей мониторинга точность измерений.

В рамках указанной задачи нами выполнен анализ основных характеристик вибрационного сигнала типичного стержневого элемента. Поперечная вибрация в стержне возбуждалась коротким импульсным (ударным) воздействием, а вибрационный сигнал с пьезоэлектрического акселерометра типа AP98-100-1 с помощью компьютерной измерительной системы ВИКМА-1 оцифровывался на частоте дискретизации около 47 кГц и фиксировался в виде текстового файла (временная диаграмма виброускорения для одного из таких сигналов приведена на рисунке 1).

Исходный сигнал обладает двумя важными особенностями: очень высокой (до 40-60 g) амплитудой начальных ускорений возбуждаемого колебания; быстрым затуханием вынужденных колебаний (их амплитуда снижается более чем в 100 раз в течение 0,5 с). Для корректности анализа выделим временной интервал, соответствующий преимущественно свободным колебаниям стержня. В дальнейшем исследовались отрезки сигналов длительностью $t_1 = 5$ с и $t_2 = 3,33$ с, примыкающие к концу общего 10-секундного

интервала. Цифровой спектральный анализ сигнала осуществлялся методом периодограмм [2].

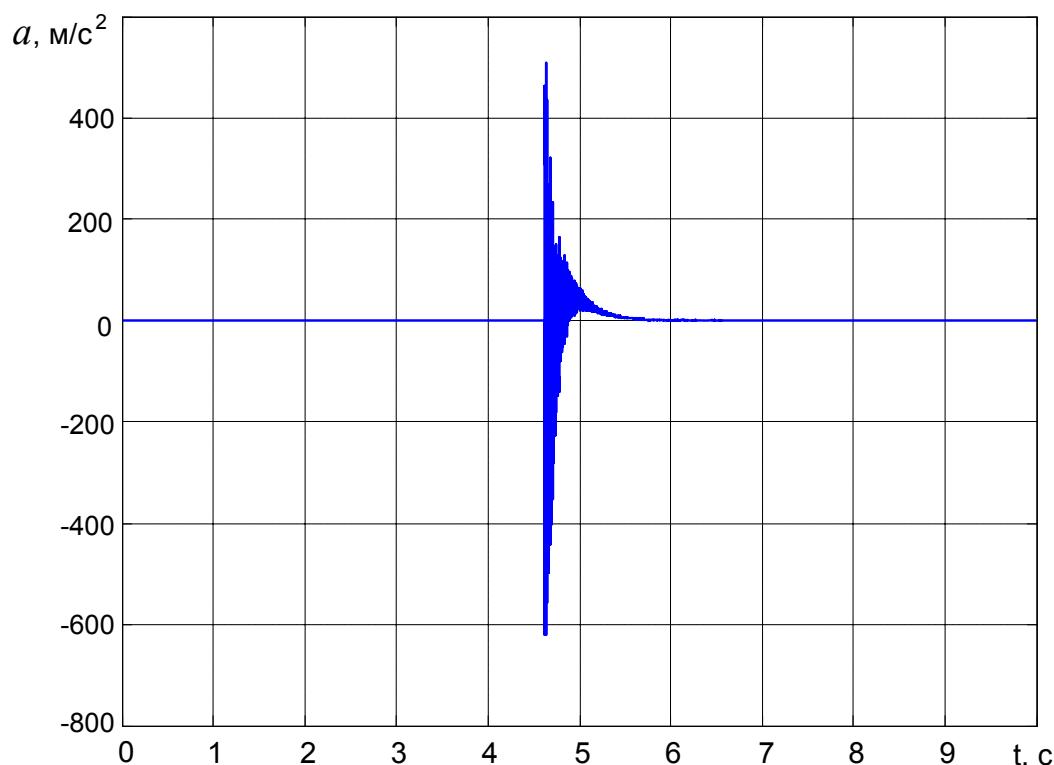


Рисунок 1 – Временная диаграмма виброускорения затухающих поперечных колебаний стержневого элемента, возбужденных импульсным воздействием

Расчеты выполнялись в среде Matlab 6.5 с использованием пакета Signal Processing Toolbox, в том числе его функции `fft`, реализующей алгоритм одномерного дискретного прямого преобразования Фурье (с прямоугольным окном, т.е. без взвешивания отсчетов):

$$\dot{A}(k) = \sum_{j=1}^N a(j) \cdot \exp(-i2\pi(j-1)(k-1)/N),$$

где $\dot{A}(k)$ – комплексные отсчеты дискретного спектра сигнала ($k = \overline{1, N}$), $a(j)$ – отсчеты временной диаграммы, N – количество отсчетов сигнала и полного (симметричного относительно нулевой частоты) спектра импульса.

Поскольку первая собственная частота типичных стержневых элементов, подвергнутым растягивающим / сжимающим усилиям до ± 350 кН, лежит в пределах 35...70 Гц, частотный спектр сигналов анализировался в таком же диапазоне. Учитывая, что исходная частота дискретизации $f_d = 47$ кГц, применительно к низкочастотному спектральному анализу, является избыточной, выполнялась целочисленная децимация (прореживание) исходных отсчетов сигнала с коэффициентом 100, что эквивалентно снижению частоты дискретизации на два порядка. Результаты цифровой обработки сигналов в указанном режиме приведены на рисунке 2.

Для нахождения первой собственной частоты колебаний стержневых элементов (в приведенном случае: 55,3-55,5 Гц) достаточными оказались следующие режимы цифровой обработки сигнала: частота дискретизации $f_d = 0,4 \dots 0,5$ кГц, длительность ис-

следуемых отрезков $t = 5 \dots 3,33$ с. При этом разрешающая способность анализа составляет $\Delta F = 1/t = (0,2 \dots 0,3)$ Гц.

