

Министерство науки и высшего образования Российской Федерации
Сибирский федеральный университет

ПРОСПЕКТ СВОБОДНЫЙ – 2020

Материалы Международной конференции молодых ученых

Красноярск, 20 апреля – 18 мая 2020 г.

Красноярск
СФУ
2020

УДК 004
ББК 74.263.2
П827

П827 **Проспект свободный – 2020** : материалы Междунар. конф. молодых ученых. Красноярск, 20 апреля – 18 мая 2020 г. / отв. за выпуск М. В. Носков. – Красноярск : Сиб. федер. ун-т, 2020. – 92 с.
ISBN 978-5-7638-4393-4

Представлены результаты научной работы студентов и молодых учёных Института космических и информационных технологий Сибирского федерального университета в сфере вычислительной техники, прикладной лингвистики и системного анализа.
Предназначены для студентов различных направлений и специальностей, аспирантов, научных работников и преподавателей.

**Ответственность за аутентичность и точность цитат,
имён и иных сведений, а также за соблюдение законов об
интеллектуальной собственности несут авторы
публикуемых материалов.**

Электронный вариант издания см.:
<http://catalog.sfu-kras.ru>

УДК 004
ББК 74.263.2

ISBN 978-5-7638-4393-4

© Сибирский федеральный
университет, 2020

СОДЕРЖАНИЕ

<i>Антоненко О. В.</i>	
Проектирование ансамблей деревьев решений с использованием селекции по информативности и гипермутации	5
<i>Васильев Д. Д.</i>	
Исследование эффективности гибридного эволюционного со случайным вымиранием по времени жизни	8
<i>Власенко М. С., Фокин Д. К., Якимов В. С., Андрианов В. С.</i>	
Помощник для настольных ролевых игр	11
<i>Высотенко Е. И.</i>	
Оценка релевантности текстовых отзывов сервиса анкетирования.....	13
<i>Галин Д. Н.</i>	
Модуль кластеризации данных для сайта-агрегатора интернет-магазинов nazya.com	17
<i>Ганзий Н. С.</i>	
Развитие android-приложения для запоминания картин в игровой форме	19
<i>Дымко К. С.</i>	
Исследование эмоционального окраса пользовательских комментариев.....	22
<i>Иерусалимов С. К.</i>	
Исследование высотных характеристик ТРДДсм на базе динамической модели	25
<i>Качкынбеков Н. Р., Каримов Р. А., Шапошник С. С.</i>	
Разработка информационной системы для сбора и анализа данных по космической деятельности	31
<i>Кемпф Д. А.</i>	
Алгоритм непараметрического дуального управления многомерными дискретно-непрерывными процессами с запаздыванием.....	34
<i>Матюхин Н. Е., Абрамович Р. К.</i>	
Использование генетического алгоритма для выбора структуры искусственной нейронной сети	37
<i>Михайленко Л. А.</i>	
3D модель системы управления ориентации солнечной батареи в среде SIMINTECH.....	41
<i>Мызникова В. А.</i>	
Построение системы управления движением мобильным роботом в среде SIMINTECH.....	44

<i>Сергиенко С. В.</i>	
Моделирование показателей эффективности образовательной	
деятельности высшего учебного заведения.....	48
<i>Соболев Р. Н.</i>	
Управление мощностью производственного участка.....	52
<i>Стебеньков А. С.</i>	
Математическая модель на основе алгоритма случайного	
леса для предсказания успешности обучения	55
<i>Табаков Е. В., Зинина А. И., Киреев Д. С.</i>	
Разработка программного обеспечения для отладки встроенного	
программного обеспечения систем управления космических аппаратов...	58
<i>Терёшина А. В., Яреценко Д. И.</i>	
Идентификация и управление многомерными системами	
с запаздыванием с особенностями	61
<i>Ткачёв Е. С.</i>	
Разработка проекта робота-манипулятора.....	65
<i>Хабибуллоев М. С., Карнаухова Л. А.</i>	
Сегментация и классификация локальных объёмных образований	
на медицинских изображениях	68
<i>Хоров Д. В., Перехрест В. Д.</i>	
Программный комплекс численного исследования жёстких задач	71
<i>Шабашов А. А.</i>	
Разработка алгоритма оценки угла аэродинамического	
скольжения беспилотного летательного аппарата	73
<i>Шатоба А. А.</i>	
Средства повышения производительности JavaScript-кода	77
<i>Шкуринский Д. О., Соколов А. А.</i>	
Сравнение технологий беспроводной связи,	
используемых в IoT-проектах	80
<i>Юришин В. Г.</i>	
Гибридный метод обучения нейронной сети с помощью	
дифференциальной эволюции и обратного распространения ошибки	83
<i>Ященков К. Г.</i>	
Повышение качества исходных метеорологических данных	
как способ улучшения прогнозирования погоды.....	86

ПРОЕКТИРОВАНИЕ АНСАМБЛЕЙ ДЕРЕВЬЕВ РЕШЕНИЙ С ИСПОЛЬЗОВАНИЕМ СЕЛЕКЦИИ ПО ИНФОРМАТИВНОСТИ И ГИПЕРМУТАЦИИ

О. В. Антоненко*

*Научный руководитель – В. В. Становов,
кандидат технических наук, доцент
Сибирский федеральный университет*

Деревья решений – метод анализа данных, основанный на применении различных функций разделения исходного набора данных. Они используются для решения задач классификации и регрессии и состоят из двух типов вершин – внутренние и листовые. Решающие деревья хорошо описывают процесс принятия решений. Для того чтобы построить дерево, его необходимо обучить. Алгоритм построения деревьев решений основывается на выборе наиболее информативного признака на каждой итерации. Также для повышения точности классификации можно использовать ансамбли деревьев решений. Ансамбли – группа деревьев решений, которая формирует прогноз на основе совместных предсказаний. Мутация – один из этапов генетического программирования, при котором содержащиеся в поколении индивиды мутируют случайным образом с определённой вероятностью [1].

В данной работе представлен способ обучения ансамбля деревьев решений, при котором для увеличения обобщающей способности алгоритма будет использоваться мутация и метод разбиения по случайно выбранному признаку.

В первую очередь для построения ансамбля нужно построить деревья. Для того чтобы построить дерево, используется функционал качества, на основе которого осуществляется разбиение дерева в каждой вершине:

$$Q(R_m, j, s) = H(R_m) - \frac{|R_l|}{|R_m|} H(R_l) - \frac{|R_r|}{|R_m|} H(R_r), \quad (1)$$

где $Q(R_m, j, s)$ – функционал качества; $H(R)$ – критерий информативности; R_m – множество объектов, попавших в вершину, разбивающую на данном шаге; R_l и R_r – объекты, попадающие в левое и правое поддерево соответственно при заданном предикате [2].

Критерий информативности оценивает качество распределения целевой переменной среди множества входных объектов.

* © Антоненко О. В., 2020

Энтропийный критерий:

$$H(R) = - \sum_{i=1}^I p_i \log p_i, \quad (2)$$

где p_i – доля объектов класса i ; $H(R)$ – критерий информативности.

Критерий информативности Джини:

$$H(R) = - \sum_{i=1}^I p_i(1 - p_i), \quad (3)$$

где p_i – доля объектов класса i ; $H(R)$ – критерий информативности.

На каждом шаге построения дерева проверяется критерий останова. Он используется, чтобы принять решение: разбивать вершину дальше или сделать листовой.

Критерии останова:

- ограничение максимальной глубины дерева принятия решений;
- ограничение минимального числа объектов в листе;
- остановка в случае, если все объекты в листе относятся к одному классу;
- ограничение максимального количества листьев в дереве принятия решений.

Для того чтобы увеличить разнообразие в ансамбле деревьев решений и повысить их обобщающую способность, на каждой итерации проводится разбиение по случайно выбранному признаку, а не по признаку с максимальной информативностью. Вероятность быть выбранным прямо пропорциональна значению информативности:

$$P(R_m, i) = \frac{Q(R_m, i, s_i)}{\sum_{j=1}^I Q(R_m, j, s_j)}, \quad (4)$$

где $P(R_m, i)$ – вероятность предиката быть выбранным; $Q(R_m, j, s)$ – функционал качества.

Благодаря этой процедуре все предикаты получают возможность быть выбранными и, следовательно, разнообразие деревьев возрастает.

Также для увеличения обобщающей способности используется мутация. Во время мутации происходит обход каждого дерева в ансамбле и с вероятностью 0,5 текущий узел будет изменён, причём для выбора нового предиката также используется разбиение по случайно выбранному признаку. Изменяемый предикат в разбиении не участвует.

На вход программы подаётся информация о размере ансамбля и о максимальной глубине дерева решений. Размер ансамблей равен 10, максимальная глубина дерева равна количеству признаков.

Для подсчёта точности используется десятичастная кросс-валидация. Кросс-валидация – процедура эмпирического оценивания обобщающей способности алгоритмов [3]. Сравнение точности классификации стандартного дерева решений с ансамблем мутирующих деревьев решений на основе случайного разбиения и с ансамблем мутирующих деревьев решений на основе разбиения по лучшему признаку представлено в табл. 1.

Таблица 1

Сравнение точности

Название набора данных	Стандартное дерево решений	Ансамбль мутирующих деревьев решений на основе случайного разбиения	Ансамбль мутирующих деревьев решений на основе разбиения по лучшему признаку
<i>Car</i>	0,801	0,888	0,881
<i>Iris</i>	0,940	0,960	0,955
<i>Balance</i>	0,616	0,708	0,680

Из табл. 1 следует, что точность стандартного алгоритма уступает его модификациям, а ансамбль на основе случайного разбиения превосходит реализацию ансамбля на основе разбиения по лучшему признаку.

Список литературы

1. Радченко В. Открытый курс машинного обучения. Тема 5. Композиции: бэггинг, случайный лес / В. Радченко. URL: <https://habr.com/ru/company/ods/blog/324402/> (дата обращения: 02.04.2020).
2. Соколов Е. Лекция 3. Решающие деревья / Е. Соколов. М.: ФКН ВШЭ, 2018. 3 с.
3. Скользящий контроль. URL: <http://www.machinelearning.ru/wiki/index.php?title=Кросс-валидация> (дата обращения: 02.04.2020).

ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ
ГИБРИДНОГО ЭВОЛЮЦИОННОГО
СО СЛУЧАЙНЫМ ВЫМИРАНИЕМ ПО ВРЕМЕНИ ЖИЗНИ

Д. Д. Васильев*

*Научный руководитель – В. В. Становов,
кандидат технических наук, доцент
Сибирский федеральный университет*

Для решения многих проблем необходимо использовать оптимизацию: например, определить параметры изделия на производстве, определить необходимое количество ресурсов и персонала для выполнения поставленной задачи, найти кратчайший маршрут, и для множества других задач. Одним из подходов к оптимизации являются эволюционные алгоритмы, суть которых сводится к случайному поиску решения с применением особых правил, моделирующих принципы природной эволюции.

Для исследования был разработан алгоритм, основанный на схеме дифференциальной эволюции, имеющий оператор скрещивания, применяющийся в генетических алгоритмах, и имеющий особый оператор случайного вымирания, через который проходят все индивиды в конце каждого поколения. Индивидам назначается вероятность выживания, зависящая от их пригодности в текущей популяции, при этом вероятность уменьшается с каждым поколением, имитируя старость.

Общая схема реализации генетического алгоритма со случаем вымиранием:

1. Инициализация.
2. Основной цикл.
 - 2.1. Сортировка популяции по пригодности.
 - 2.2. Оценивание индивидов, присвоение и изменение шанса выживания.
 - 2.3. Мутация.
 - 2.4. Скрещивание.
 - 2.5. Сортировка популяции по пригодности.
 - 2.6. Архивирование индивидов.
 - 2.7. Вымирание.
 - 2.8. Проверка условия остановки (если не удовлетворено, переход к шагу 2.1).
3. Конец алгоритма, вывод данных.

Оператор оценивания необходим для присвоения индивиду ранга, в зависимости от которого ему присваивается соответствующая вероятность

* © Васильев Д. Д., 2020

выживания, или, если индивид был оценён на предыдущих поколениях, уменьшается на соответствующий коэффициент. Для деления на ранги текущая популяция делится на три части по правилу Золотого Сечения (отсекается большая часть, равная 62 %, остаток также делится на 62 %, получаются части, равные 62, 24 и 14 %). Первый ранг (высший) соответствует самым пригодным индивидам и наименьшей части популяции, т. е. 14 %, третий ранг (низший) соответствует самой большой части, т. е. 62 % популяции и худшим индивидам. Вероятность выживания для первого ранга = 0,9, для второго = 0,8, для третьего = 0,7; коэффициенты уменьшения выживания для первого ранга = 0,8, для второго = 0,7, для третьего = 0,6.

Схема алгоритма основана на дифференциальной эволюции, оттуда же взята мутация (1), которая была модифицирована. Вместо p % лучших берутся индивиды с первым рангом (14 % самых пригодных индивидов) из предыдущего шага:

$$\vec{X}_{i,G} = \vec{X}_{i,G} + F \times (\vec{X}_{g\ best,G} - \vec{X}_{i,G}) + F \times (\vec{X}_{r1,G} - \vec{X}_{r2,G}), \quad (1)$$

где G – текущее поколение; X_i – текущий индивид; F – масштабирующий фактор, принадлежащий промежутку [0, 1] и обычно равный 0,5; $X_{g\ best}$ – случайный индивид из g % лучших индивидов этого поколения ($g = 14\%$); X_{r1} – случайный индивид; X_{r2} – случайный индивид, с вероятностью 0,2 взятый из архива, причём $X_i \neq X_{g\ best} \neq X_{r1} \neq X_{r2}$.

В оператор скрещивания подаётся каждый индивид с его мутантом – тем же индивидом, прошёдшим через оператор мутации. Сам оператор описан К. Дебом (*Simulated Binary Crossover*) [1].

В архиве сохраняются самые лучшие уникальные индивиды, выбранные перед вымиранием. Размер архива равен максимальному размеру популяции (в данном исследовании – 100). Для эффективной работы этого оператора применена следующая схема: выбирается худший индивид из архива, он сравнивается по пригодности с отсортированной популяцией до тех пор, пока не найдётся индивид хуже архивного, все индивиды до него проверяются на уникальность, все уникальные индивиды добавляются в архив, архив сортируется и урезается до его максимального размера.

Предложенная стратегия отбора является модификацией классической стратегии, где создаётся количество потомков, равное размеру популяции, и чтобы сохранить размер популяции постоянным, необходимо отбросить худших индивидов из потомков и родителей [2]. Модификация заключается в добавлении стохастики, при которой вероятность выживания зависит от пригодности, как было описано выше.

Для сравнения эффективности был выбран генетический алгоритм с турнирной селекцией (размер турнира – 2 индивида) операторами *Polynomial mutation* и *Simulated Binary Crossover* [1]. Реализация на языке *Python* представлена библиотекой *DEAP* [3].

Ниже представлена таблица сравнения разработанного алгоритма с реализованным на языке *Python* в *DEAP* (табл. 1). Сверху обозначено количество переменных, по которым проводилось исследование, слева – названия функций. Сравнения проводились *U*-критерием Манна – Уитни, размер популяции – 100 индивидов, для двух переменных количество вычислений – 5000, для пяти – 20000, для десяти – 50000. Для разработанного алгоритма параметр *F* (мутация) генерировался по нормальному закону с математическим ожиданием = 0,5 и стандартным отклонением = 0,2, параметр η (скрещивание) – с математическим ожиданием = 20 и стандартным отклонением = 5. Для алгоритма *DEAP* параметр η (мутация, скрещивание) равен 20. Вероятность скрещивания и мутации в обоих алгоритмах была равна 1. Шанс мутации в алгоритме *DEAP* = 0,05 для каждой переменной. Обозначения в ячейках: 0 – алгоритмы равны; 1 – разработанный алгоритм лучше; –1 – алгоритм *DEAP* лучше.

Таблица 1
Результаты сравнения работы алгоритмов

	2	5	10
<i>Ackley</i>	0	–1	1
<i>Cigar</i>	–1	–1	1
<i>Griewank</i>	–1	–1	1
<i>Rastrigin</i>	1	–1	1
<i>Rosenbrock</i>	–1	–1	–1
<i>Schaffer</i>	–1	–1	1
<i>Schwefel</i>	1	1	1
<i>Sphere</i>	–1	–1	1

Разработанный алгоритм показал себя лучше на задачах с большим количеством переменных в 7 из 8 задач, а также на тестовой функции Швефеля, но уступает алгоритму *DEAP* почти во всех остальных случаях.

Список литературы

1. Deb K. Simulated Binary Crossover for Continuous Space Search / K. Deb, R. B. Agrawal // Complex Systems. 1994. Vol. 1. № 9. P. 115–148.
2. Beyer H.-G. Evolution strategies. A comprehensive introduction / H.-G. Beyer, H.-P. Schwefel // Natural Computing. 2002. № 1. P. 1–52.
3. Distributed Evolutionary Algorithms in Python. URL: <https://github.com/DEAP/deap> (дата обращения: 04.04.2020).