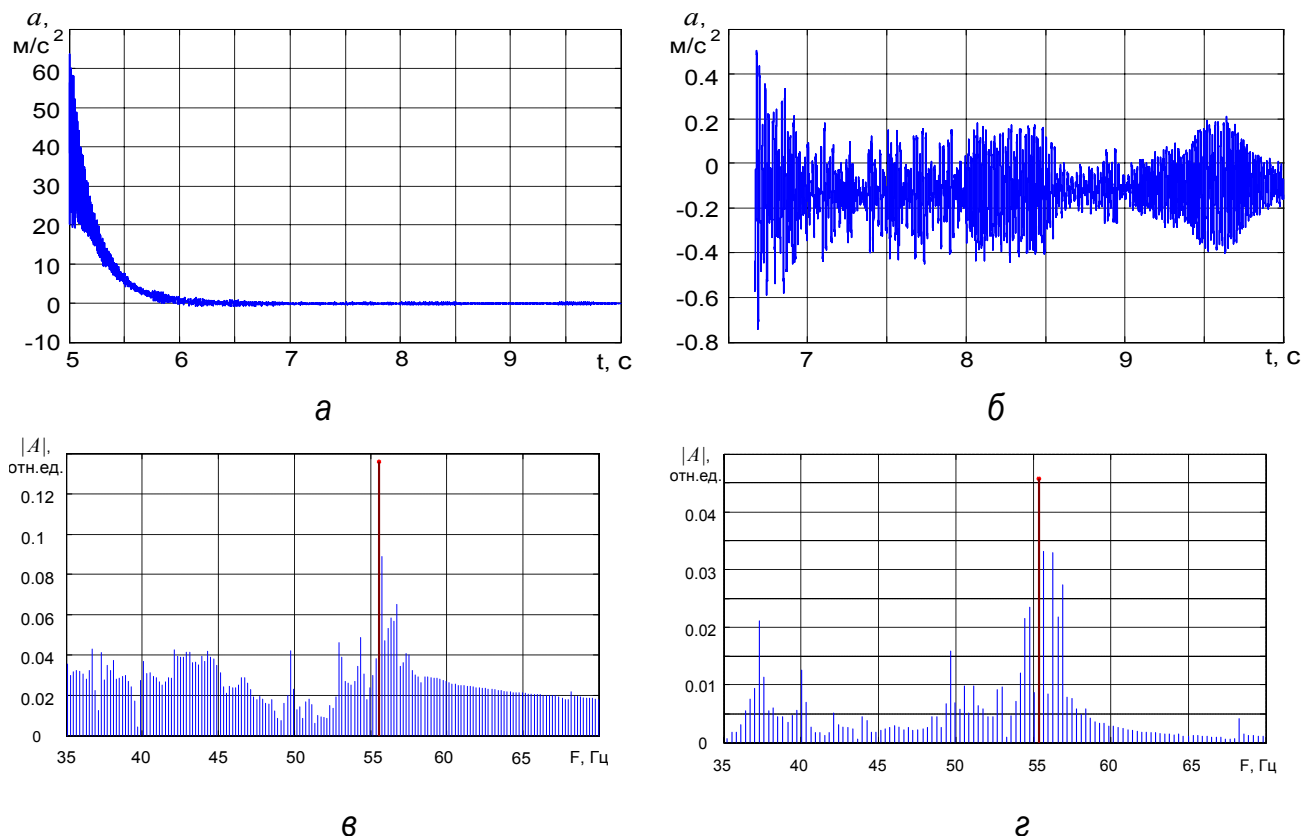
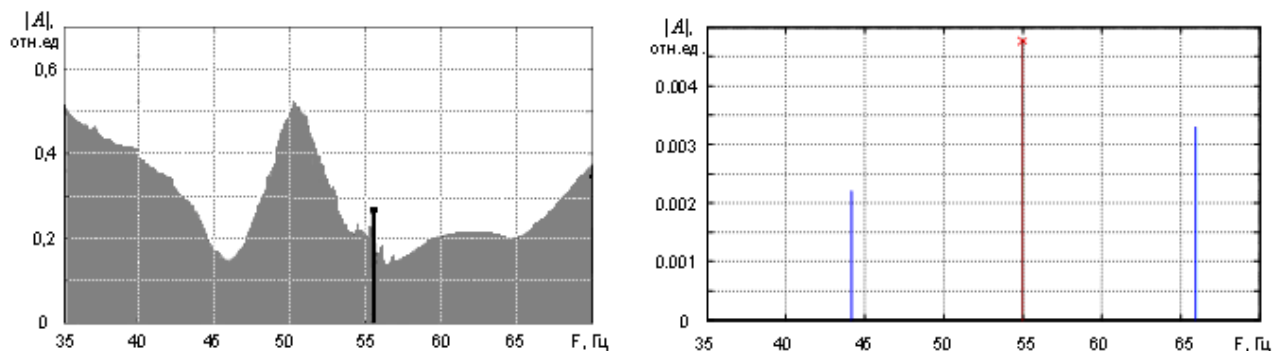


Рисунок 2 – Отрезки временной диаграммы виброколебаний длительностью 5 с (а) и 3,33 с (б) и соответствующие им спектрограммы (в, г), полученные при частоте дискретизации 470 Гц и изображенные в относительных единицах спектральной мощности

Повышение разрешения путем увеличения длительности виброграммы за счет включения в анализируемый интервал участка вынужденных колебаний неэффективно, т.к. наложение широкополосного спектра возбуждающего (ударного) импульса практически исключает корректную расшифровку спектра (рисунок 3а). Интересно, что обработка сверхкоротких интервалов может иногда (впрочем, вполне случайно) давать удовлетворительные результаты, если искомая частота близка к целочисленной кратности интервалу ΔF между линиями спектра. Например, анализируя рассматриваемый выше сигнал на интервале длиной $t = 0,091$ с, удается получить максимальный отсчет спектра на частоте 55 Гц (рисунок 3б). Конечно, точно такой же отсчет был бы наибольшим также при обработке сигнала с другой собственной частотой (например, 50 или 59 Гц), но ровно той же длительности.



а

б

**Рисунок 3 – Спектрограммы виброколебаний для временных интервалов 10 с (а) и 0,091 с (б).
Первая собственная частота колебаний составляет $55,4 \pm 0,2$ Гц**

Таким образом, вибромониторинг стержневых конструктивных элементов с собственными частотами колебаний до 70 Гц путем цифрового спектрального анализа выполняется с разрешающей способностью на уровне 0,2 Гц. Для обработки сигналов применяются автономные встраиваемые средства на базе распространенных микроконтроллеров, которые обеспечивают необходимую разрядность аналогово-цифрового преобразования на частоте до 500 Гц.

Список цитированных источников

1. Мигель, А.В. Строительный мониторинг большепролетного сооружения летнего амфитеатра в г. Витебске / А.В. Мигель, В.И. Драган // Вестник Брестского государственного технического университета. – 2009. – № 1 (55): Строительство и архитектура. – С. 54–58.
2. Гольденберг, Л.М. Цифровая обработка сигналов / Л.М. Гольденберг, Б.Д. Матюшкин, М.Н. Поляк. – М.: Радио и связь. – 256 с.

УДК 519.852

ПРОГРАММНАЯ ВИЗУАЛИЗАЦИЯ ГРАФИЧЕСКОГО МЕТОДА РЕШЕНИЯ ЗАДАЧ ЛП

Трофимов А.Ю.

*УО «Гродненский государственный университет им. Янки Купалы», г. Гродно
Научный руководитель – Будько О.Н., канд. физ.-мат. наук, доцент*

Как известно, линейное программирование (ЛП) – это наука о методах исследования и нахождения наибольших и наименьших значений линейной функции, на неизвестные которой наложены линейные ограничения. Таким образом, задачи ЛП относятся к простейшим задачам на условный экстремум. Казалось бы, что для исследования линейной функции многих переменных на условный экстремум достаточно применить хорошо разработанные методы математического анализа, однако невозможность их использования можно довольно просто проиллюстрировать.

Так как $f(x)$ – линейная функция, то $\frac{\partial f(x)}{\partial x_j} = c_j$, $j = \overline{1, n}$. Все коэффициенты линейной

функции не могут быть равны нулю, следовательно, внутри области, образованной системой ограничений, экстремальных точек не существуют. Они могут быть только на границе области, а математический аппарат производных там не работает.

Простейшим методом решения задач ЛП в случае двух переменных является хорошо известный графический метод [1].

Несмотря на всю простоту графического метода, при написании приложения-визуализации этого метода можно столкнуться с некоторыми трудностями:

- Интерпретация самого алгоритма
- Выявление области допустимых решений (ОДР)

Постановка задачи. Рассмотрим задачу ЛП от двух переменных с ограничениями-неравенствами вида (1):

$$Z(x) = C_1 x_1 + C_2 x_2 \rightarrow \text{extr}$$

(1)

$$\begin{cases} a_{i1}x_1 + a_{i2}x_2 \leq b_i, & i = \overline{1, k} \\ a_{i1}x_1 + a_{i2}x_2 \geq b_i, & i = \overline{k+1, m} \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

Требуется разработать приложение, позволяющее:

- находить решение задач ЛП вида (1);
- проиллюстрировать алгоритм графического метода решения задач ЛП.

Общий алгоритм.

1. По системе ограничений задачи (1) формируем массивы из координат угловых точек ОДР.

2. Находим решение задачи (1) или устанавливаем, что его не существует.

3. Изображаем ОДР, вектор нормали ЦФ и её линии уровня графически.

При разработке алгоритма визуализации использовались свойства задач ЛП, алгоритм графического метода, некоторые идеи вычислительной геометрии.

Обсуждение алгоритма.

Определение угловых точек ОДР. Это можно реализовать, организовав полный перебор граничных прямых. При этом необходимо проверить, принадлежит ли точка пересечения прямых ОДР, то есть удовлетворяет ли она системе ограничений задачи (1). Для последующей идентификации типа решения необходимо дополнительно запомнить, какие прямые дали угловую точку.

Нахождение решения задачи ЛП. Основная сложность – идентификация типа решения. Как известно, возможно четыре случая:

1. Задача имеет единственное решение, которое находится в угловой точке ОДР.
2. Задача имеет бесконечное множество решений на отрезке между двумя угловыми точками.
3. Задача не имеет решений в силу неограниченности ЦФ (бесконечно возрастает (убывает)).
4. Задача не имеет решения, так как система ограничений несовместна.

Найдём значения ЦФ в каждой угловой точке. В первом случае ЦФ будет иметь в некоторой одной точке максимальное (минимальное) значение.

Если же ЦФ принимает максимальное (минимальное) значение в двух точках, то задача имеет бесконечное множество решений, заключённых между данными точками.

Если среди граничных прямых имеются прямые, содержащие только одну угловую точку, то множество неограниченно и может случиться так, что ЦФ будет неограниченно возрастать (убывать).

Если ни одна точка пересечения прямых ОДР не удовлетворяет системе ограничений, то задача ЛП не имеет решений, ОДР есть пустое множество.

Изображение ОДР. По массиву угловых точек изображаем ОДР и закрашиваем её. Через угловые точки строим линии уровня. Линию уровня, проходящую через угловую точку (точки, если их две), содержащую решение задачи, выделяем цветом. Перпендикулярно линиям уровня строим вектор нормали (из начала координат).

Полученные результаты. Таким образом, в работе представлен разработанный алгоритм визуализации графического метода решения задач ЛП, основные элементы которого реализованы программно.

Применение. Применение в учебном процессе при изучении темы «Линейное программирование» для формирования индивидуальных задач. При небольшой доработке может быть использован в качестве обучающего материала по теме «Графический ме-

тод решения задач ЛП».

Список цитированных источников

1. Акулич, И.Л. Экономико-математические методы и модели. Компьютерные технологии решения: учеб. пособие / И.Л. Акулич, Е.И. Велеско, П. Ройш, В.Ф. Стрельчонок. – Минск: БГЭУ, 2003. – 348с.
2. Банди, Б. Основы линейного программирования; пер. с англ. / Б. Банди. – М.: Радио и связь, 1989. – 176 с. УДК 681.327

ИССЛЕДОВАНИЕ СИСТЕМЫ ПОДАЧИ ТОПЛИВА АВТОМОБИЛЯ МАЗ НА ОСНОВЕ КОМПЬЮТЕРНОЙ МОДЕЛИ

Федорович С.А., Шахнова А.А.