ПОМОЩНИК ДЛЯ НАСТОЛЬНЫХ РОЛЕВЫХ ИГР

М. С. Власенко, Д. К. Фокин, В. С. Якимов, В. С. Андрианов*

*Научный руководитель – А. В. Хныкин,
кандидат технических наук, доцент
Сибирский федеральный университет*

Настольные ролевые игры (НРИ) имеют большую популярность в мире развлечений, постоянно развиваясь. Уже давно выйдя за пределы нишевого хобби для отдельных индивидов, такие игры, как *Dungeons & Dragons*, *Pathfinder*, *Savage Worlds* и др., стали частью жизни многих людей по всему миру. По последним данным [1], самая популярная НРИ в мире *Dungeons & Dragons* насчитывает около 13,7 миллионов активных игроков по всему миру.

НРИ – это вид ролевой игры, в которой участники устно описывают действия своих персонажей, опираясь на их особенности, и успех этих действий зачастую зависит от игровой системы. *Dungeons & Dragons* была первой коммерчески реализованной НРИ, будучи выпущенной в 1974 г. Наиболее важнейшим звеном между игроками и миром игры является мастер – человек, который ведёт игроков по миру, описывает ситуации, в которые попадают персонажи, а также является рефери для каждой игры. Для того чтобы сделать игру более интересной и добавить в неё элемент случайности, используются дайсы – игральные кости с различным количеством сторон, а в качестве свода правил и рекомендаций для построения собственной игры, персонажей и мира используется игровая система.

Люди создают компьютерные игры, пишут книги и снимают фильмы по вселенным, созданным на основе НРИ, что привлекает новых игроков всего мира в это хобби, и, конечно же, в России не остаются в стороне от этой деятельности. К сожалению, далеко не все компании желают адаптировать свои игры под российскую аудиторию, да и удобные инструменты для игроков и мастеров НРИ зачастую не имеют адекватных аналогов на русском языке. Ввиду этого авторами сделана попытка улучшить ситуацию с данной проблемой.

Наш проект направлен на то, чтобы создать приложение, которое поможет игрокам и мастерам разнообразить свой опыт, а также упростить задачу в создании мира, персонажей и отдельных игровых кампаний. Мы объединяем в одном месте такие инструменты, как генераторы имён и названий, приложение для бросков дайсов, создатель листов персонажей и многое другое. Самое главное – мы делаем всё это доступным для российского пользователя. Большинство существующих программ включают в себя

* ©Власенко М. С., Фокин Д. К., Якимов В. С., Андрианов В. С., 2020

только что-то одно, неудобны для мобильного пользования либо не поддерживают русский язык. Наш же проект решает все вышеперечисленные недостатки аналогов для максимально удобного доступа ко всем нужным инструментам.

Реализовав весь необходимый функционал в одном приложении, мы также включим возможность добавления различных новых функций и систем и даже возможность поддержки авторских систем, чтобы каждый игрок мог найти себе что-нибудь по вкусу. Пока что мы обеспечим те функции, которые, по нашему мнению, наиболее нужны пользователям. Позже, основываясь на отзывах и предложениях от пользователей, в приложение будут постепенно вводиться обновления, включающие доработку прежнего функционала и добавление новых функций.

В качестве платформы для приложения была выбрана операционная система *Android*. А в качестве инструмента для разработки была выбрана среда разработки *Android Studio*, т. к. это официальная *IDE*, содержащая актуальные обновления [2]. В качестве языка программирования клиентской части мы решили использовать *Java*, потому что проектная команда уже имеет опыт работы с данным языком. Для серверной части было решено использовать платформу *Firebase*, поскольку она позволяет быстро и удобно устроить работу базы данных, аутентификации, а также данная платформа позволяет легко проводить аналитику приложения. В качестве базы данных мы используем *Firebase Database*, которая использует *NoSQL* структуру. Также используется *Firebase Authentication*, которая позволяет без лишнего кода, легко и просто организовать аутентификацию пользователя самыми разными способами. А *Firebase Functions* используется для создания серверной части приложения. В перспективе также будет использоваться *Firebase Storage* для хранения и извлечения файлов пользователей.

Наш проект является уникальным в своём роде. Среди мобильных приложений нет ничего, что в точности повторяет функционал нашего приложения. Мы создаём удобный сборник самых нужных инструментов для проведения игры, сравнимый с функционалом полноценной онлайн-площадки для НРИ, но спокойно помещающейся в смартфоне. Наше приложение, в отличие от многих других, не требует обязательного подключения к интернету и предоставляет некоторые функции в режиме оффлайн, а подключение с возможностью входа через *Google Play* обеспечивает более полный доступ ко всему, что может предоставить наше приложение.

Список литературы

1. How Many D&D Players Are There Worldwide? 2017. URL: <https://www.dungeongvault.com/how-many-dnd-players-are-there-worldwide> (дата обращения: 10.04.2020).
2. Meet Android Studio. 2020. URL: <https://developer.android.com/studio/intro> (дата обращения: 10.04.2020).

ОЦЕНКА РЕЛЕВАНТНОСТИ ТЕКСТОВЫХ ОТЗЫВОВ СЕРВИСА АНКЕТИРОВАНИЯ

Е. И. Высотенко*

*Научный руководитель – А. А. Даничев,
кандидат технических наук, доцент
Сибирский федеральный университет*

Данная статья посвящена выбору методов нормализации текста и машинного обучения для автоматизации обработки текстовых отзывов студентов Института космических и информационных технологий Сибирского федерального университета (ИКИТ СФУ). Для автоматизации процесса анализа отзывов студентов в персональной образовательной среде института используются два критерия оценки релевантности [1]:

- соответствие текста отзыва словарю вопроса;
- соответствие эмоциональных оттенков отзыва числовой оценке.

Исходный корпус состоит из 4900 текстовых отзывов студентов. Особенностью является малое количество слов в отзыве (в среднем – 7,8 слов на отзыв). Из табл. 1 видно, что половина отзывов содержит от 1 до 3 слов. Как правило, задачи анализа текста применяются к большим объёмам текста. В случае коротких отзывов данных для корректной работы классификатора текста может оказаться недостаточно. Необходимо произвести тестирование различных подходов и алгоритмов на эффективность для различного количества слов в произвольном тексте.

Таблица 1

Количество слов в текстовом отзыве

Процент отзывов	Количество слов в отзыве					
	1	2	3	4	5–10	11–300
	9 %	28 %	15 %	7 %	22 %	19 %

Как правило, оценка соответствия текста отзыва словарю вопроса выполняется с помощью классификатора [2]. При обработке текстовой части отзыва выделяется определённый набор слов, на основе которого вычисляется степень соответствия перечню вопросов. Непосредственное использование слов неприемлемо, т. к. их различные словоформы (лексемы) будут восприниматься различными единицами. Следовательно, обучать и применять классификатор возможно только к большим текстам.

В случае выделения эмоциональной окраски набор слов сопоставляется с тональным словарём. В словарях указывается базовая форма слова (лемма).

Таким образом, для вычисления соответствия текста отзыва словарю вопроса и числовой оценке необходимо:

- во-первых, выполнить нормализацию текста;
- во-вторых, обучить классификатор определять тематику текста;
- в-третьих, эффективно использовать существующие словари.

Нормализация текста возможна с помощью стемминга или лемматизации.

Стемминг в большинстве случаев не требует ресурсов, но может приводить разные по смыслу слова к одной словоформе. При анализе эффективности стемминга за основу была взята библиотека «*SnowballStemmer*» [3].

Лемматизация, в свою очередь, за основу использует морфологические словари и методы контекстного анализа. Вследствие чего данный алгоритм нормализации более зависим от внешних источников и представляет собой ресурсоёмкий процесс. В данной работе использовалась наивная лемматизация (без контекстного анализа). За основу был взят автоматический лемматизатор с поддержкой русского языка – «*TreeTaggerWrapper*» [4].

Для классификации текста рассматривались:

- наивный байесовский классификатор;
- метод k -средних;
- метод опорных векторов.

Результаты сравнения эффективности работы трёх классификаторов представлены в табл. 2, на рис. 1 (предварительный стемминг текста) и рис. 2 (автоматический лемматизатор).

Таблица 2
Основные показатели классификации текстовых отзывов

Метод классификации	Доля правильных решений		
	минимум	максимум	среднее
<i>Стемминг</i>			
наивный байесовский классификатор	0,34	0,7	0,52
метод k -средних	0,01	0,18	0,09
метод опорных векторов	0,68	0,95	0,77
<i>Лемматизация</i>			
наивный байесовский классификатор	0,34	0,68	0,52
метод k -средних	0,01	0,12	0,05
метод опорных векторов	0,66	0,96	0,77

Метод опорных векторов проявил себя одинаково эффективно как для стемминга, так и для наивной лемматизации.

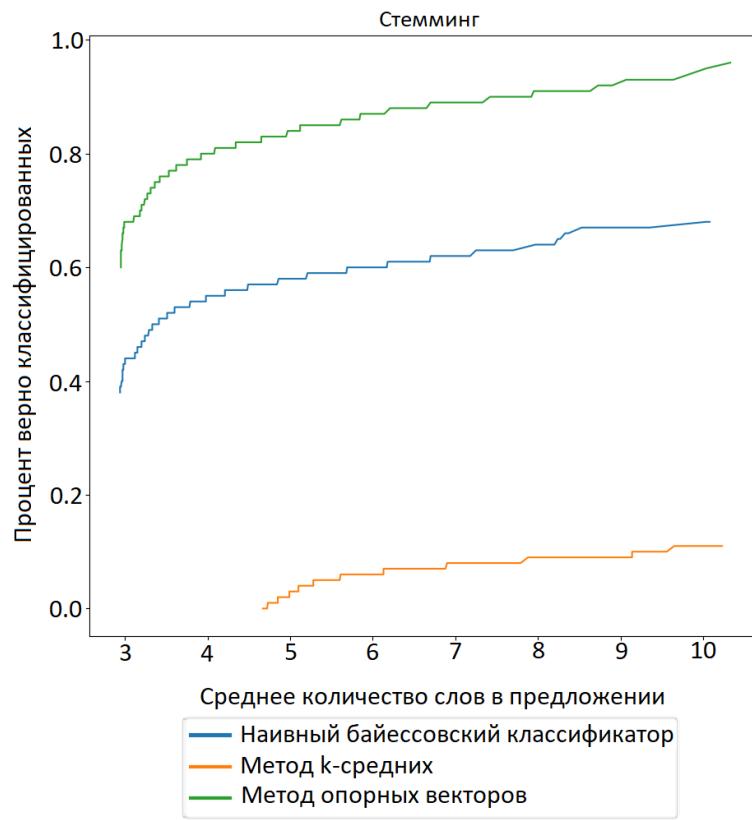


Рис. 1. Результаты классификаторов при предварительном стемминге текста

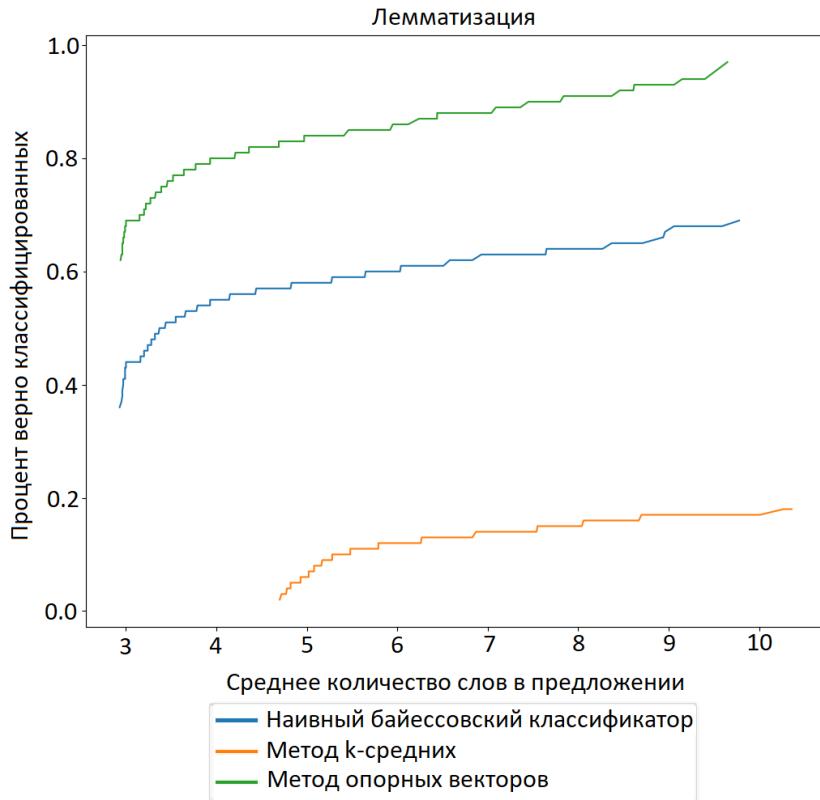


Рис. 2. Результаты классификаторов при предварительной лемматизации текста

Тест применения **тонального словаря** русского языка [5] показал значительное преимущество наивной лемматизации (рис. 3). Нормализация текста посредством стемминга приводит к частым ошибкам.

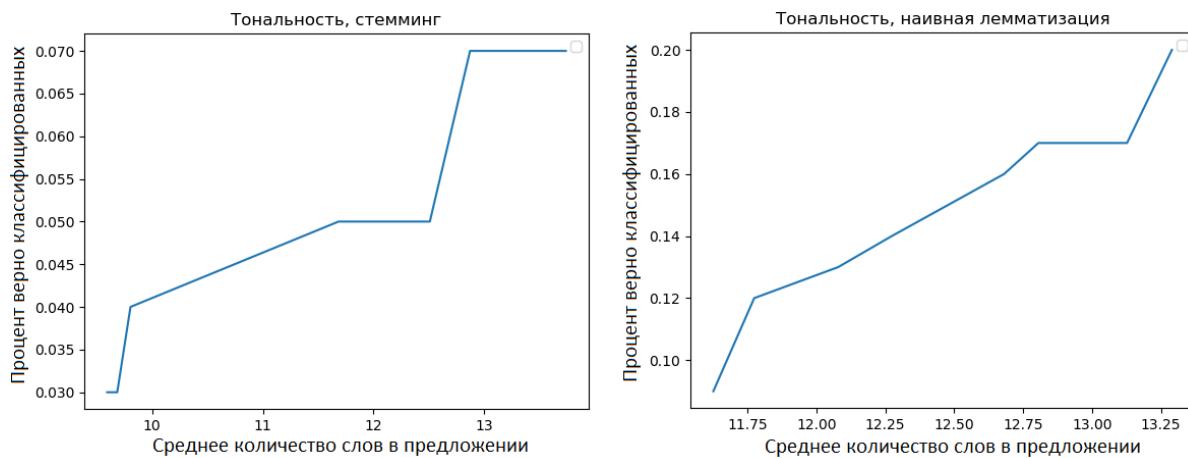


Рис. 3. Результаты определения тональности текстовых отзывов

В результате был произведён анализ методов нормализации и классификации. На основе полученных результатов можно сделать вывод о том, что наиболее подходящим алгоритмом для определения класса текста является метод опорных векторов. Данный классификатор показал наилучший результат в классификации текстовых отзывов с использованием предварительной лемматизации текста размером от 3 до 14 слов. В случае анализа эмоциональной окраски текста наиболее эффективной обработкой текста является лемматизация.

Список литературы

1. Даничев А. А. Аспектный анализ тональности отзывов в образовательной среде / А. А. Даничев, Ю. Ю. Якунин // Информатизация образования и методика электронного обучения. Материалы III Международной научной конференции. В 2-х ч. Красноярск: ИКИТ СФУ, 2019. С. 61–65.
2. «TProger.ru»: обзор методов классификации в машинном обучении. URL: <https://tproger.ru/translations/scikit-learn-in-python/> (дата обращения: 13.04.2020).
3. Документация модуля «Tree Tagger». URL: <https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/> (дата обращения: 12.04.2020).
4. Документация библиотеки «nltk», описание стеммера «SnowballStemmer». URL: <https://kite.com/python/docs/nltk.SnowballStemmer> (дата обращения: 12.04.2020).
5. Онлайн-тезаурус русского языка «Карта слов». URL: <https://github.com/dkulagin/kartaslov> (дата обращения: 12.04.2020).

МОДУЛЬ КЛАСТЕРИЗАЦИИ ДАННЫХ ДЛЯ САЙТА-АГРЕГАТОРА ИНТЕРНЕТ-МАГАЗИНОВ NAZYA.COM

Д. Н. Галин*

*Научный руководитель – А. А. Даничев,
кандидат технических наук, доцент
Сибирский федеральный университет*

Наука о данных, в частности наука о больших данных давно заинтересовала аналитиков из разных сфер. Значительный интерес к работе с внушительным объёмом данных проявляют в сфере торговли, что неудивительно. Обычным явлением в наши дни является сохранение данных покупателей и их покупок для дальнейшей работы с ними. Согласно анализу интернет-рынка, проведённому в ходе выполнения исследования, предложение провести кластеризацию данных является достаточно актуальным и пользуется спросом, однако все эти предложения платные и принцип их работы показан лишь в общих чертах; программный код, используемый для обработки данных, скрыт.