УО «Белорусский национальный технический университет», г. Минск

Научный руководитель – Напрасников В.В., к.т.н., доцент

Многие задачи, с которыми приходится в настоящее время сталкиваться исследователям и инженерам, не поддаются аналитическому решению либо требуют огромных затрат на экспериментальную реализацию. Зачастую единственной возможностью экспрессного анализа инженерной проблемы является компьютерное математическое моделирование. Модуль FLOTRAN CFD (расчетная гидродинамика) программы ANSYS предлагает всесторонние инструментальные средства для анализа двумерных и трехмерных полей потока жидкости или газа.

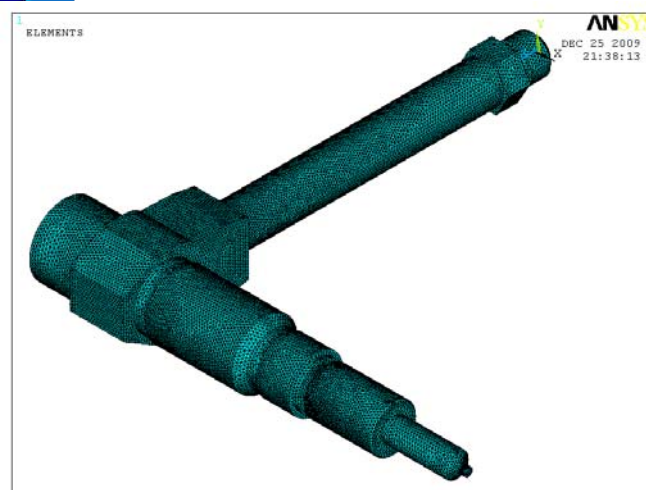
Подход, изложенный в работе, позволяет получить поля скорости и давления для нелинейного канала с турбулентным характером течения в канале системы питания двигателя автомобиля на основе использования инструментария моделирования программы Mechanical APDL (ANSYS) из параметризованного командного файла на языке APDL и расчетных возможностей встроенного модуля гидрогазодинамического анализа.

Разработка трехмерной конечно-элементной модели составляющей части системы

питания двигателя автомобиля МАЗ (форсунки), пригодной для дальнейшего расчета гидрогазодинамическим модулем ANSYS FLOTRAN CFD, осуществлялось с помощью программного модуля Ansys Myltiphysics (Рисунок 1).



**Рисунок 1 – 3D-модель форсунки
автомобиля МАЗ**

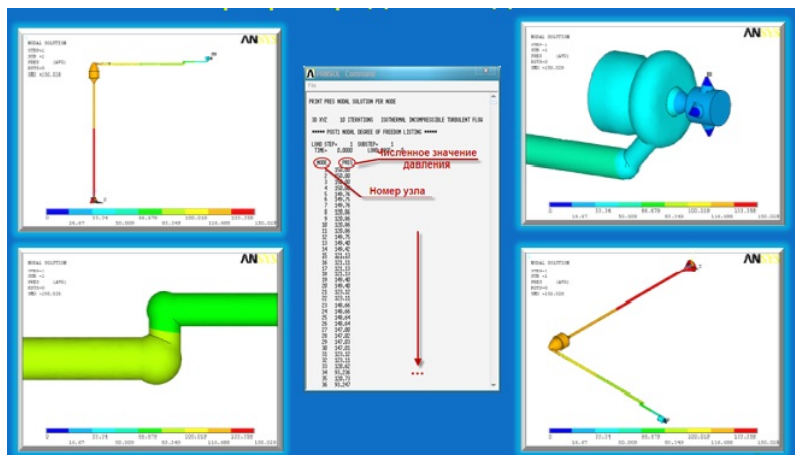


При нанесении КЭ сетки использовались интеллектуальные средства построения сеток модуля Ansys Myltiphysics. Конечно-элементная схема разбиения форсунки пред-

ставлена на рисунке 2.

Рисунок 2 – КЭ-модель форсунки автомобиля МАЗ

Поскольку топливо, рассматриваемое в данной работе, имеет достаточно низкое значение вязкости, равное 0.001, и сравнительно большое давление, принимаем характер нашего течения турбулентным. Выбор между ламинарностью и турбулентностью определяется критерием Рейнольдса, который задает соотношение инерционных сил и



сил внутреннего трения (соотношение инерционного переноса и вязкостного переноса). Результаты расчетов приведены на рисунке 3.

Рисунок 3 – Результаты расчета форсунки автомобиля МАЗ

Полученные данные позволяют делать выводы о конструктивных особенностях системы и оценивать ее прочностные характеристики.

Список цитированных источников

1. Федорович, С.А. Получение полей скорости и давления составляющей системы питания двигателя автомобиля МАЗ / Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях: сб. материалов XIII Республиканской научной конференции студентов и аспирантов ч.1. – Гомель: ГГУ им. Скорины, 2010. – С. 192-193

УДК 519.852

АЛГОРИТМЫ ГЕНЕРАЦИИ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ С НАПЕРЕД ЗАДАНЫМИ СВОЙСТВАМИ

Филиппова М.В.

*УО «Гродненский государственный университет имени Янки Купалы», г. Гродно
Научный руководитель – Будько О.Н., канд. физ.-мат. наук, доцент*

Задача линейного программирования (ЗЛП) подробно изучена в многочисленной литературе, хорошо известны её практические приложения (задача определения оптимального плана производства, раскроя материалов, составления оптимальной смеси и др.). Проблема построения ЗЛП с наперед заданными свойствами является в определенном смысле обратной задачей: по выбранному типу элементов задачи и решения генерируется один из вариантов такой задачи. Для генерации можно использовать датчик случайных чисел.

Постановка задачи. Разработать алгоритмы для генерации ЗЛП с наперед заданными свойствами в случае двух переменных ($n = 2$). Задача генерируется в виде (1):

$$f(x_1, x_2) = c_1 x_1 + c_2 x_2 \rightarrow \text{extr}$$

$$\begin{cases} a_{i1}x_1 + a_{i2}x_2 \leq b_i, & i = \overline{1, m_1} \\ a_{k1}x_1 + a_{k2}x_2 \geq b_k, & k = \overline{m_1 + 1, m} \end{cases} \quad (1)$$

Здесь a_{ij} , b_i , c_j – сформированные постоянные величины.

Общий алгоритм. Общий алгоритм построения ЗЛП с наперед заданными свойствами состоит из следующих этапов:

1. Выбор типа области допустимых решений (ОДР):
 - a) ограниченное множество;
 - b) неограниченное множество;
 - c) пустое множество.
2. Задание количества угловых точек ОДР (m).
3. Построение (генерация) ОДР.
4. Выбор типа решения:
 - a) единственное решение;
 - b) множество решений;
 - c) целевая функция (ЦФ) неограниченно возрастает (убывает);
 - d) множество решений пусто.
5. Построение (генерация) целевой функции.
6. Приведение ЗЛП к общей форме (1).

Обсуждение алгоритма. В основу алгоритмов построения ЗЛП положены свойства ЗЛП, идеи графического метода решения [2, с. 52-53], некоторые идеи из вычислительной геометрии [3, с. 120-121] и параметрического программирования [1, с. 193-194].

Задачи удобно генерировать в виде массивов координат угловых точек ОДР: $X [i]$, $Y [i]$, $i = \overline{1, m}$, а затем приводить их к виду (1).

Для каждого из типов ОДР (см. п. 1 алгоритма) характерны определенные типы решения.

При ограниченной ОДР решение ЗЛП может находиться в угловой точке многоугольника или на отрезке. Значит, существует две возможности выбора типа решения:

- a) единственное решение;
- b) множество решений.

При неограниченной ОДР решение ЗЛП может находиться в угловой точке множества, на отрезке (луче) или решение не существует: ЦФ неограниченно возрастает (убывает). В этом случае существует три возможности выбора типа решения:

- a) единственное решение;
- b) множество решений;
- c) не существует решений.

Выбранный тип решения используется при генерации ОДР и коэффициентов ЦФ.

Построение (генерация) ОДР. Общей частью алгоритма является вычисление и проверка на знак определителя для трех подряд идущих точек: P_{i-1} , P_i , P_{i+1} (2):

$$\Delta_i = \begin{vmatrix} x_{i-1} & y_{i-1} & 1 \\ x_i & y_i & 1 \\ x_{i+1} & y_{i+1} & 1 \end{vmatrix}, \quad (2)$$

где x_i, y_i – координаты точек.

Если $\Delta_i = 0$ – три точки лежат на одной прямой. Третью точку необходимо сгенерировать заново. Если $\Delta_i < 0$ – имеем правосторонний поворот, если $\Delta_i > 0$ – левосторонний поворот. Поворот (направление обхода) можно зафиксировать, а можно выбрать по первой тройке точек. Например, если выбрать обход по часовой стрелке, тогда случай $\Delta_i > 0$ не подходит. Необходимо отбросить последнюю точку и сгенерировать новую, пока не будет выполнено условие $\Delta_i < 0$.

Примеры показывают, что нахождение определителя трех подряд идущих точек не гарантирует выпуклость строящейся ОДР. Необходимо вычислять все определители с текущей точкой k ($\Delta_{i,i+1,k}, i = \overline{1, k-2}$).

По сгенерированным точкам находим уравнения прямых. Из канонического уравнения прямой получаем следующие формулы (3):

$$a_{i,1} = \frac{1}{x_{i+1} - x_i}, \quad a_{i,2} = -\frac{1}{y_{i+1} - y_i}, \quad b_i = \frac{x_i}{x_{i+1} - x_i} - \frac{y_i}{y_{i+1} - y_i}, \quad i = \overline{1, m}, \quad (3)$$

где a_{ij}, b_i – заданные постоянные величины;

где x_i, y_i – координаты точек.

Найденные уравнения прямых образуют систему ограничений. Необходимо определить, какой знак имеет каждое неравенство: больше или равно (\geq) или меньше или равно (\leq). Для этого необходимо сделать предположение в отношении определенного знака, а затем проверить правильность нашего предположения. Поэтому необходимо выбрать любую точку на координатной плоскости, не принадлежащую прямой, а принадлежащую полуплоскости (например, предыдущую по отношению к прямой точку (x_{i-1}, y_{i-1})), и подставить ее координаты в неравенство. Если неравенство будет выполняться, то данная точка является допустимым решением и знак неравенства выбран правильно. Иначе неравенство будет иметь противоположный знак. И так для каждого неравенства.

Построение (генерация) ЦФ. В ЗЛП с заранее заданными свойствами мы сами выбираем, в какой точке или на каком отрезке будет находиться решение задачи.

Если задача имеет единственное решение, необходимо задать уравнение прямой, проходящей через эту точку. Разрешающая прямая (линия уровня) может находиться в интервале между прямыми, пересечением которых является данная точка, но не совпадать с ними, т. е. вектор-градиент может менять направление в некотором интервале $[b; v]$ между этими прямыми.

Если задача имеет множество решений, то линия уровня ЦФ будет параллельна прямой, на которой будут находиться решения. Поэтому ЦФ может иметь такие же коэффициенты при неизвестных, как и уравнение прямой, на которой находится решение, или коэффициенты, умноженные на коэффициент пропорциональности k . Затем выбирается направление изменения ЦФ.

Полученные результаты и применение. В работе кратко представлены разработанные алгоритмы генерации задач линейного программирования с наперед заданными свойствами. Результаты работы будут использованы в учебном процессе при разработке учебно-методических материалов по разделу «Линейное программирование» (инди-

видуальных и тестовых заданий, контрольных работ и др.), для разработки соответствующего программного обеспечения.

Список цитированных источников

1. Акулич, И.Л. Математическое программирование в примерах и задачах/ И.Л. Акулич – М.: Высш.шк., 1986. – 319 с.
2. Костевич, Л.С. Математическое программирование. Информационные технологии оптимальных решений: учебное пособие / Л.С. Костевич. – Минск: Новое знание, 2003. – 424 с.
3. Препарата, Ф. Вычислительная геометрия: Введение; пер. с англ. / Ф. Препарата, Ф. Шеймос – М.: Мир, 1989. – 478 с.