Для создания программы, решающей задачу кластеризации пользователей сайта-агрегатора, необходимо выбрать алгоритм и удобный для его реализации язык разработки. Также необходимо учесть, что работа будет производиться с большим объёмом данных, поэтому стоит озабочиться быстрым действием системы.

В данном конкретном случае ведётся разработка программного обеспечения для сайта, собирающего у себя информацию с множества других сайтов, – своего рода торговой площадки. Основываясь на этом, принято решение произвести *RFM*-анализ и совместить его с иерархической кластеризацией с применением агломеративных методов.

Сведения о клиентах, предоставленные для анализа, даны в общем виде и имеют следующую структуру:

<i>ID</i> клиента	Название товара	Путь к товару на сайте	Дата	Путь категорий	Название категорий	Цена покупки
----------------------	--------------------	---------------------------	------	-------------------	-----------------------	-----------------

RFM-анализ – метод анализа, позволяющий сегментировать клиентов по частоте и сумме покупок и выявлять тех клиентов, которые приносят больше денег.

Аббревиатура *RFM* расшифровывается так:

- *Recency* – давность (как давно клиенты что-то у вас покупали);
- *Frequency* – частота (как часто они у вас покупают);

* © Галин Д. Н., 2020

- *Monetary* – деньги (общая сумма покупок).

В ходе работы решено пренебречь параметром *Recency*, учитывая специфику пользователей сайта *nazya.com* и особенности предоставленных для анализа данных, поскольку этот параметр сильно увеличивает время обработки, усложняет визуализацию данных и не даёт полезной информации в рамках данного периода.

Все пользователи сайта разделены на кластеры в пространстве *Frequency/Monetary*, после чего планируется провести анализ динамики клиентов со временем и вычислить траекторию клиентов для самых выгодных сегментов.

Для кластеризации данных в данной работе используется два подхода.

1. Разбиение на 5 фиксированных кластеров по покупательской способности с дальнейшей обработкой полученных данных при поддержке аналитика компании. В частности – выявление подкатегорий вида «оптовики – мелкие предприниматели».

2. Автоматическое разбиение на кластеры по 5 человек с последующим объединением соседних. Результат, наиболее близкий к нормальному распределению и с которым может работать маркетолог в дальнейшем, получен с использованием переменного шага сегментации: в кластере с людьми, часто совершающими покупки на большие суммы, остались единицы, в то время как наиболее объёмные кластеры (люди, со средней периодичностью покупающие товары на средне-малые суммы) содержат до 200 клиентов.

Кластеризованные данные можно использовать для формирования качественных рекламных предложений группам пользователей с учётом их покупательской способности. Грамотное использование таких данных позволит перевести пользователя в более выгодный для компании кластер. Также планируется кластеризация пользователей по категориям покупаемых ими товаров, что позволит ещё качественнее формировать рекламное предложение, чтобы манипулировать динамикой пользователя по кластерам.

Список литературы

1. Cluster analysys. URL: https://en.wikipedia.org/wiki/Cluster_analysis.
2. RFM. URL: [https://en.wikipedia.org/wiki/RFM_\(market_research\)](https://en.wikipedia.org/wiki/RFM_(market_research)).
3. Clustering – scikit learn 0.22.2 documentation. URL: <https://scikit-learn.org/stable/modules/clustering.html>.
4. Hierarchical clustering (scipy.cluster.hierarchy). URL: <https://docs.scipy.org/doc/scipy/reference/cluster.hierarchy.html>.

РАЗВИТИЕ ANDROID-ПРИЛОЖЕНИЯ ДЛЯ ЗАПОМИНАНИЯ КАРТИН В ИГРОВОЙ ФОРМЕ

Н. С. Ганзий*

*Научный руководитель – В. С. Васильев,
старший преподаватель
Сибирский федеральный университет*

У студентов художественного факультета Сибирского государственного института искусств им. Д. Хворостовского проходит государственная итоговая аттестация, в рамках которой проверяется знание студентом краткой информации об элементах искусства.

Нашей **целью** является доработка существующего проекта «Пазлы – Живопись» [1], позволяющего запоминать картины и краткую информацию о них в игровой форме. Исходный код приложения расположен в репозитории [2]. В проект внесены следующие изменения.

1. **Реализован модуль тестирования**, позволяющий проверить знания студента по любой доступной эпохе, вопросы для теста формируются случайным образом на основе информации об искусстве из базы данных. В рамках тестирования студенту предлагается установить соответствие между картиной и либо эпохой, либо автором.

2. Необходимо, чтобы приложение позволяло работать не только с картинами, но и с изображениями скульптуры. Скульптура, в отличие от картины, зачастую имеет множество авторов (связь «многие-ко-многим»), поэтому в базу данных **добавлена таблица** (*artsAuthors* на рис. 1), связывающая авторов и элементы искусства.

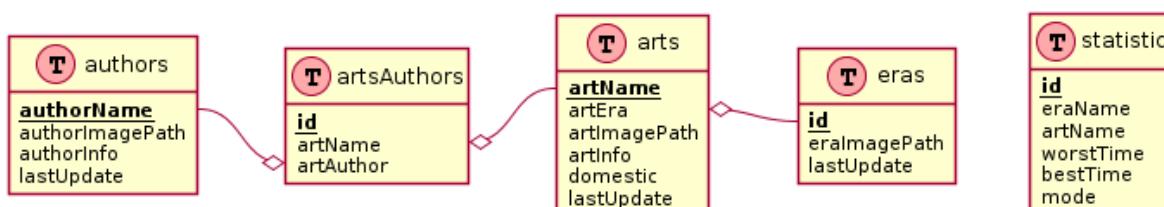


Рис. 1. Структура базы данных

3. **Улучшен модуль обновления** программы, позволяющий загрузить с сервера новые элементы искусства, – интерфейс пользователя позволяет просмотреть отличие локальной и серверной баз данных и выбрать элементы искусства для загрузки.

4. Приложение планируется использовать в двух учебных курсах – «Отечественное искусство» и «Зарубежное искусство», для этого в таблицу

* © Ганзий Н. С., 2020

элементов искусства БД **добавлен признак принадлежности** к тому или иному виду (поле *domestic* в таблице *arts* на рис. 1).

5. Реализован модуль статистики. После успешного сбора пазла собирается статистика, включающая лучшее и худшее время сбора картины для каждого режима игры. Статистика сохраняется в соответствующей таблице локальной БД. Также пользователю выводится прогресс освоения искусства каждой эпохи.

6. Проведены рефакторинг и оптимизация существующего кода:

- с помощью шаблона проектирования *Proxy* [3] и ряда более мелких оптимизаций, которые можно проследить в репозитории [2], решена проблема эффективного отображения большого количества объектов меню выбора уровня;
- улучшена организация исходного кода, в частности вместо копирования зависимостей (исходного кода) теперь используется *git-submodule* [4].

До рефакторинга логика переключения экранов делегировалась специальному классу – *ScreenController*, который хранил ссылки на каждое окно и в процессе обработки сигналов изменял их видимость, реализуя логику стека виджетов [5]. Этот класс имел сильные зависимости от окон, хранил часть их внутреннего состояния – в частности информацию о том, какой выбор пользователь сделал на предыдущих окнах. Чтобы упростить зависимости между окнами, предложено использовать иерархию контроллеров, как показано на рис. 2:

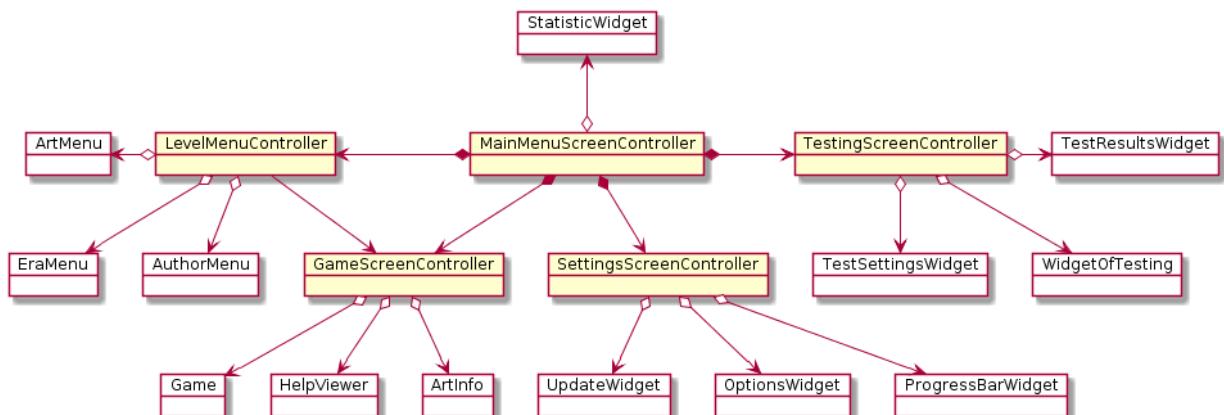


Рис. 2. UML-диаграмма классов – иерархия *ScreenController*-ов

Исходный класс *ScreenController* был разделён на несколько более простых контроллеров, занимающихся только своей частью экранов, например, *GameScreenController*:

- при запуске пазла инициализирует сбор статистики и отображает окно *Game*;
- в процессе игры может выводить окно-помощь (*HelpViewer*);
- после игры выводит информацию о картине (*ArtInfo*) и выполняет возврат к предыдущему контроллеру (из стека).

Подобный подход (иерархия *ScreenController*-ов) может использоваться и в других приложениях со сложной логикой переключения окон. Доработанное приложение «Пазлы – Живопись» планируется применять в учебном процессе.

Разработка проводилась в рамках выполнения выпускной квалификационной работы бакалавра.

Список литературы

1. Android-приложение для запоминания картин в игровой форме // elib.sfu-kras.ru: архив электронных ресурсов СФУ. 2019. URL: <http://elib.sfu-kras.ru/handle/2311/125719> (дата обращения: 25.03.2020).
2. Android-puzzle // GitHub.com: веб-сервис для хостинга IT-проектов и их совместной разработки. URL: <https://github.com/speysinger/Android-puzzle/tree/refactoring> (дата обращения: 26.03.2020).
3. Приёмы объектно-ориентированного программирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес; пер. с англ. СПб.: Питер, 2001. 368 с.
4. Git Tools – Submodules // git-scm: вспомогательная информация по использованию Git. URL: <https://git-scm.com/book/en/v2/Git-Tools-Submodules> (дата обращения: 26.03.2020).
5. Стек виджетов Qt // Блог программиста. 2016. URL: <https://prof.com/forums/topic/screensstack-qt> (дата обращения: 27.03.2020).

ИССЛЕДОВАНИЕ ЭМОЦИОНАЛЬНОГО ОКРАСА ПОЛЬЗОВАТЕЛЬСКИХ КОММЕНТАРИЕВ

К. С. Дымко*

*Научный руководитель – А. В. Хныкин,
кандидат технических наук, доцент
Сибирский федеральный университет*

Современный мир характеризуется огромным потоком коммуникационной информации, который генерируется ежедневно посредством множества инструментов, например, социальными сетями, мессенджерами и мобильными короткими сообщениями. Имея доступ к этой информации, можно проводить различного рода аналитические исследования. Их результаты находят своё прикладное применение, например, в виде персонализированной контекстной рекламы. Однако ещё не все возможности анализа коммуникационной информации достаточно эффективно используются. Большой прикладной интерес представляет оценка эмоционального окраса различных диалогов и сообщений. Автоматизация процесса их анализа позволит определять отношение пользователей к какому-либо товару, рекламе, организации, событию. Поэтому заинтересованной в нём стороной могут быть как сами рекламодатели, в стремлении улучшить качество предоставляемых услуг, так и различные организации, поддерживающие свой положительный социальный имидж, в том числе публичные, непосредственно контактирующие с обществом.

На данный момент существует два чаще всего используемых подхода к проблеме извлечения эмоционально окрашенной лексики [1]: подход, основанный на методах машинного обучения (обучение с частичным участием учителя), и лингвистический подход, основанный на использовании словарей тональной лексики (обучение с учителем). Каждый из этих подходов имеет свои преимущества и недостатки. Методы машинного обучения предполагают, что на основе некоторых данных, полученных из заранее размеченных по тональности текстов, происходит обучение и автоматическое построение классифицирующей функции. Подобные классификаторы демонстрируют высокое качество, но только в рамках одной заданной тематики. Наиболее эффективными и перспективными среди методов глубокого машинного обучения считаются нейронные сети. Они стали широко использоваться при автоматической обработке текстов на естественном языке, в том числе и в области анализа тональности текстов.

Методы лингвистического подхода [2] предполагают поиск эмоциональной лексики в тексте на основе заранее созданных словарей, правил и

* © Дымко К. С., 2020

шаблонов. В данном случае анализу подвергается тональность отдельных слов (термов) в тексте. Термы сравниваются с имеющимся словарём, и каждому выставляется оценка в соответствии с заданной шкалой эмоциональности. Затем, согласно оценкам отдельных слов, которые являются своеобразными маркерами, определяется тональность всего текста в целом. Методы, относящиеся к лингвистическому подходу, не требуют большой коллекции размеченных текстов для обучения и характеризуются высокой точностью результатов анализа в рамках заданной области. Однако они достаточно трудоёмки и ресурсоёмки, поскольку составление словаря или наборов правил требует участия человека.

Метод определения наивероятнейшей общей полярности слов в тексте заключается в оценке частоты встречи этих слов с множеством слов с заранее известной и недвусмысленной полярностью. Например, к таким словам относятся «хороший», «отвратительный» и т. д. Смысл заключается в том, что слова, имеющие позитивную окраску, чаще всего появляются с другими словами с такой же окраской, и наоборот. Это значит, что если нам уже известна эмоциональная окраска слова «великолепный» и это слово соединяется со следующим прилагательным союзами «и» и «или», то система автоматически отнесёт второе слово в базу данных позитивно окрашенных слов. А если после такого слова следует союз «но», система отнесёт слово, находящееся после союза, в противоположную базу данных. Такой подход позволяет существенно упростить и ускорить процесс, уменьшив количество сравнений слов текста со словарём.

Рассмотрев различные методы [3], предлагаемые в рамках этих двух подходов, можно сделать вывод об их достаточно высоком качестве, но при этом крайне низком уровне адаптивности. Каждый метод хорош для определённого узкого круга задач. Перед нами же стоит задача определения эмоциональной окраски комментариев на примере сервиса отзывов о работодателях. Особенностью данного анализа является необходимость определения эмоциональной окраски с учётом межпользовательского общения. Так, просто «плохой» комментарий может быть оценён по-разному. Один вариант, когда имеется только отзыв об организации, и другой вариант, когда этот отзыв имеет отношение к себе от других пользователей. В этом случае эмоциональный оттенок может относиться не к самой компании, а к одному из комментариев о ней. Такая ситуация значительно усложняет анализ и ставит перед нами нетривиальную задачу. Кроме того, задача усложняется также лингвистическими особенностями русского языка, например, словами с двояким смыслом, обилием саркастических оборотов, достаточно разнообразным словарём.

Список литературы

1. Пескишева Т. А. Методы анализа тональности текстов на естественном языке / Т. А. Пескишева // Сборник «Общество. Наука. Инновации» (НПК – 2017). Киров: ВятГУ, 2017. С. 1730–1742.

2. Денисова Д. С. Анализ эмоциональной окраски текста / Д. С. Денисова // Синергия наук. 2018. № 19. С. 1415–1424.

3. Тутубалина Е. В. Тестирование методов анализа тональности текста, основанных на словарях / Е. В. Тутубалина, В. В. Иванов и др. // Электронные библиотеки. 2015. Т. 18. № 3–4. С. 138–162.

ИССЛЕДОВАНИЕ ВЫСОТНЫХ ХАРАКТЕРИСТИК ТРДДсм НА БАЗЕ ДИНАМИЧЕСКОЙ МОДЕЛИ

С. А. Иерусалимов*

Научный руководитель – Г. М. Макарьянц,
доктор технических наук

Самарский национальный исследовательский университет
им. академика С. П. Королёва

Повышение эффективности работы авиационного газотурбинного двигателя (ГТД) связано с внедрением новых алгоритмов управления, реализация которых невозможна без использования новых интеллектуальных агрегатов и электронных цифровых систем автоматического управления (САУ). Работа современных средств электроавтоматики базируется на применении бортовой математической модели объекта управления – газотурбинного двигателя.

Для повышения эффективности управления рабочим процессом силовой установки на протяжении всего полётного цикла необходимо в каждый момент времени знать значения основных параметров двигателя. Однако высокой точности расчёта параметров ГТД удалось достичь только для статических режимов. В настоящей статье для ТРДДсм на примере двигателя АИ–222–25 представлена методика моделирования динамики ГТД в пакете *MATLAB Simulink*.