УДК 681.327

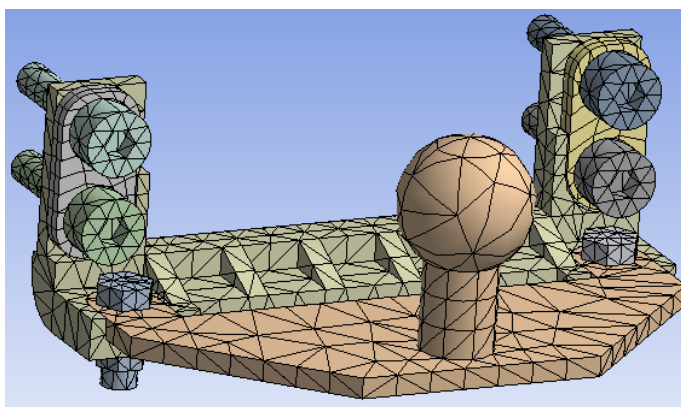
ВЫБОР РАЦИОНАЛЬНЫХ ПАРАМЕТРОВ ПРИЦЕПНОГО УСТРОЙСТВА С ИСПОЛЬЗОВАНИЕМ КОНЕЧНО-ЭЛЕМЕНТНОЙ МОДЕЛИ

Шахнова А.А., Федорович С.А.

*УО «Белорусский национальный технический университет», г. Минск
Научный руководитель – Напрасников В.В., к.т.н., доцент*

Для разработки в сжатые сроки конкурентоспособных машин и механизмов требуется современный подход, а именно: применение компьютерной техники и специализированных программных продуктов для проведения целого комплекса расчетов, важнейшими из которых являются прочностные. Для этих целей предназначены современные конечно-элементные вычислительные комплексы. Такой подход дает предприятию важное преимущество – уже на этапе проработки технического задания анализ конечно-элементной модели позволяет определить, какие технические изменения необходимо внести в конструкцию для полного удовлетворения требований заказчика.

В данной работе на основе использования инструментария моделирования программного модуля Ansys Myltiphysics была разработана параметрическая модель прицепного устройства трактора с использованием языка программирования APDL, затем была создана её конечно-элементная схема разбиения в среде Ansys Workbench (Рисунок 1).



цепного устройства трактора с использованием языка программирования APDL, затем была создана её конечно-элементная схема разбиения в среде Ansys Workbench (Рисунок 1).

Рисунок 1 – Конечно-элементная расчетная схема

При моделирования были получены следующие результаты (Рисунок 2):

- полные смещения,
- запасы прочности,
- эквивалентные напряжения по Мизесу,
- максимальные сдвиговые напряжения,

- контактные давления,
- относительные смещения в зоне контакта элементов конструкции.

Также был проведен расчет с учетом предварительного натяжения в болтовых соединениях.

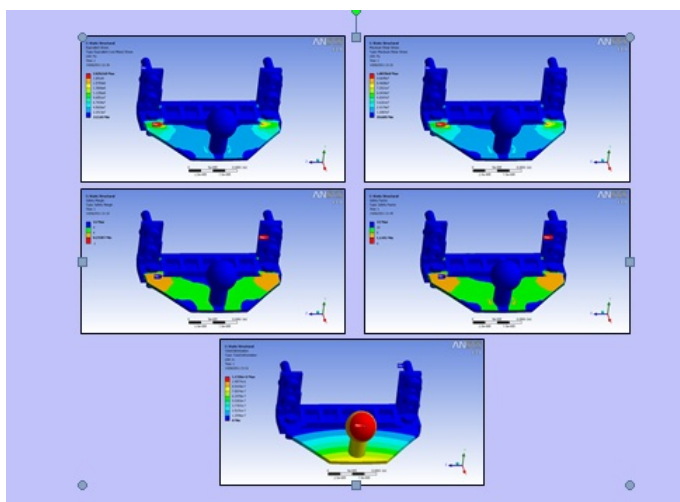
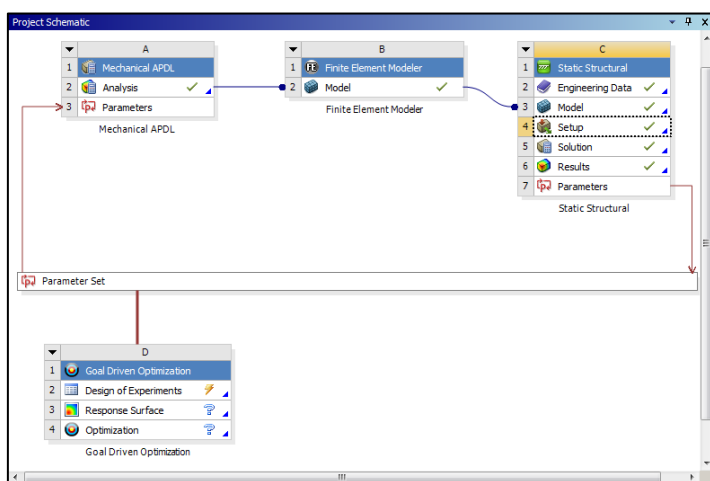


Рисунок 2 – Результаты расчета устройства

Оптимизация конструкции прицепа-



ного устройства выполнялась с использованием технологии ANSYS DesignXplorer, позволяющей осуществлять управление параметрами для прогнозирования влияния параметрических или геометрических изменений на поведение конструкции. Структура проекта при оптимизации представлена на рисунке 3.

Рисунок 3 – Структура проекта для оптимизации

В качестве контролируемых параметров были приняты минимальные значения эквивалентных напряжений по Мизесу и максимальные значения сдвиговые напряжений. Таблица экспериментов изображена на рисунке 4.

Рисунок 4 – Таблица испытаний

J...	P1 - D1	P2 - D11	P3 - D6	P4 - R6	P5 - HK	P6 - D...	P7 - PH1	P8 - Equivalent Stress Maximum (Pa)	P9 - Maximum Shear Stress Maximum (Pa)
2	0.07	0.32	0.112	0.03	0.08	0.009	0.009	1.7257E+08	9.1506E+07
3	0.062	0.293	0.13	0.035	0.06	0.007	0.011	2.8074E+08	1.5007E+08
4	0.0625	0.315	0.122	0.03015	0.072	0.0072	0.0104	3E+08	3E+08
5	0.063	0.301	0.116	0.03405	0.064	0.0079	0.0092	3E+08	3E+08
6	0.069	0.308	0.12	0.0303	0.073	0.0081	0.0109	1.7473E+08	9.2129E+07
7	0.068	0.305	0.124	0.0319	0.063	0.0086	0.0103	1.4916E+08	8.4536E+07
8	0.066	0.314	0.128	0.0328	0.061	0.0071	0.0108	2.4486E+08	1.321E+08
9	0.0665	0.311	0.118	0.0333	0.074	0.0078	0.0091	2.1837E+08	1.2368E+08
10	0.0635	0.318	0.125	0.03215	0.062	0.0085	0.0094	1.7496E+08	9.487E+07
11	0.0675	0.299	0.119	0.0326	0.075	0.0078	0.0107	2.2206E+08	1.1813E+08
12	0.064	0.297	0.117	0.03218	0.063	0.0073	0.0096	2.234E+08	1.2027E+08
13	0.0645	0.307	0.121	0.03115	0.065	0.0087	0.0105	1.4722E+08	7.8125E+07
14	0.065	0.295	0.114	0.03243	0.07	0.008	0.01	2.0262E+08	1.0878E+08

В результате конечно-элементный анализ прицепного устройства навесного оборудования трактора может помочь разработчикам значительно уменьшить расходы на проектирование, изготовление данного вида продукции ещё до начала этапа его производства или эксплуатации.

Список цитированных источников

1. Шахнова, А.А. Параметрическая конечно-элементная модель прицепного устройства /Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях: сб. материалов XIII Республиканской научной конференции студентов и аспирантов – Гомель: ГГУ им. Скорины, 2010. – Ч.1 – С. 198-199.

УДК 004.4

ОСОБЕННОСТИ ПОСТРОЕНИЯ КОМАНДНОГО ФАЙЛА ANSYS ПРИ ПОДГОТОВКЕ ОПТИМИЗАЦИОННЫХ МОДЕЛЕЙ СЛОЖНЫХ ТЕХНИЧЕСКИХ ОБЪЕКТОВ

Якимуш И. С., Мирзаванд М.А.

*УО «Белорусский национальный технический университет», г. Минск
Научный руководитель – Напрасников В.В., к.т.н., доцент*

В процессе проектирования обычно ставится задача определения либо наилучшей структуры объекта, либо наилучших значений параметров объекта. Если оптимизация связана с расчетом оптимальных значений параметров при заданной структуре объекта, то она называется параметрической. Задача выбора оптимальной структуры является структурной оптимизацией. Такая задача называется оптимизационной.

ANSYS обладает встроенными методами и средствами оптимизации.

Первым этапом создания оптимизационной модели является создание командного файла, в котором решается задача однократного анализа на основе скалярных параметров и назначение его в качестве файла для оптимизационных итераций.

Далее следует назначить диапазон изменения и погрешность сходимости для скалярных параметров, которые будут являться переменными оптимизации (optimization variables). К ним относятся:

– Переменные проекта (design variables) – это параметры, которые изменяются с целью нахождения оптимального проекта. Ими обычно являются геометрические параметры, такие как длина, толщина, диаметр или координаты точек. Переменные проекта могут принимать только положительные значения.

– Переменные состояния (state variables) – это параметры, которые служат ограничениями для проекта. Как правило, они представляют собой параметры отклика, являющиеся функциями переменных проекта. Переменные состояния могут быть ограничены максимальным и минимальным значениями или иметь только одно из этих ограничений. Примерами переменных состояния являются напряжения, температуры, скорости тепловых потоков, собственные частоты, деформации и т. п.

– Целевая функция (objective) – это зависимая переменная, которую требуется минимизировать. Она должна быть функцией переменных проекта, т. е. изменение значений переменных проекта должно изменять значение целевой функции. В оптимизационной задаче может быть определена только одна целевая функция.

Существует несколько различных методов, которые можно использовать в ANSYS для выполнения оптимизации.

При использовании метода аппроксимации программа на каждой итерации производит аппроксимацию целевой функции и переменных состояния (методом наименьших квадратов) с использованием данных о предыдущих итерациях.

Пользователь может выбирать тип аппроксимирующей функции ([OPEQN] Curve-fitting control). Можно использовать линейную функцию (Linear), квадратичную функцию без перекрестных членов (Quadratic) и квадратичную функцию с перекрестными членами (Quadratic + X-Term).

Задача считается сошедшейся, если текущий, предыдущий и наилучший проекты (наборы параметров) являются возможными и выполнено одно из следующих условий:

– разность значений целевой функции между лучшим возможным проектом и текущим проектом меньше погрешности сходимости целевой функции (TOLER Convergence tolerance);

– разность значений целевой функции между двумя последними проектами меньше погрешности сходимости целевой функции (TOLER Convergence tolerance);

– разности значений всех переменных проекта между лучшим возможным проектом и текущим проектом меньше их погрешностей сходимости (TOLER);

– разности значений всех переменных проекта между двумя последними проектами меньше их погрешностей сходимости (TOLER).

Иногда процедура оптимизации может быть прервана до достижения сходимости. Это происходит в случае выполнения одного из приведенных ниже условий:

– исчерпано указанное количество итераций (NITR Maximum iterations);

– количество последовательных невозможных проектов достигло указанного предела (NINFS Max infeasible sets).

Сходимость не всегда означает нахождение глобального минимума. Она означает только то, что был выполнен один из указанных выше критериев. Поэтому именно пользователь должен определить, был ли проект достаточно оптимизирован. Если это не так, то можно выполнить дополнительные итерации.

Указанный метод использовался для поиска оптимальной толщины лопасти вентилятора при минимально возможной массе вентилятора компрессорно-конденсаторного агрегата.