В работе рассматривается методика математического моделирования узлов ГТД. Многие вопросы, связанные с этим процессом, уже решены, однако на необходимом для практической реализации уровне не рассматривается вопрос газодинамической связи между соплом и окружающей средой. В ходе переходного процесса в двигателе возникает докритическое истечение продуктов сгорания в сопле, требующее учёта зависимости относительной пропускной способности от располагаемой степени понижения давления на турбине (фактически от давления за турбиной). Неучёт этой зависимости приводит к значительному расхождению расчётного значения расхода воздуха через двигатель и его фактической величины. Это вызывает неустойчивость в работе математической модели и «развал» решения. Целью работы является создание динамической математической модели ТРДДсм как объекта регулирования на базе уравнений термогазодинамики узлов двигателя и уточнённой модели динамики сопла для дальнейшего использования при исследовании различных вариантов и способов управления ГТД.

Задача решается на примере основного контура ТРДДсм АИ–222–25. Параметры окружающей среды взяты из стандартных атмосферных условий. При проектировании математической модели использовался программный пакет *MATLAB Simulink*, набор инструментов которого предоставляет возможность создания системы дифференциальных уравнений и её решения методом Рунге – Кута. Разработанная модель определяет газодинамические и механические связи между конструктивными узлами двигателя. Используемые уравнения подразделяются на несколько основных групп:

- уравнения баланса мощности, расходов, давлений;
- уравнения сохранения массы.

Необходимым условием устойчивой работы математической модели является наличие отлаженного модуля сопла, основным свойством которого является однозначная зависимость тяги от режима работы двигателя. Чтобы обеспечить такое условие, необходимо описать функциональную зависимость относительной пропускной способности сопла от параметров на входе сопла. Известно, что относительная пропускная способность сопла является стандартной газодинамической функцией, поэтому она может быть найдена через величину приведённой скорости потока. В свою очередь, приведённая скорость потока является функцией располагаемого перепада давления в сопле, а значит, при постоянном давлении окружающего воздуха однозначно определяется давлением воздуха за турбиной:

$$\lambda_{c.kp} = \sqrt{\frac{k+1}{k-1} \times \left(1 - \left(\frac{1}{\pi_{c.p.}^*} \right)^{\frac{k-1}{k}} \right)}. \quad (1)$$

Далее от приведённой скорости мы переходим к расчёту величины относительной пропускной способности сопла. Данная зависимость описывается следующим уравнением:

$$q(\lambda_{c.kp}) = \lambda_{c.kp} \times \left(1 - \left(\frac{k-1}{k+1} \right) \times \lambda_{c.kp}^2 \right)^{\frac{1}{k-1}} \times \left(\frac{k+1}{2} \right)^{\frac{1}{k-1}}. \quad (2)$$

Чтобы иметь возможность использовать разработанную модель в дальнейших исследованиях, необходимо доказать её адекватность реальным показателям двигателя. Для этого проведена верификация по статическим режимам, где сравниваются результаты разработанной математической модели на нескольких режимах с расчётом в программе АСТРА, адекватность которой подтверждается экспериментальными данными и описывается в работе [6]. Результаты сравнения приведены ниже в табл. 1.

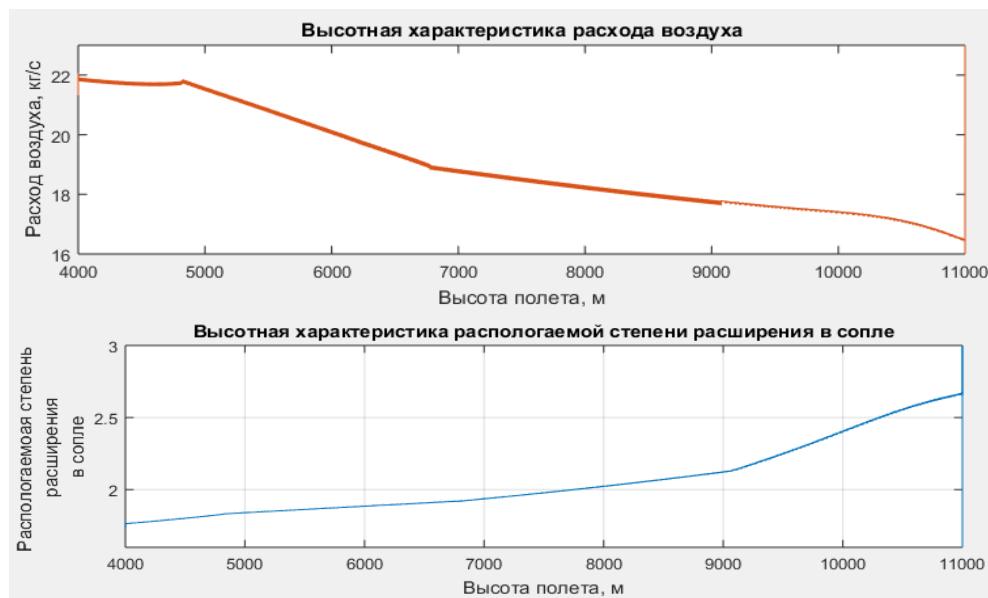
Таблица 1

Сравнение результатов расчёта в ACTRA и MATLAB

Режим ($M = 0,6$)	$H = 0 \text{ м};$ $Gt = 0,458 \text{ кг/с; на земле}$			$H = 8000 \text{ м};$ $Gt = 0,105 \text{ кг/с}$			$H = 11000 \text{ м};$ $Gt = 0,089 \text{ кг/с}$		
	ACTRA	MATLAB	Отклонение, %	ACTRA	MATLAB	Отклонение, %	ACTRA	MATLAB	Отклонение, %
$T_g, \text{К}$	1401,3	1398	0,235	1024	1062	3,711	1074	1064	0,931
$G_{\text{возд}}, \text{кг/с}$	49,8	52,7	5,823	18,43	17,2	6,674	13,8	14,2	2,266
$P_{\text{двиг}}, \text{kН}$	25,2	25,1	0,398	3,88	3,68	5,155	3,69	3,52	4,830
$C_{\text{уд}}, \text{кг}/(\text{kН} \times \text{с})$	65,1	65,8	1,075	97,43	102,7	5,409	86,74	91,12	5,050

При расчётах было принято допущение, что режим двигателя описывается только одним параметром – расходом топлива. Погрешность верификации на основе данных, полученных в результате расчёта статических режимов по главному параметру – тяге двигателя – не превышает на всех трёх режимах 5,16 %, а это значит, что модель можно считать адекватной реальным параметрам двигателя.

Для дальнейших исследований также необходимо удостовериться в качественной физической адекватности модели при расчёте переходных процессов с помощью анализа высотных характеристик двигателя. Их можно найти, определив зависимость изменения основных параметров двигателя. Высотная характеристика представлена в виде функций $G_{\text{возд}}$, $P_{\text{двиг}}$, T_g , $C_{\text{уд}}$, $n_{\text{ВД}}$, p_T , $\pi_{\text{с.р.}}^*$ = $f(H_{\text{полёта}})$, представленных на рис. 1. Высота полёта ЛА задаётся в виде линейно убывающего сигнала для диапазона $H_{\text{п.}} = [0,4 \dots 1,1] \times H_{\text{п.макс}}$, где $H_{\text{п.макс}} = 10 \text{ км}$.



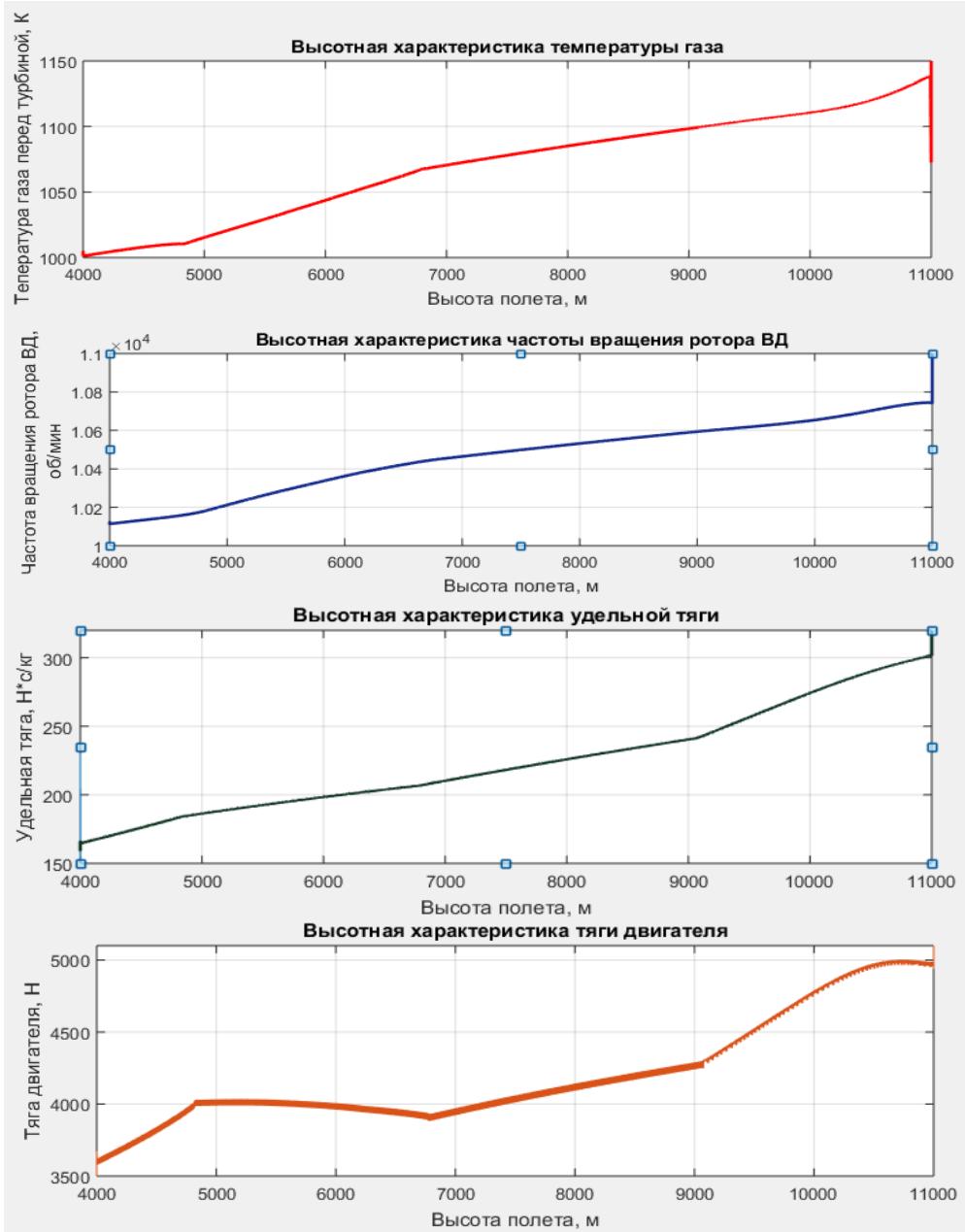


Рис. 1. Высотные характеристики основного контура ТРДДсм

Из графиков, представленных на рис. 1, видно противоречивое влияние различных факторов на тягу двигателя. С увеличением высоты полёта значительно изменяется давление окружающего воздуха, которое оказывает решающее влияние на давления во всех сечениях двигателя. Следствием этого является снижение расхода воздуха через двигатель. Однако то же снижение давления приводит к раскрутке роторов, а значит, и к увеличению степени повышения давления в компрессорах. Данный факт приводит к увеличению температуры газа перед турбиной, а значит, и к повышению КПД цикла, что влечёт за собой увеличение располагаемой степени расширения в сопле. Последняя оказывает значительное влияние на величину удельной тяги, которая увеличивается быстрее, чем уменьшается расход воздуха. Противодействие двух факторов приводит к сложному характеру

изменения тяги. Данные зависимости согласуются с теорией двигателей, описанных в книге [1].

В результате разработки модели были получены следующие результаты. Проведена верификация модели по статическим режимам: на всех трёх режимах погрешность с расчётом в программе АСТРА по тяге двигателя не превысила 4 %. Были получены физически адекватные результаты при тестировании математической модели по высотным характеристикам, которые тем не менее в дальнейшем нуждаются в динамической верификации с экспериментальными данными.

На данный момент решается задача разработки математической модели для моделирования динамических процессов на режиме «малый газ», что подразумевает обработку большого массива реальных характеристик двигателя. Также целесообразно продолжать разработку САУ на основе выполненной математической модели. Работа будет связана с проектированием регулятора, позволяющего контролировать переходные процессы при выходе с режима «малый газ» до взлётного режима и далее в ходе всего полётного цикла по границе режима устойчивой работы и предельной температуры газа перед турбиной.

Список литературы

1. Кулагин В. В. Теория, расчёт и проектирование авиационных двигателей и энергетических установок. Кн. 1. Основы теории ГТД. Рабочий процесс и термогазодинамический анализ. Кн. 2. Основы теории ГТД. Совместная работа узлов выполненного двигателя и его характеристики: учебник для студентов вузов, обучающихся по специальности «Авиационные двигатели и энергетические установки» направления подготовки дипломированных специалистов «Двигатели летательных аппаратов» / В. В. Кулагин. 2-е изд., испр. М.: Машиностроение, 2003. 614 с.
2. Гуревич О. С. Управление авиационными газотурбинными двигателями: учеб. пособие / О. С. Гуревич. М.: Изд-во МАИ, 2001. 100 с.
3. Гольберг Ф. Д. Математические модели газотурбинных двигателей как объектов управления: учеб. пособие / Ф. Д. Гольберг, А. В. Батенин. М.: Изд-во МАИ, 1999. 80 с.
4. Костюков В. М. Модель газотурбинного двигателя и его системы управления для особых значений углов атаки и метеорологических условий / В. М. Костюков, Н. И. Капырин // Труды МАИ. 2011. № 49. 13 с.
5. Казанцев А. В. Метод идентификации динамических моделей авиационных ГТД на основе задач оптимизации с локальными критериями / А. В. Казанцев. Уфа, 2019. 18 с.
6. Кузьмичёв В. С. Формирование виртуальной модели рабочего процесса газотурбинного двигателя в САЕ-системе «АСТРА» / В. С. Кузьмичёв, В. В. Кулагин, И. Н. Крупенич, А. Ю. Ткаченко, В. Н. Рыбаков // Труды МАИ. 2013. № 67. 13 с.

7. Ахметов Ю. М. Моделирование неустановившихся режимов авиационных ГТД с учётом гистерезиса границы устойчивой работы компрессора / Ю. М. Ахметов, Д. А. Ахмедзянов, А. Б. Михайлова, А. Е. Михайлов // Вестник УГАТУ. 2014. Т. 18. № 2 (63). С. 3–9.

РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ СБОРА И АНАЛИЗА ДАННЫХ ПО КОСМИЧЕСКОЙ ДЕЯТЕЛЬНОСТИ

Н. Р. Качкынбеков, Р. А. Каримов, С. С. Шапошник*

*Научный руководитель – А. В. Хныкин,
кандидат технических наук, доцент
Сибирский федеральный университет*

Освоение космоса – одно из приоритетных направлений развития человечества. Ежегодно тратится огромное количество финансовых ресурсов на развитие космических технологий [1].

На сегодняшний день кроме государственных корпораций успешно развиваются и частные космические компании, которые реализуют свои собственные космические программы. Они вносят огромный вклад в развитие космических технологий. В сети Интернет существует огромное количество сайтов, посвящённых освоению космоса.

Вместе с тем обнаружен недостаток актуальной информации по отечественной космической деятельности, осуществляющейся государственной корпорацией Роскосмос. Поэтому было решено восполнить этот пробел в информационном обеспечении, объединив функции некоторых интернет-ресурсов и добавив статистику о результатах космической деятельности корпорации Роскосмос. Для решения такой задачи разрабатывается автоматизированная система, которая состоит из трёх основных модулей:

1) подробная статистика по российским запускам (где и когда совершён, описание полезной нагрузки, космодром и т. п.);

2) отсчёт до ближайших космических запусков, причём не обязательно этот запуск совершается Роскосмосом, отсчёт будет вестись и для зарубежных запусков;

3) графическое представление прошедших запусков.

Для решения поставленной задачи уже разработано веб-приложение на языке *Python*, служащее в качестве *API* (для серверной части), с использованием *JavaScript* библиотеки *ReactJS* (для клиентской части), которая имеет доступ к вышеупомянутой *API*. Серверная часть имеет функцию автоматического сбора данных из открытых источников.

За основой автоматизации сбора данных стоит парсер. Парсер – программа, преобразующая входные данные в структурированный формат [2]. Парсер выполняет синтаксический анализ текста. В нашем случае это не совсем текст, а *html*-код страниц сайтов, на которых содержится нужная информация.

* © Качкынбеков Н. Р., Каримов Р. А., Шапошник С. С., 2020

Часть сервера, которая обеспечивает хранение обработанной информации и доступ к ней, работает с помощью фреймворка *Django* [3], который служит каркасом серверной части приложения и библиотеки *Django Rest Framework* [4], дающей возможность с лёгкостью создать *Rest API*.

Также не менее важным решением является выбор *Docker*. *Docker* – программное обеспечение для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации [5]. Основной причиной использования *Docker* является то, что серверная часть использует большое количество фреймворков, библиотек и других зависимостей, некоторые из которых трудно настраивать для определённых операционных систем.

При запуске приложения пользователя встречает таблица всех предстоящих мировых космических запусков. Далее можно по очереди увидеть данные по российским запускам, это:

- предстоящие запуски;
- статистика по прошедшим запускам;
- статистика по космодромам;
- статистика по разгонным блокам;
- статистика по ракетоносителям;
- статистика по космическим аппаратам;
- статистика по орбитальным группировкам;
- статистика по космической обсерватории.

Каждый блок статистики имеет несколько режимов отображения, такие как график, гистограмма и текстовое отображение, которые позволяют более удобно и наглядно знакомиться с данными. Также в некоторых блоках, таких как космодромы, ракетоносители, космические аппараты и т. д., включены режимы ознакомления с основной информацией о них.

В результате реализации проекта будет разработана автоматизированная информационная система, позволяющая отслеживать прошедшие и предстоящие космические запуски по всему миру и имеющая графическое представление прошедших запусков на карте Земли. Также система позволяет анализировать и отображать статистику по существующим данным российских запусков. Информационная система реализует автоматизированный сбор данных из открытых источников.

Список литературы

1. Россия планирует потратить на космос более 600 млрд рублей в 2020–2022 годах. URL: www.tass.ru/kosmos/6944606 (дата обращения: 13.04.2020).

2. 10 инструментов, позволяющих парсить информацию с веб-сайтов, включая цены конкурентов + правовая оценка для России. URL: www.habr.com/ru/post/340038/ (дата обращения: 13.04.2020).

3. Django documentation. URL: docs.djangoproject.com/en/3.0/ (дата обращения: 13.04.2020).

4. Django Rest Framework. URL: www.django-rest-framework.org (дата обращения: 13.04.2020).

5. Docker documentation. URL: <https://docs.docker.com/> (дата обращения: 15.04.2020).

**АЛГОРИТМ НЕПАРАМЕТРИЧЕСКОГО ДУАЛЬНОГО УПРАВЛЕНИЯ
МНОГОМЕРНЫМИ ДИСКРЕТНО-НЕПРЕРЫВНЫМИ
ПРОЦЕССАМИ С ЗАПАЗДЫВАНИЕМ**

Д. А. Кемпф*

*Научный руководитель – Е. А. Чжан,
кандидат технических наук, доцент
Сибирский федеральный университет*

Предметом исследования данной статьи является создание самообучающегося адаптивного управляющего устройства. В статье предлагается непараметрический алгоритм дуального управления дискретными непрерывными процессами, когда исследователь не обладает достаточной априорной информацией об объекте для определения его параметрической структуры.

Теория дуализма в системах управления впервые была предложена А. А. Фельдбаумом [1]. Особенность дуального управления состоит в том, что управляющие воздействия имеют двойственный характер. Они, как писал А. А. Фельдбаум, «должны быть в известной мере изучающими, но, в известной мере, направляющими» [2].

Рассмотрим случай, когда уравнение, описывающее процесс, неизвестно. Тогда целесообразно использовать схему непараметрического дуального управления (рис. 1).

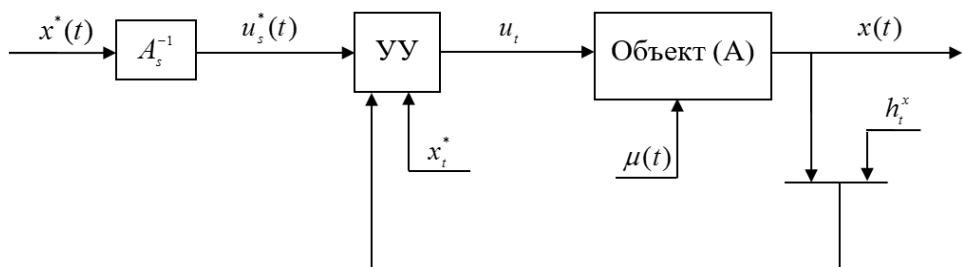


Рис. 1. Схема непараметрического дуального управления

Непараметрический алгоритм дуального управления может иметь вид [3]:

$$u_{s+1} = u_s^* + \Delta u_{s+1}, \quad (1)$$

где u_s^* – оценка обратного оператора A^{-1} по выборке наблюдений объёма s ; Δu_{s+1} – «поисковый» шаг.

* © Кемпф Д. А., 2020

В многомерном случае каждую компоненту управляющего воздействия u предлагается вычислять последовательно, по цепочке. Каждое последующее значение u_m зависит от величины u_{m-1} , найденной на предыдущем шаге:

$$u_{m,s}^* = \frac{\sum_{i=1}^s u_m^i \prod_{l=1}^{m-1} \Phi\left(\frac{u_l - u_l^i}{c_s^{u_l}}\right) \prod_{j=1}^n \Phi\left(\frac{x_{j,s+1}^* - x_j^i}{c_s^{x_j}}\right) \prod_{q=1}^k \Phi\left(\frac{\mu_{q,s+1} - \mu_q^i}{c_s^{\mu q}}\right)}{\sum_{i=1}^s \prod_{l=1}^{m-1} \Phi\left(\frac{u_l - u_l^i}{c_s^{u_l}}\right) \prod_{j=1}^n \Phi\left(\frac{x_{j,s+1}^* - x_j^i}{c_s^{x_j}}\right) \prod_{q=1}^k \Phi\left(\frac{\mu_{q,s+1} - \mu_q^i}{c_s^{\mu q}}\right)}, \quad (2)$$

где u_m^i – входное управляющее воздействие; x_j^i – выходное управляемое значение; x_j^* – желаемый результат управления; μ_q – неуправляемое, но измеряемое входное воздействие; $\Phi(\cdot)$ – ядерная колоколообразная функция; c_s^u , c_s^x и c_s^μ – коэффициенты размытости ядра [4].

Для вычисления «поискового» шага Δu_{s+1} может быть использована формула:

$$\Delta u_{s+1} = \gamma(x_{s+1}^* - x_s), \quad (3)$$

где γ – некоторый настраиваемый коэффициент.

Для вычислительного эксперимента был выбран объект, который описывается нелинейным уравнением:

$$x(t) = u_1^4(t) + 2u_1(t) + 2\sqrt{u_2(t)} - e^{u_3(t)} + \mu(t), \quad (4)$$

где $x(t)$ – выход объекта; $u_1(t)$, $u_2(t)$, $u_3(t)$ – входные управляющие воздействия. Добавим на вход объекта неуправляемую переменную: $\mu(t) = \sin(0,018t)$. Уточним, что данная информация об объекте известна лишь в рамках вычислительного эксперимента. Задающее воздействие представлено в ступенчатом виде, траекторией, случайным набором. Всё это описывается следующей системой уравнений:

$$\begin{cases} x^*(t) = 3, & \text{если } 0 < t \leq 20; \\ x^*(t) = 5, & \text{если } 20 < t \leq 40; \\ x^*(t) = 3,7 + \cos(0,5t), & \text{если } 40 < t \leq 80; \\ x^*(t) = \text{rand}(1; 12), & \text{если } 80 < t \leq 100; \\ x^*(t) = 3,5 + 0,7 \cos(0,7t) + \sin(0,2t), & \text{если } 100 < t \leq 150. \end{cases} \quad (5)$$

Задаём параметры для непараметрического алгоритма дуального управления: коэффициент для поискового шага $\gamma = 0,03$, начальные входные воздействия $u_0^1 = u_0^2 = u_0^3 = 1$. Результат работы алгоритма представлен на рис. 2.

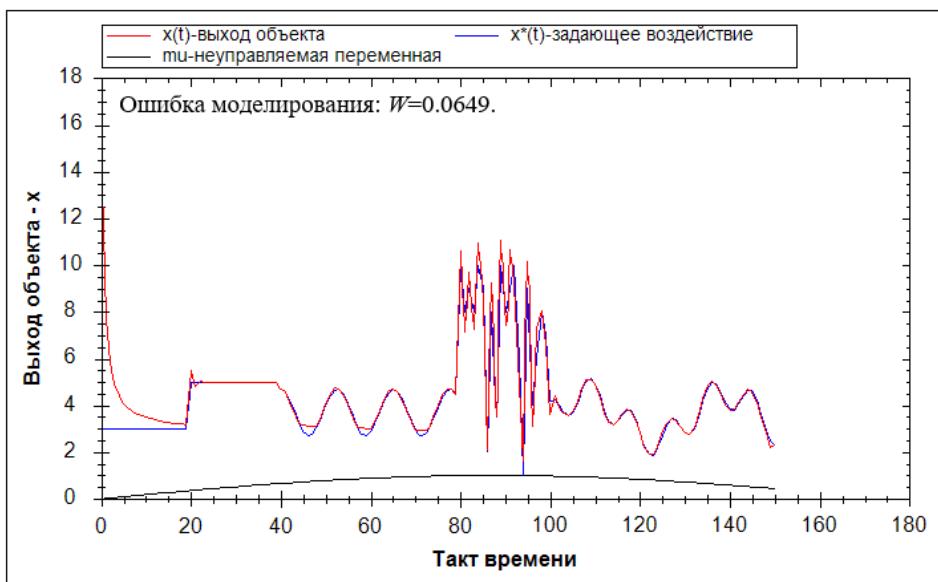


Рис. 2. График вычислительного эксперимента

Основной итог статьи состоит в том, что предложен непараметрический алгоритм управления многомерным объектом. Особенность данного алгоритма в том, что не требуется априорная информация о математической структуре управляемого объекта. Данный алгоритм показал свою эффективность в работе и имеет низкий показатель ошибки моделирования, о чём свидетельствует рис. 2.

Работа выполнена при финансовой поддержке гранта Президента Российской Федерации для государственной поддержки молодых учёных МК–763.2020.9.

Список литературы

1. Фельдбаум А. А. Теория дуального управления. Ч. 1: «Автомат. и Телемех.» / А. А. Фельдбаум. М.: Физматгиз, 1960. С. 1240–1249.
2. Фельдбаум А. А. Основы теории оптимальных автоматических систем / А. А. Фельдбаум. М.: Физматгиз, 1963. 552 с.
3. Медведев А. В. Основы теории адаптивных систем: монография / А. В. Медведев. Красноярск: СибГАУ, 2015. 526 с.
4. Корнеева А. А. Непараметрическое дуальное управление безынерционными системами / А. А. Корнеева, А. В. Медведев // Кибернетика и высокие технологии XXI века: труды XIII международной научно-технической конференции, г. Воронеж. 2013. С. 250–261.
5. Корнеева А. А. Об адаптивном управлении последовательностью технологических объектов / А. А. Корнеева, М. Е. Корнет, Н. А. Сергеева, Е. А. Чжан // Вестник СибГАУ им. М. Ф. Решетнёва. 2015. Т. 16, № 1. С. 72–78.

ИСПОЛЬЗОВАНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ ВЫБОРА СТРУКТУРЫ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ

Н. Е. Матюхин¹, Р. К. Абрамович²

*Научный руководитель – А. С. Михалёв,
старший преподаватель
Сибирский федеральный университет*

Искусственные нейронные сети – достаточно эффективный инструмент решения задач, сложных для решения алгоритмическим путем. Нейронные сети успешно применяются для решения задач распознавания образов, аппроксимации, прогнозирования, создания экспертных систем [1]. Они нашли широкое применение в нашей жизни, взяв на себя решение многих не самых простых задач [2]. Распознавание изображений, воспроизведение и перевод человеческой речи, обработка видеопотока в реальном времени – все эти задачи уже являются посильными для некоторых видов нейронных сетей.

Проблема подбора оптимальной архитектуры для конкретной нейронной сети составляет основную трудность при работе с ними [3]. Архитектура сети определяется числом нейронов на скрытых слоях, их активационными функциями, а также наличием связи между конкретными нейронами. Применение ГА обусловлено тем, что генетический алгоритм обеспечивает глобальный просмотр пространства параметров, позволяя избегать локальных минимумов, а также не требует информации о градиентах. В данной работе генетический алгоритм будет применён для решения задачи подбора архитектуры полно связной искусственной нейронной сети прямого распространения.

Типы операторов селекции (турнирная, элитарная, рулеточная), рассматриваемые в данной работе, обычно показывают себя одинаково при решении задачи подбора структуры искусственной нейронной сети [4]. В связи с этим было принято решение остановиться на гибридце рулеточного метода и элитарной селекции. Хромосомой в этой реализации генетического алгоритма является массив из чисел, отвечающих за число нейронов на конкретном скрытом слое нейронной сети. При проведении операции скрещивания возможны два разных исхода событий: длина хромосом родителей будет совпадать (одинаковое число скрытых слоёв) или будет различной.

В случае одинаковой длины хромосом применяется одноточечный кроссинговер без модификации, однако в случае, когда длины хромосом

* © Матюхин Н. Е.¹, Абрамович Р. К.², 2020

родителей не совпадают, нижняя часть архитектуры скрещивается по обычным правилам, а с вероятностью 50 % к потомку добавляется верхняя часть архитектуры более длинной хромосомы (рис. 1).

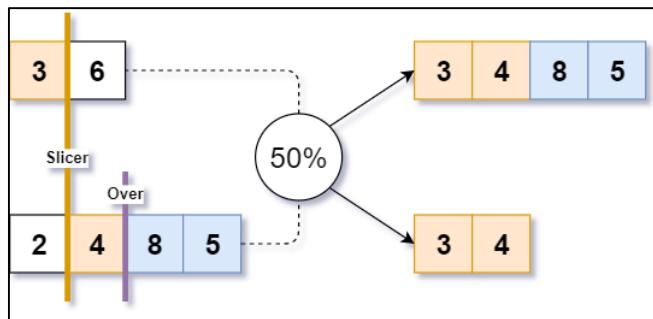


Рис. 1. Одноточечный кроссинговер с модификацией

Для исследования генетического подхода для настройки структуры нейронной сети был разработан модуль, реализующий искусственную нейронную сеть, а также генетический алгоритм на языке программирования *Python 3.6.9*.

В качестве исходной задачи была выбрана задача диагностики неисправностей синхронного двигателя на постоянных магнитах (*permanent magnet synchronous motor, PMSM* [5]).

Во всех примерах используется логистическая функция (сигмоида).

Таблица 1

Настраиваемые параметры генетического алгоритма

№	Настраиваемый параметр	Значение настраиваемого параметра
1	Оператор селекции	Гибрид рулеточного метода и элитарной селекции
2	Оператор скрещивания	Модифицированный одноточечный кроссинговер
3	Шанс мутации конкретного гена особи	0,5

Таблица 2

Настраиваемые параметры искусственной нейронной сети

№	Настраиваемый параметр	Значение настраиваемого параметра
1	Максимальное число скрытых слоёв	[1, 3]
2	Максимальное число нейронов на каждом слое	[1, 30]

Каждая нейронная сеть обучалась с использованием метода обратного распространения ошибки с коэффициентом скорости обучения η , равным 0,01. Размер батча – 200, количество эпох – 50.

В результате работы генетического алгоритма с размером популяции в 100 особей и количеством эпох 50 была получена особь, показавшая

наилучшие результаты предсказания нужного значения среди всех имеющихся особей на каждой эпохе.

Результаты предсказания нейронной сети с архитектурой, полученной в ходе запуска генетического алгоритма, представлены на рис. 2.

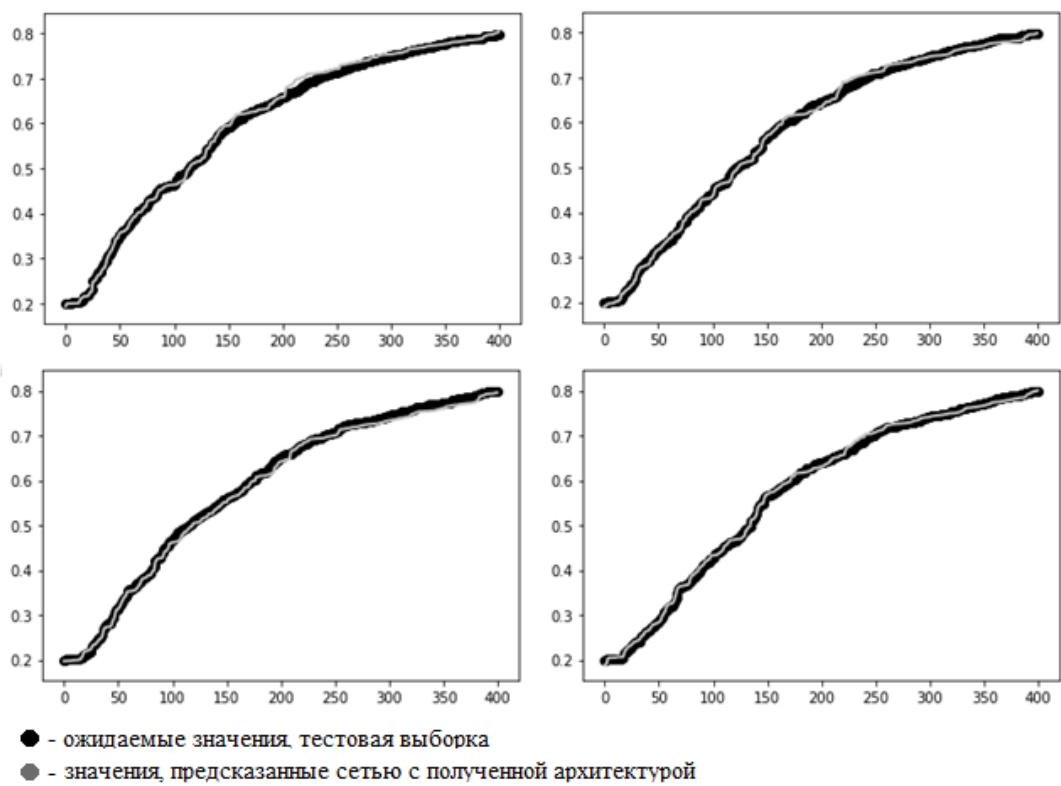


Рис. 2. Значение температуры $PMSM$

Сеть с подобранный генетическим алгоритмом архитектурой показала 0,4 % ошибки на тестовой выборке, что является отличным результатом для данной задачи [1], а поиск оптимальной архитектуры в этом случае не превышал 4 часов реального времени.