Список цитированных источников

1. Сергейкин, О.А. Обзор оптимизационных возможностей программы ANSYS / Сергейкин, О. А.; МГТУ им. Н. Э. Баумана, кафедра МТ-1 (“Металлорежущие станки”).
2. Напрасников, В.В. Параметрическая модель вентилятора компрессорно-конденсаторного агрегата для исследования напряженного состояния при ударных воздействиях на основе командного файла ANSYS / А.В. Скалиух, В.В. Напрасников, И.С. Якимуш // Математическое моделирование и биомеханика в современном университете - Ростов-на-Дону, 2011. – С. 68-69.

СОДЕРЖАНИЕ

ИНТЕЛЛЕКТУАЛЬНЫЕ ТЕХНОЛОГИИ ОБРАБОТКИ ДАННЫХ.
СОВРЕМЕННЫЕ ПРОБЛЕМЫ РОБОТОТЕХНИКИ

Бурич А.Ю. Методы генерирования доверенной цифровой подписи.....	3
Войцехович О.Ю. Система управления транспортом в режиме реального времени	4
Давидюк Ю.И., Савицкий Ю.В. Нейросетевой анализ сигналов ЭЭГ	7
Жукевич А.И. Об одном методе построения компьютерной системы диагностики на основе наблюдаемых данных	9
Касьяник В.В., Дунец А.П. Оценка погрешности одометров мобильного робота с помощью нейронных сетей	11
Кузьмицкий Н.Н. Выбор признаков и модели классификатора в задаче распознавания изображений реальных сцен	14
Кулажевко С.В., Бобров Д.В., Петров П.В., Кольчевский Н.Н. Измерение пространственных координат тестового объекта при помощи web-камер.....	17
Мартинovich О.А. Аддитивная система бизнес-анализа на основе olap-технологий	19
Масалитина Н.Н. Принятие решения по управлению потенциально кризисными процессами на основе нейронной сети с адаптивной структурой.....	21
Мешко Е.Э. Богуш А.С. Создание автономного робота на базе радиуправляемой модели.....	23
Пучик А.А., Хомиченко Д.В. Автономный мобильный робот для участия в кольцевых гонках	25
Смолякова О.Г., Блинов И.Н. Методы отдельного кодирования информации	28
Суворов В.В., Лещёв А.Е. Построение корпоративных бизнес-процессов на ibm websphere process server.....	30
Хомиченко Д.В., Пучик А.А. Алгоритм движения автономного мобильного робота для участия в кольцевых гонках	32

АНАЛИЗ И МОДЕЛИРОВАНИЕ СЛОЖНЫХ СИСТЕМ

Батура П.М., Шешко Е.В. Поиск кратчайших путей на сетях со структурными ограничениями	35
Бекиш Н.В., Китурко О.М. О нахождении ожидаемых доходов в системах нм-сети с приоритетными заявками.....	38
Болтromeюк А.И. О моделировании адаптивной сервисной системы службы телекоммуникации	40
Валеева Я. Е. О моделировании металлургических расплавов.....	43
Вашкевич Д.И. Параметрическая модель для расчета напряженно-деформированного состояния каркаса информационного экрана в среде «ansys»	46
Ивановская Т.К. Анализ и оптимизация моделей прогнозирования доходов предприятий с использованием hm-сетей.....	47

Коваленко И.В. Исследование работоспособности социально-экономических систем	49
Козлова О.А., Левченко Н.В. Ab-initio моделирование электронных свойств наноструктурных материалов с использованием программного комплекса vasp.....	52
Лысюк А.Н., Васильев Д.И. Модель конкуренции кафедр за трудовой ресурс.....	54
Монько В.Д., Маталыцкий М.А. О решении некоторых задач оптимизации смо, применяемых при моделировании систем межбанковских расчетов	56
Пешков И.А. Моделирование процесса разбрасывания удобрений с учетом угла схода с диска средствами "mathematica"	59
Самойленко А.П. Гибкая методология разработки программного обеспечения.....	61
Семашко Д.В., Плышевская О.В. Параметрическая модель для расчета напряженно-деформированного состояния колеса обозрения в среде «ansys»	63
Середенко Е.И., Середенко Р.С. Разработка программного обеспечения для оценки кумулянтных характеристик сигналов.....	65
Статкевич С.Э. О применении нм-сетей с ограниченными временами заявок при прогнозировании доходов.....	68
Степанова Т.В. О двух условиях для выявления несущественных ограничений в задаче линейного программирования	70
Стойко В.С., Дереченник С.С. - мл. Режимы цифрового спектрального анализа для вибромониторинга трубчатых элементов большепролетных сооружений	72
Трофимов А.Ю. Программная визуализация графического метода решения задач лп	76
Федорович С.А., Шахнова А.А. Исследование системы подачи топлива автомобиля МАЗ на основе компьютерной модели	78
Филиппова М.В. Алгоритмы генерации задач линейного программирования с наперед заданными свойствами	79
Шахнова А.А., Федорович С.А. Выбор рациональных параметров прицепного устройства с использованием конечно-элементной модели.....	82
Якимущ И.С., Мирзаванд М.А. Особенности построения командного файла ansys при подготовке оптимизационных моделей сложных технических объектов.....	83

Научное издание

СОВРЕМЕННЫЕ ПРОБЛЕМЫ МАТЕМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

Материалы VII республиканской научной конференции
молодых ученых и студентов
24–26 ноября 2011 года

Часть I

Ответственный за выпуск: **Махнист Л.П.**

Редактор: **Строкач Т.В.**

Компьютерная вёрстка: **Боровикова Е.А.**

Корректор: **Никитчик Е.В.**

ISBN 978-985-493-205-7



9 789854 932057

Лицензия № 02330/0549435 от 8.04.2009 г.

Подписано к печати 11.11.2011 г.

Формат 60x84 1/16. Бумага «Снегурочка».

Гарнитура Arial Narrow. Усл. печ. л. 5,12. Уч.-изд. л. 5,5.

Заказ N 1036. Тираж 60 экз.

Отпечатано на ризографе учреждения образования
«Брестский государственный технический университет».
224017, г. Брест, ул. Московская, 267.