Список литературы

1. О применении параллельных генетических алгоритмов для автоматизации нейросетевого моделирования / С. Н. Ефимов, Е. С. Семёнкин, В. С. Тынченко, В. В. Тынченко // Сибирский журнал науки и технологий. 2009. № 6. С. 13.
2. Arifovic A. Using genetic algorithms to select architecture of a feed-forward artificial neural network / A. Arifovic, R. Gencay // Physica A: Statistical mechanics and its applications. 2000. С. 289.
3. Сараев П. В. Конструктивный подход построения нейроструктурных моделей на основе блочного псевдообращения / П. В. Сараев // Вестник ВГТУ. 2012. Т. 8. № 7–1. С. 46–49.

4. Истомина Ю. А. Особенности процесса проектирования нейросетевых моделей произвольной архитектуры / Ю. А. Истомина // Актуальные проблемы авиации и космонавтики. 2014.

5. Electric Motor Temperature: «Kaggle: Your home for Data Science». URL: <https://www.kaggle.com/wkirgsn/electric-motor-temperature>.

3D-МОДЕЛЬ СИСТЕМЫ УПРАВЛЕНИЯ ОРИЕНТАЦИИ СОЛНЕЧНОЙ БАТАРЕИ В СРЕДЕ SIMINTECH

Л. А. Михайленко*

**Научный руководитель – А. В. Чубарь,
кандидат технических наук, доцент
Сибирский федеральный университет**

В настоящий момент актуальность солнечной энергетики возрастает. Это связано с экологически чистой энергией, а также с ресурсоёмкостью. Использование солнечной энергии не оказывает влияния на окружающую среду и является очень выгодным.

Установить солнечные батареи для выработки энергии можно на собственной крыше или на прилегающем участке.

Прогнозирование эффективности использования батарей является трудоёмким процессом. Поэтому для более лёгкого расчёта получаемой энергии и наглядного примера слежения батареи используют модуль 3D в среде визуального моделирования *SimInTech* [1].

Одна из ведущих особенностей *SimInTech* – применение «Базы данных», «Базы сигналов», которые облегчают обмен данных между расчётными схемами в единой модели.

Модель слежения солнечной батареи в среде *SimInTech* состоит из проектов: модель управления солнечной панели (рис. 1), модель управления траекторией движения Солнца (рис. 2) и модуль визуализации 3D. Солнечная панель, в свою очередь, включает в себя две подсубмодели: управление углом относительно «оси Y» и управление углом относительно «оси X» (рис. 1). Сигналы, поступающие в модель процесса управления, обрабатываются субмоделью и сохраняются. Полученный результат выводится на экран в виде графика.

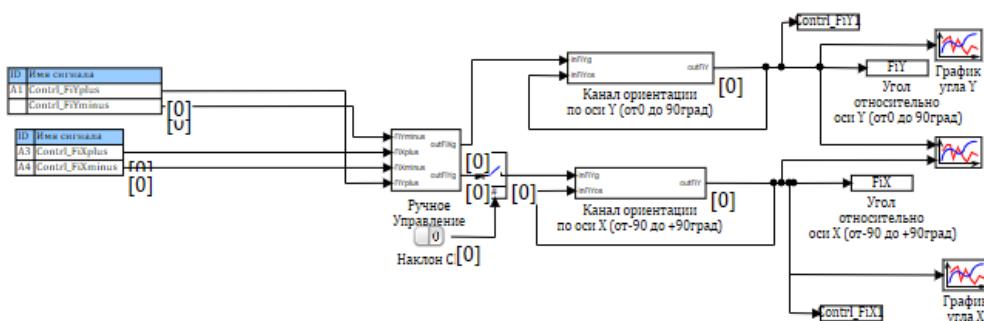


Рис. 1. Модель управления солнечной батареи
в среде для динамического моделирования *SimInTech*

Солнечная батарея, получая данные о местоположении Солнца, производит необходимый поворот и угол наклона [2].

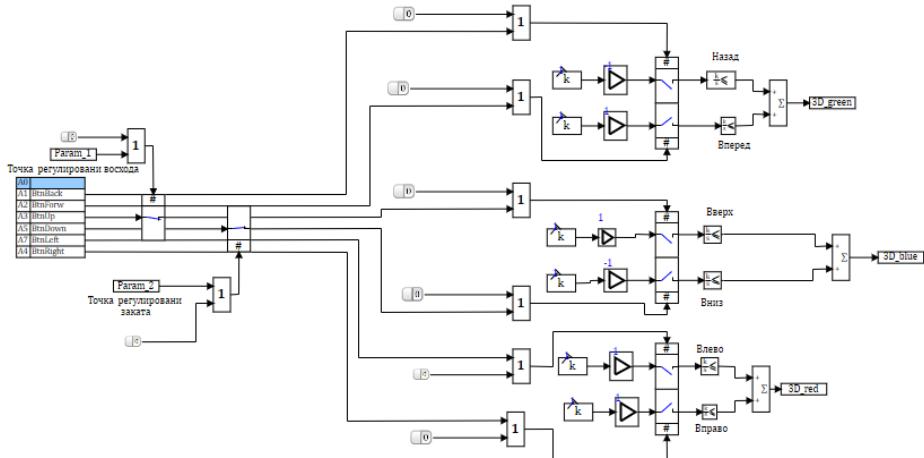


Рис. 2. Модель управления Солнцем
в среде для динамического моделирования *SimInTech*

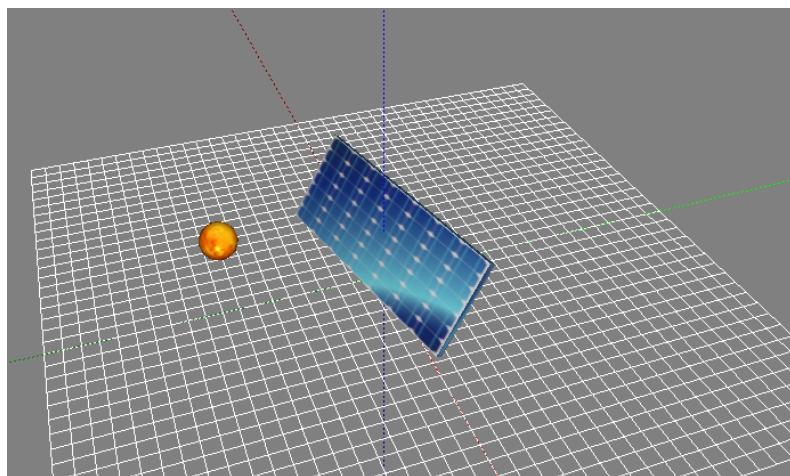


Рис. 3. Субмодель. Пример работы пакета в среде моделирования *SimInTech*

Для подтверждения высокой производимости солнечной батареи производится расчёт процента получаемой энергетической мощности.

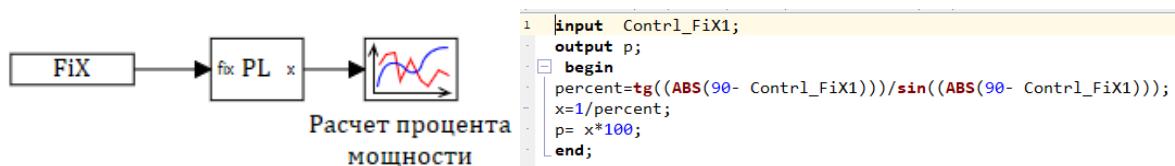


Рис. 4. Субмодель реализации расчёта процента мощности
угла наклона солнечной батареи в среде моделирования *SimInTech*

Входными данными для определения процента получаемой мощности является сигнал о состоянии угла наклона солнечной батареи.

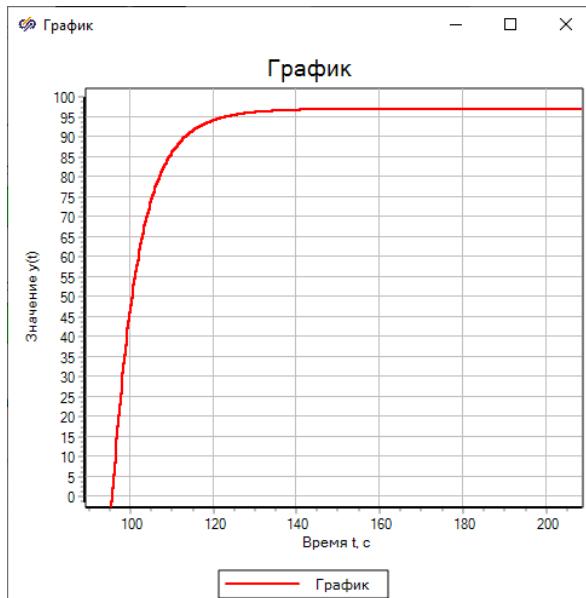


Рис. 5. Результат расчёта

Таким образом, разработанная на основе типовых блоков *SimInTech* модель солнечной батареи и Солнца позволяет визуально отследить высокоэффективность использования оборудования.

Список литературы

1. Среда динамического моделирования технических систем *SimInTech* / Б. А. Карташов, О. С. Козлов, Е. А. Шабаев и др. М.: ДМК Пресс, 2017. 424 с.
2. Аржанов К. В. Двухкоординатная система наведения солнечных батарей на Солнце / К. В. Аржанов // Известия ТПУ. 2014. Т. 324. № 4. С. 139–146.

ПОСТРОЕНИЕ СИСТЕМЫ УПРАВЛЕНИЯ ДВИЖЕНИЕМ МОБИЛЬНЫМ РОБОТОМ В СРЕДЕ SIMINTECH

В. А. Мызникова*

*Научный руководитель – А. В. Чубарь,
кандидат технических наук, доцент
Сибирский федеральный университет*

Изучение мобильных роботов стимулируется многочисленными их приложениями в различных областях, где жизнедеятельность человека либо затруднена, либо вообще невозможна, например, [1].

Для выполнения своих функциональных задач роботы должны уметь автономно перемещаться по траекториям, заданным оператором. Математическое моделирование поможет сэкономить материальные и трудовые ресурсы при разработке алгоритмов управления и проведении испытаний робота [2].

Конструкция робота включает основание и двигательную систему, которая обеспечивает необходимую траекторию, а также скорость перемещения корпуса робота. Предположим, что робот состоит из абсолютно твёрдой платформы и имеет 4 колеса, которые могут вращаться в прямом или обратном направлениях.

Достаточно очевидными являются следующие суждения:

- 1) для прямолинейного движения колёса должны вращаться с одинаковыми угловыми скоростями;
- 2) для того чтобы робот развернулся на месте, необходимо установить угловые скорости одинаковыми по модулю, но противоположными по направлению, т. е. знаку;
- 3) другие комбинации угловых скоростей приводят к движению по дуге.

Пусть робот движется с разными по модулю, но одинаковыми по знаку угловыми скоростями колёс. Изобразим движение робота на рис. 1.

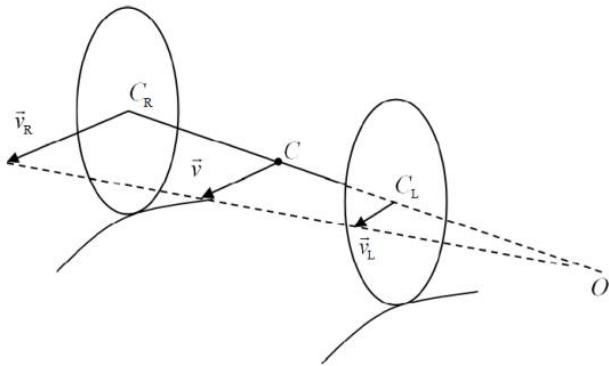


Рис. 1 Движение робота при однонаправленных угловых скоростях колёс

Угловая скорость движения робота определяется так:

$$\omega = \frac{U_R - U_L}{D} = \frac{\omega_R - \omega_L}{D} \times r. \quad (1)$$

Здесь r – радиус колеса робота; U_L и ω_L – линейная и угловая скорости левого колеса; U_R и ω_R – линейная и угловая скорости правого колеса; D – расстояние между точками C_L и C_R . Если угловые скорости колёс постоянны, то угловая скорость движения робота также постоянна, что соответствует движению робота по окружности.

Обозначим теперь расстояние от центра окружности O до центра робота C как R . Тогда расстояние от центра окружности до центра правого колеса будет равно $R + 0,5 \times D$. Отсюда получим:

$$\frac{U_R}{R + 0,5 \times D} = \omega = \frac{U_R - U_L}{D}. \quad (2)$$

Отсюда получим выражение для радиуса окружности R :

$$R = \frac{D}{2} \times \frac{U_R + U_L}{U_R - U_L} = \frac{D}{2} \times \frac{\omega_R + \omega_L}{\omega_R - \omega_L}. \quad (3)$$

В частном случае, если угловые скорости колёс совпадают, то угловая скорость робота равна нулю, радиус окружности стремится к бесконечности. Когда скорость правого (левого) превышает скорость левого (правого) колеса, то робот движется против (по) часовой стрелки и угловая скорость ω получается положительной (отрицательной). Это верно и для положительных, и для отрицательных скоростей колёс.

Если угловые скорости колёс имеют разные знаки, то угловая скорость движения робота определяется выражением (1). Траекторией также является окружность с вычисляемым по формуле (3) радиусом, но её центр расположен под платформой, между колёсами робота, ведь радиус получается меньше величины $0,5 \times D$.

Расчёт угловой скорости и радиуса окружности по формулам (1) и (3) организуем с помощью блоков «Перемножитель», «Сумматор», «Сравнивающее устройство», «Усилитель», «Абсолютное значение» из вкладки «Операторы». Радиус колёс зададим с помощью блока «Константа» из вкладки «Источники». Угловые скорости для расчёта значений будут поступать из базы данных. Создадим в базе данных группу *SkKol* и сигналы *wR* и *wL*. Таким образом, угловой скорости левого колеса будет соответствовать сигнал *SkKol_wL*, а правого – *SkKol_wR*. Для их ввода в схему будем использовать блок «Чтение» из списка сигналов из вкладки «Сигналы».

Кроме этого, в базу данных нужно записывать рассчитанные значения угловой скорости робота и радиуса окружности, по которой он движется. Для этого будем использовать блок «Запись в список сигналов» из вкладки «Сигналы». Для радиуса окружности создадим группу сигналов *CM* и сигнал *Rkr* в этой группе. Для угловой скорости создадим в той же самой группе *CM* сигнал *w*. Радиус колес примем равным 0,028 м, а расстояние между колёсами – 0,16 м.

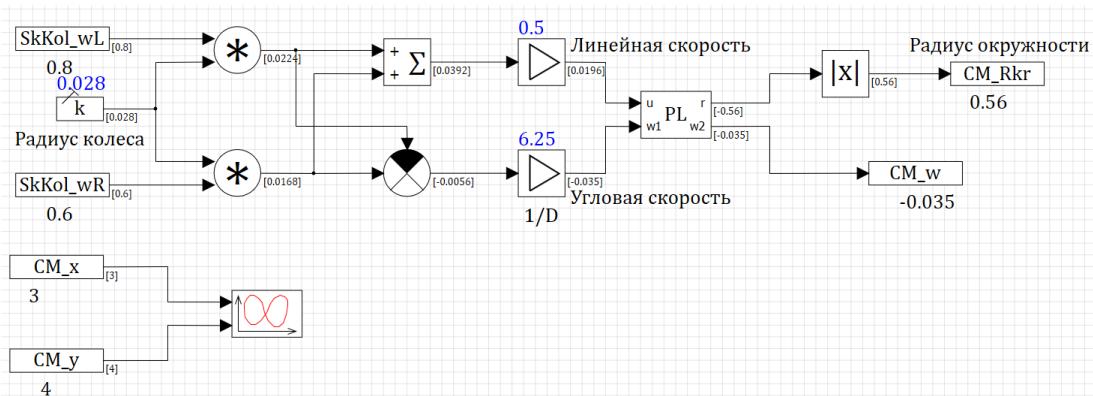


Рис. 2. Схема для расчёта радиуса и угловой скорости и вывода траектории в среде моделирования *SimInTech*

Для построения траектории будем использовать блок «Фазовый портрет» из вкладки «Вывод данных», который позволяет строить график по поступающим на его вход координатам *x* и *y*. Эти координаты мы будем рассчитывать с помощью скрипта и передавать в схему через базу данных. Для координаты *x* создадим в группе *CM* сигнал *x*, для координаты *y* – сигнал *y*. Блок «Программирование» обеспечивает исключение деления на ноль в формуле (3) при равенстве угловых скоростей правого и левого колёс.

Текущая ориентация робота отображается в окне анимации с помощью стрелочного прибора, если настроить в нём угол охвата шкалы, равный 2π . Нижний предел равен 0, а верхний, соответственно, 2π .

Список литературы

1. Звонарёв Д. А. Управление мобильным роботом с электрическим двигателем / Д. А. Звонарёв // Известия ТулГУ. 2011. № 2. С. 368–372.
2. Бартенёв В. В. Математическая модель движения мобильного робота с двумя независимыми ведущими колёсами по горизонтальной плоскости / В. В. Бартенёв, С. Ф. Яцун, А. С. Аль-Еззи // Известия Самарского научного центра РАН. 2011. Т. 13. № 4. С. 288–293.
3. Карташов Б. А. Среда динамического моделирования технических систем SimInTech: практикум по моделированию систем автоматического регулирования / Б. А. Карташов, Е. А. Шабаев, О. С. Козлов, А. М. Щекатуров. М.: ДМК Пресс, 2017. 424 с.
4. Мызникова В. А. Построение системы управления движением автономного объекта в среде SimInTech / В. А. Мызникова, Е. Д. Толстенков, А. В. Чубарь // Робототехника и искусственный интеллект: материалы XI Всероссийской научно-технической конференции с международным участием. 2019.

МОДЕЛИРОВАНИЕ ПОКАЗАТЕЛЕЙ ЭФФЕКТИВНОСТИ ОБРАЗОВАТЕЛЬНОЙ ДЕЯТЕЛЬНОСТИ ВЫСШЕГО УЧЕБНОГО ЗАВЕДЕНИЯ

С. В. Сергиенко*

*Научный руководитель – А. А. Корнеева,
кандидат технических наук, доцент
Сибирский федеральный университет*

В настоящее время особое внимание уделяется реформированию системы образования, а именно повышению его эффективности. Повышение эффективности образования заставляет высшие учебные заведения (вузы) искать и внедрять новые методы и технологии, обеспечивающие конкурентные преимущества на рынке образовательных услуг. Для оценки эффективности образовательной деятельности аналитическим путём, на основе системного анализа, выявляется набор показателей, с помощью которого может проводиться оценивание какого-либо свойства, вырабатывается система шкал оценивания и сами критерии оценки [1, 2, 4]. Осуществляется моделирование показателей эффективности образовательной деятельности вуза посредством математических моделей, методов и автоматизированных систем [2, 3, 4]. В настоящей работе представлен обзор научно-исследовательских работ, направленных на повышение эффективности системы образования, а также сформулирована постановка задачи для дальнейшего исследования.

В соответствии с действующим положением вуз обязан периодически проходить лицензионную и аккредитационную экспертизу, по результатам которой определяется его статус (университет, академия, институт) и принимается решение о продолжении деятельности в сфере образования. Имеются критериальные показатели аккредитации для вузов различных типов. В работе [2] рассмотрено моделирование показателей аккредитации вуза для эффективного контроля образовательной деятельности. Существующая процедура лицензирования и аккредитации вузов РФ обладает рядом особенностей, снижающих эффективность процесса контроля и управления качеством образовательного процесса. Практически не учитываются неизбежные изменения показателей аккредитации вуза между двумя экспертизами, вызванные воздействием внешних факторов. Оценка эффективности функционирования вуза представляется недостаточно достоверной, особенно с учётом человеческого фактора и тщательной подготовки, предшествующей проведению экспертизы. Существующий методоло-

* © Сергиенко С. В., 2020

гический аппарат не даёт возможность осуществить прогноз этих показателей на интервале между аккредитациями, что не позволяет руководству вуза своевременно устранить возникающие негативные тенденции и уменьшает практическую ценность проведённой экспертизы.

Указанные обстоятельства обуславливают необходимость разработки и внедрения новых математических моделей и алгоритмов, позволяющих осуществлять оценку основных показателей вуза на всём интервале его аккредитации, что позволит преодолеть указанные трудности.

В работе [2] для наиболее точного описания взаимосвязи элементов в образовательном процессе предложена методика, основанная на принципе системной динамики – методе изучения сложных систем с нелинейными обратными связями Дж. Форрестера, которая позволяет осуществлять эффективный контроль качества образовательного процесса вуза. Для оценки адекватности модели предложено использовать аппарат регрессионного анализа. Для автоматизированной системы контроля качества образовательного процесса вуза в среде *GUIDE MatLab Version 7.6.0.324 (R2008a)* был разработан программный комплекс «*Inform_System_CQEP*», позволяющий осуществить оперативное управление данным процессом.

Разработанная математическая модель позволяет более полно учесть особенности образовательного процесса и оперативно управлять его качеством на временных интервалах между аккредитациями.

В работе [1] рассмотрено моделирование показателей эффективности образования на основе компетентностного подхода. Образовательный процесс рассматривается как процесс приобретения компетенций вследствие изучения дисциплин в составе образовательной программы. Поэтому оценка эффективности полученного образования при компетентностном подходе заключается в оценке уровня сформированности заявленного набора компетенций.

В работе [1] на примере оценки процесса приобретения компетенций по направлению подготовки бакалавров 230100.62 Информатика и вычислительная техника:

- проанализированы ФГОС ВО 3-го поколения;
- определены направления модернизации образования при переходе к этим стандартам;
- выделены специальные компетенции специалиста;
- выработаны критерии оценки уровня сформированности компетенций с использованием методов когнитивного моделирования, которые позволяют строить динамические модели роста компетенций в зависимости от управляющих факторов. Отметим, что компетенции накапливаются по семестрам достаточно равномерно, что свидетельствует о достаточно рациональном планировании учебного процесса.

Одним из подходов повышения эффективности образовательной деятельности является обеспечение заданного уровня качества процесса обучения, которое оценивается успеваемостью студентов. В работах [5, 6] предлагается моделирование информационного взаимодействия со студен-

тами технических направлений подготовки на основе показателей успеваемости студентов. Одним из инструментов захвата внимания студентов и влияния на их успеваемость может стать персональная образовательная среда (ПОС). ПОС студента СФУ включает в себя: систему электронных курсов вуза, сайт вуза, сайт института, электронную библиотеку, мессенджеры, социальные сети, внешние электронные курсы и др.

Для этого определялась зависимость между использованием студентом ПОС и его успеваемостью в учебном процессе методами кластеризации и математической статистики.

Результаты проведённого исследования [5, 6] показали, что активная работа в ПОС связана с высокими показателями успеваемости студентов. Кластерный анализ позволил выделить четыре группы студентов с различными профилями показателей успеваемости. Обоснована возможность гибкого применения нового инструментария ПОС (рейтинги, достижения, значки, активные уведомления) для повышения успеваемости студентов из разных кластеров. Показана возможность и целесообразность использования ПОС как инструмента повышения успеваемости студентов технических направлений подготовки.

Планируется провести работу по моделированию ПОС для повышения показателей успеваемости студентов гуманитарных направлений подготовки.

Для исследования изучения взаимосвязи использования ПОС и показателей успеваемости студентов будут измеряться следующие показатели:

- активность студента в ПОС;
- средняя оценка по дисциплинам промежуточной аттестации;
- среднее число пересдач по сформировавшимся задолженностям в результате неуспешного прохождения промежуточной аттестации;
- общий результат промежуточной аттестации;
- отчисление за неуспеваемость.

Таким образом, планируется определить связь между успеваемостью студентов гуманитарных направлений подготовки в процессе обучения и использованием ими ПОС.

Список литературы

1. Инновационная деятельность в системе образования /
3. А. Аксютина, Н. А. Бирюкова, З. В. Никонова, Г. А. Доррер и др. М.: Перо, 2013. 203 с.
2. Яндыбаева Н. В. Математическая модель для прогнозирования показателей аккредитации вуза / Н. В. Яндыбаева, В. А. Кушников // Управление большими системами. 2013. № 41. С. 314–343.
3. Яндыбаева Н. В. Разработка программного продукта для определения эффективности деятельности высшего учебного заведения /

Н. В. Яндыбаева, Е. Р. Кожанова, В. А. Кушников // Вестник СГТУ. 2014. № 2. С. 214–219.

4. Гельруд Я. Д. Математическая модель управления образовательным процессом / Я. Д. Гельруд, Д. А. Стайн // Управление инвестициями и инновациями. 2017. № 4. С. 14–21.

5. Якунин Ю. Ю. Персональная образовательная среда в системе управления институтом / Ю. Ю. Якунин, А. К. Погребников // Информатика и образование. 2017. № 2. С. 55–60.

6. Погребников А. К. Персональная образовательная среда как инструмент повышения успеваемости студентов / А. К. Погребников, В. Н. Шестаков, Ю. Ю. Якунин // Перспективы науки и образования. 2019. № 6 (42). С. 473–483.

УПРАВЛЕНИЕ МОЩНОСТЬЮ ПРОИЗВОДСТВЕННОГО УЧАСТКА

Р. Н. Соболев*

*Научный руководитель – Е. А. Чжан,
кандидат технических наук, доцент
Сибирский федеральный университет*

Производственная мощность является особым товаром, покупка которого предоставляет участнику оптового рынка право требования к выработке продукта установленного качества в количестве, необходимом для удовлетворения потребности данного участника [3]. Мощность предприятия определяется по мощности ведущих цехов, участков, агрегатов. К ведущим относятся цеха, участки, агрегаты, в которых выполняются основные наиболее трудоёмкие технологические процессы и операции по изготавлению изделий или полуфабрикатов [1].

Торговля мощностью является надёжным и бесперебойным источником для поставки товара. Покупатели должны обеспечить покрытие не только затрат производителей мощности на фактическое производство и отпуск продукта, но и затрат на улучшение производства, а также на возможность для покупателей получить интересующий их товар в большем количестве [2]. Если готовность не будет обеспечена, возникает риск недостачи товара вплоть до потери потребителей, неустойка, а также издержки производства. Таким образом, контролирование мощности является тем механизмом, с помощью которого компании имеют возможность возмещать свои затраты на производство, а также подстраивать его под спрос рынка.

Одним из способов контролировать мощность производства являются инструменты, которые могут отслеживать её изменения в течение дня и накапливать статистику. Они позволяют принимать решения о том, чтобы передавать незанятую мощность под плановые потребности потребителей. В качестве инструмента, который может выполнять данные функции, выступает дашборд. Дашборд – это интерфейс между аналитическим движком и тем, кто выступает в роли пользователя-аналитика [5].

Данный способ контроля мощности был применён на одном из производственных предприятий г. Красноярска. Дашборд был разработан для решения ситуаций с повторной продажей уже реализованной в продажу мощности, а также с целью показать лицу, принимающему решения, информацию о проданной мощности. Проблема заключалась в том, что показатели мощности обновлялись несколько раз в течение рабочего дня и в

* © Соболев Р. Н., 2020

промежутках времени было сложно отследить вручную, кто и сколько купил мощности. Из-за чего аналитик – контролёр мощности – не успевал принимать решения по ограничению доступности мощности по определённым агрегатам. Поэтому могли случаться ситуации, при которых часть объёма мощности продавалась нескольким разным потребителям – вследствие чего производящая компания могла нести дополнительные траты из-за невыполнения обязательств по срокам производства продукции.

Для более быстрого создания и ввода в работу такого инструмента была применена продуктовая модель разработки. Данная методология позволяет выявить наиболее важные места для пользователей, которые требуют более проработанного решения, и отобразить их (рис. 1) [4].

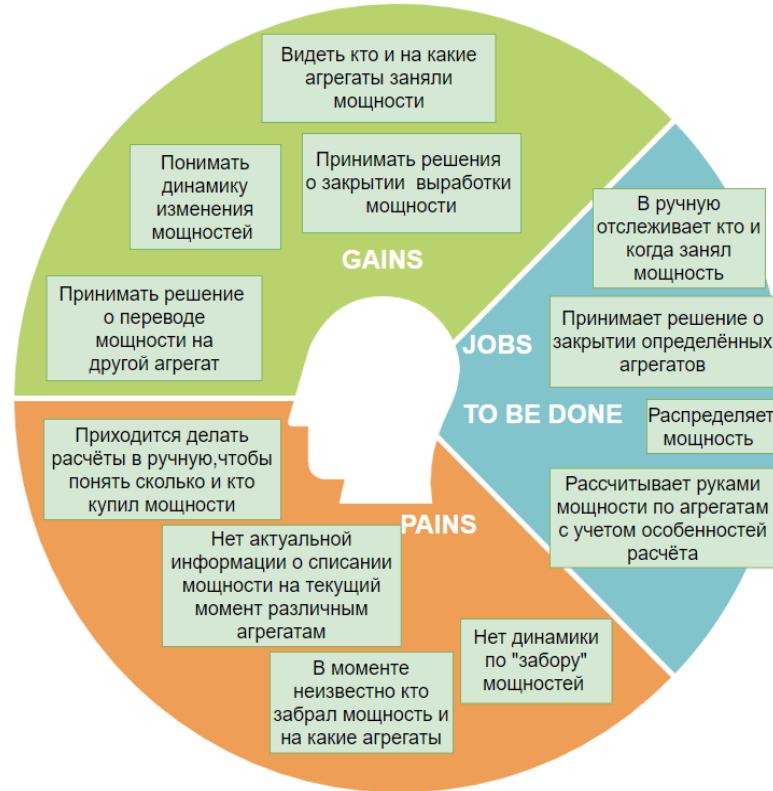


Рис. 1. Аналитик – контролёр мощности

При продаже мощности дэшборд получает данные: название станка, объём проданной мощности, тип сырья, тип проданной мощности, индивидуальный номер покупателя, наименование покупателя, дата покупки, тип изделия, характеристики изделия. Далее ведётся их предобработка и приведение в соответствие с нужными типами данных. Затем информация ставится в соответствие с информацией, полученной из производственных цехов, и рассчитывается процент проданной мощности в текущий момент времени. Информация обновляется каждые 5 минут и сохраняется в базе данных. Сохранение информации в базе позволяет строить динамику продажи мощности, например, как представлено на рис. 2.

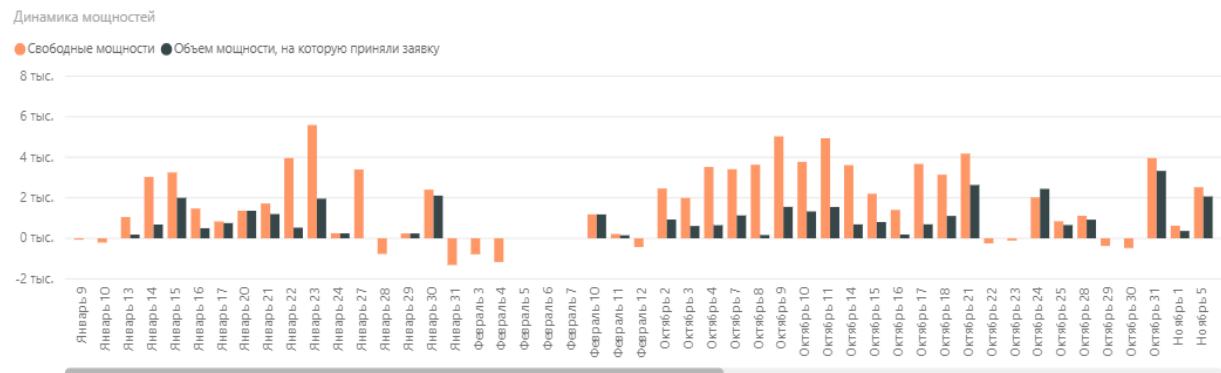


Рис. 2. Динамика мощности

В настоящее время улучшение использования производственных мощностей связано с повышением качества и конкурентоспособности продукции, улучшением маркетинговой деятельности, расширением сбыта продукции – и аналитические панели, и дашборды выступают инструментом для более эффективного управления производственными показателями.

Список литературы

1. Фролова Т. А. Экономика предприятия / Т. А. Фролова. Таганрог: ТТИ ЮФУ, 2009. С. 15.
2. Торговля мощностью – что это? URL: <https://www.vegaslex.ru/analytics/publications/31885/>.
3. П. 36 Правил оптового рынка электрической энергии (мощности) переходного периода (утверждены Постановлением Правительства РФ от 31.08.2006 № 529) // СЗ РФ. 04.09.2006. № 36. Ст. 3835.
4. Product Architecture Framework. URL: <http://productframework.ru/>.
5. Few S. Information dashboard design: The effective visual communication of data / S. Few. O'Reilly Media, Inc., 2006.

**МАТЕМАТИЧЕСКАЯ МОДЕЛЬ
НА ОСНОВЕ АЛГОРИТМА СЛУЧАЙНОГО ЛЕСА
ДЛЯ ПРЕДСКАЗАНИЯ УСПЕШНОСТИ ОБУЧЕНИЯ**

А. С. Стебеньков*

*Научный руководитель – Т. А. Кустацкая,
кандидат физико-математических наук
Сибирский федеральный университет*

Потребность в анализе данных вышла далеко за пределы технологических и интернет-компаний, методы машинного обучения всё активнее используются в различных областях, решая разнообразные прикладные задачи. Одна из таких областей применения – образование. Анализ образовательных данных и использование моделей для принятия решений, построенных с использованием этих данных, является глобальной тенденцией образования в последнее время [1]. Специалисты в данной области пытаются извлечь закономерности из данных для дальнейшей оптимизации траекторий обучения и итоговых оценок у студентов [2]. Они помогают эффективно выстроить образовательный процесс, качественно отслеживать прогресс обучения и достичь лучших результатов у учащихся.

Цель исследования – разработать математическую модель для предсказания успешности сдачи экзамена студентом по дисциплине «Теория вероятности и математическая статистика» на основе алгоритма случайного леса. В рамках исследования считаем, что экзамен сдан успешно («успешность = 1»), если студент получил оценку «удовлетворительно», «хорошо» или «отлично»; и неуспешно («успешность = 0»), если студент сдал на «неудовлетворительно» или не явился на экзамен. Основные задачи: анализ имеющихся данных, их предобработка, применение алгоритма случайного леса для построения математической модели и анализ качества полученной модели и результатов работы.

Данные для анализа были собраны в рамках исследования по прогнозированию успешности обучения студентов с помощью байесовской сети [3]. Они были представлены в виде таблицы, содержащей сведения об успеваемости студентов. Среди признаков имелись:

- отметка о посещении конкретной лекции;
- отметка о посещении конкретного практического занятия;
- отметка о получении плюса за работу на практическом занятии;
- количество баллов за 1-ю, 2-ю и 3-ю контрольную работу;
- количество баллов за задания, выполненные в электронных курсах;

* © Стебеньков А. С., 2020

- количество попыток прохождения заданий из электронного курса;
- итоговая оценка за экзамен.

Учитывая, что для каждого студента известна его итоговая оценка, задача представляет собой задачу обучения с учителем: весь набор данных делится на обучающую и тестовую выборку. На обучающей выборке модель настраивается таким образом, чтобы она выдавала прогноз, максимально приближённый к истинным значениям успешности (оценкам, реально полученным на экзамене и переведённым в бинарную шкалу). Для оценки полученной модели делаем предсказания для тестовой выборки, сравниваем их с истинными значениями и получаем вывод о точности работы. Перед построением модели была произведена предобработка данных. После данные разделили на обучающую и тестовую выборку в соотношении 70 на 30 % соответственно.

В качестве метода для построения модели был выбран алгоритм случайного леса [4]. Выбор был сделан, потому что данный алгоритм обладает привлекательными свойствами, подходящими для данной задачи [5]: обеспечивает существенное повышение точности за счёт ансамбля деревьев по сравнению с решающими деревьями и отсутствие проблемы переобучения. Также данный алгоритм успешно применяют в области анализа образовательных данных при оценке качества учебного процесса [6]. Построение моделей осуществлялось с использованием языка программирования *Python 3.5* и библиотек *sklearn*. Для анализа алгоритма было произведено 15 случайных разделений выборки на обучающие и тестовые данные. Для каждого разбиения были обучены 17 моделей. Так, i -я модель строится по тем признакам, значения которых становятся известными к концу i -й недели обучения.

Для оценки качества прогноза для каждой недели обучения были вычислены средние значения стандартных метрик качества классификации (*accuracy*, *sensitivity*, *specificity*, *precision*, *f-1 score*, *f-beta score*). Результаты представлены на рис. 1. Таким образом, можно увидеть, что чем больше прошло недель, тем лучше качество. До четвёртой учебной недели качество прогноза неприемлемо, потому что значения метрик находятся ниже 0,7. Сама модель перестраховывается и склонна к большей осторожности, относя людей, сдавших экзамен, к противоположному классу, – это можно увидеть по малым значениям метрики *specificity*. Уже после четвёртой учебной недели модель начинает лучше оценивать связь между признаками и качество прогноза возрастает. Интересный момент можно увидеть на седьмой неделе: метрика *sensitivity* начинает принимать значения, близкие к единице, это отражает существенную долю положительных результатов, которые правильно определены как таковые. Наиболее значимой метрикой, с точки зрения решения задачи выявления слабых студентов, является метрика *f-1 score* (среднее гармоническое между *precision* и *sensitivity*), которая, начиная с шестой недели, принимает значение, близкое к 0,8, и с увеличением номера учебной недели и, соответственно, количества полученной о студентах информации продолжает расти.

В итоге для конкретной дисциплины было продемонстрировано, что модель, основанная на алгоритме случайного леса, показывает хорошие результаты, начиная с середины семестра. Модель можно применить в рамках проведения этого курса для выявления слабых студентов, чтобы вовремя им помочь в освоении предмета и лучше подготовить их к экзамену. Для улучшения качества прогноза предполагается сделать следующее:

- расширить данные добавлением новых студентов, изучавших данный предмет;
- добавить новые признаки, такие как оценки по другим математическим дисциплинам;
- подобрать и найти оптимальные параметры алгоритма.

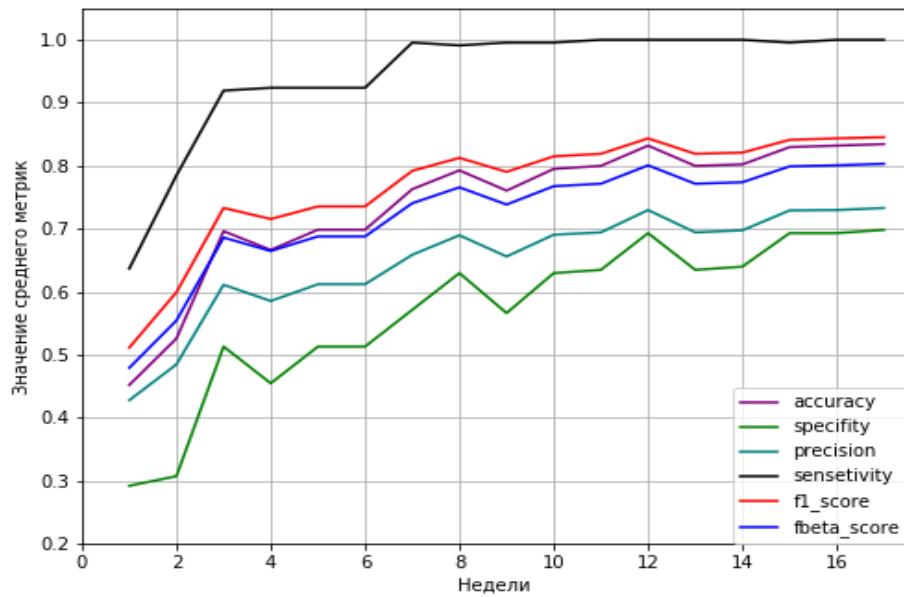


Рис. 1. График понедельного изменения значений метрик качества

Список литературы

1. Белоножко П. П. Анализ образовательных данных: направления и перспективы применения / П. П. Белоножко, А. П. Карпенко, Д. А. Храмов // Интернет-журнал «Науковедение». 2017. Т. 9. № 4.
2. Носков М. В. Управление успешностью обучения студента на основе марковской модели / М. В. Носков, М. В. Сомова, И. М. Фёдорова // Информатика и образование. 2018. № 10. С. 4–11.
3. Кустицкая Т. А. Прогнозирование успешности обучения студента с помощью байесовской сети / Т. А. Кустицкая // Информатизация образования и методика электронного обучения. 2019. С. 257–262.
4. Breiman L. Random forests / L. Breiman // Machine Learning. 2001. Vol. 45. № 1. P. 5–32.
5. Чистяков С. П. Случайные леса / С. П. Чистяков // Труды Карельского научного центра РАН. 2013. № 1. С. 117–136.
6. Буре Н. А. Применение статистических методов при оценке качества учебного процесса / Н. А. Буре, Н. Л. Гребенникова, К. Ю. Староверова // Вестник СПбГУ. 2018. С. 325–331.

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ОТЛАДКИ ВСТРОЕННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СИСТЕМ УПРАВЛЕНИЯ КОСМИЧЕСКИХ АППАРАТОВ

Е. В. Табаков, А. И. Зинина, Д. С. Киреев*

**ФГУП МОКБ «Марс»
НИУ МАИ**

Создание системы управления (СУ) космического аппарата (КА) остаётся весьма трудной и актуальной задачей. Данная система является наиболее важной системой КА, и от её работы зависит функционирование всех остальных систем. Ниже представлена типовая упрощённая структурная схема КА (рис. 1).

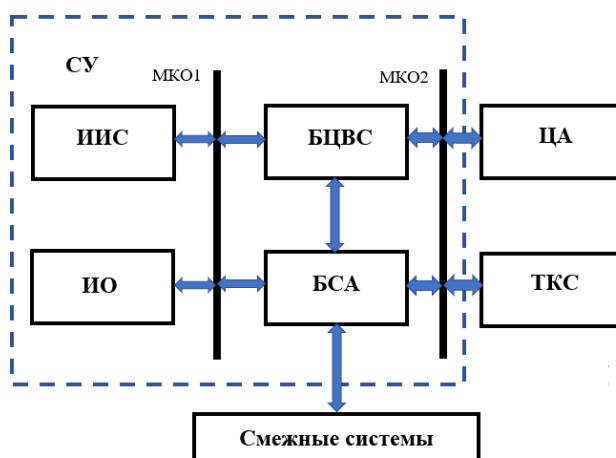


Рис. 1. Упрощённая структурная схема КА

Бортовая цифровая вычислительная система (БЦВС) является основным блоком СУ. В зависимости от показаний информационно-измерительной системы (ИИС) и команд с Земли, получаемых с помощью телекомандной системы (ТКС), БЦВС передаёт команды на целевую аппаратуру (ЦА), исполнительные органы (ИО) и смежные системы по шинам мультиплексного канала обмена (МКО). При этом взаимодействие вычислительной системы с аналоговыми ИО и смежными системами (система обеспечения тепловых режимов, система электроснабжения, система управления пиропатронами и т. д.) осуществляется через блоки силовой автоматики (БСА) [1].

В своём составе БЦВС и БСА имеют вычислительные устройства. В процессе работы данных блоков в их вычислительных устройствах про-

* © Табаков Е. В., Зинина А. И., Киреев Д. С., 2020

исходит выполнение встроенного программного обеспечения (ПО), которое позволяет им решать свои задачи.

От надёжности функционирования СУ зависит нормативность работы всего аппарата. Поэтому при её создании огромное внимание уделяется вопросам нормативности её функционирования.

С целью решения данных вопросов проводятся длительные наземные испытания элементов СУ на испытательных стендах и их отладка. При этом возникает задача как тестирования и отладки электрического взаимодействия между элементами СУ, так и встроенного ПО, выполняемого на вычислительных устройствах. В рамках данной работы уделено внимание второй задаче.

Как уже было отмечено ранее, бортовое ПО выполняется на вычислительных устройствах БЦВС и БСА. По своей структуре вычислительные устройства представляют собой четыре идентичных микроконтроллера, резервирующих друг друга [2].

Для работы с данными микроконтроллерами применяются специальные программные средства, называемые отладчиками. Их функционал позволяет просматривать содержимое памяти микроконтроллеров и регистров их процессоров и изменять его. Также отладчики позволяют запускать процессор, останавливать его и просматривать текущую команду, выполняемую процессором. Таким образом, данные программные средства позволяют просматривать ход выполнения программы и изменять его.

Следует отметить, что используемые в настоящий момент на предприятии отладчики имеют ряд недостатков, связанных со спецификой бортового оборудования КА. Одним из таких недостатков является невозможность вести отладку в реальном времени, т. е. без остановки процессора. Также неудобство создавало то, что из одного экземпляра программы можно вести отладку только одного микроконтроллера, что не позволяло одновременно работать со всеми микроконтроллерами одного вычислителя. Вследствие этого возникла задача разработки нового программного средства отладки.

С целью решения поставленной задачи выбран алгоритм взаимодействия отладчика с оборудованием, который заключается в следующем. Клиентская часть отладчика реализована в программе *QT* на языке программирования *C++*. В процессе работы клиентская часть, установленная на рабочем компьютере программиста, отправляет по протоколу *TCP/IP* через локальную сеть предприятия команды, соответствующие стандарту *OpenOCD*. Выбор *OpenOCD* обусловлен тем, что он предоставляет обширные средства для работы с микроконтроллерами. Данные команды принимаются компьютером, установленным на испытательных стенах, который непосредственно соединён с аппаратурой. При получении команд он посредством программного пакета *OpenOCD* обеспечивает взаимодействие с подключёнными микроконтроллерами.

Также в ходе разработки реализовано взаимодействие клиентской части отладчика с базами данных переменных. Дело в том, что встроенное

ПО СУ формирует значения ряда признаков, которые характеризуют состояние вычислителя. В базах данных находятся названия этих переменных и адреса в памяти, в которых записаны их значения. Таким образом, путём использования баз данных реализована возможность обращения к данным переменным по имени.

Созданное программное средство имеет несколько режимов работы. Кратко рассмотрим некоторые из них.

Режим «*Memory content (Table)*» позволяет просматривать и изменять значения, записанные в памяти микроконтроллера (рис. 2). Средства данного режима позволяют выбирать начальный адрес считывания, выбирать размер читаемого слова данных и систему счисления, в которой отображаются записанные значения.

Режим «*Processors*» предоставляет средства для просмотра хода выполнения программы и его изменения (рис. 3). Он позволяет отслеживать текущую выполняемую команду, изменять её, запускать и останавливать работу процессора. Также с помощью средств *OpenOCD* реализована возможность представления выполняемой команды на дисассемблере.

The screenshot shows a table titled "Memory content(Table)" with columns for Address, Value, and Type. The table lists memory starting at address 0x00000010 up to 0x00000035. The values are mostly zeros, except for some non-zero entries like 0x000002eb at address 0x00000010 and 0x000002f9 at address 0x00000012.

Рис. 2. Окно программы
в режиме «*Memory content (Table)*»

The screenshot shows the "Processors Window" with assembly code. The code includes instructions like LDRB, LDRB.W, LDRB, LDRB r1,[r3,#0x1], LDRB r1,[r3,#0x2], LDRB r2,[r3,#0x3], LDR r8,0x00001338, STRH r4,[r1, #0x10], AND. W r6,r1,14,r6, LSL5 r0,r0, #0x10, ANDS r0,r5, ORRS r0,r6, ANDS r2,r1. A context menu is open over the instruction at address 0x0000112c, listing options like AddBreakPoint, RemoveBreakPoint, RemoveAllBreakPoint, BP := 0x00001130, 0x0000112c, MovePC.

Рис. 3. Окно программы
в режиме «*Processors*»

В ходе работы реализовано программное средство для проведения отладки встроенного ПО, которое в полной мере учитывает особенности, связанные со строением оборудования СУ.

Список литературы

- Проектирование и испытания бортовых систем управления: учеб. пособие / В. П. Андреев, Н. В. Волович, В. М. Глебов и др.; под ред. А. С. Сырова. М.: МАИ-ПРИНТ, 2011. С. 344.
- Синицын С. В. Системное бортовое программное обеспечение. Операционная среда разработки: учебник. Ч. 1. / С. В. Синицын, Д. В. Орлов. М.: МОКБ «МАРС», 2018. 148 с.

ИДЕНТИФИКАЦИЯ И УПРАВЛЕНИЕ МНОГОМЕРНЫМИ СИСТЕМАМИ С ЗАПАЗДЫВАНИЕМ С ОСОБЕННОСТЯМИ

А. В. Терёшина, Д. И. Ярещенко*

*Научный руководитель – А. В. Медведев,
доктор технических наук, профессор
Сибирский федеральный университет*

В настоящее время актуальным является построение моделей и алгоритмов управления для многомерных процессов дискретно-непрерывного характера. Такие процессы возникают в разных отраслях промышленности: например, в металлургии (плавка стали), в строиндустрии (производство цемента), в нефтепереработке (повышение хладотекучести дизельного топлива), а также в социальных системах, включая образовательный процесс (процесс приобретения знаний студентами). В подобных системах измерение некоторых выходных переменных происходит через большие промежутки времени, которое превосходит постоянную времени этого объекта. Поэтому исследуемые процессы рассматриваются как безынерционные с запаздыванием. Если компоненты выходных переменных каким-то образом являются зависимыми, а зависимость эта неизвестна, то подобные процессы были названы T -процессами, а их модели – T -моделями [1].

Основной особенностью многомерных процессов со стохастической зависимостью выходных переменных является то, что процесс описывается в виде системы неявных стохастических уравнений:

$$F_j(u(t), \mu(t), x(t + \tau), \xi(t)) = 0, \quad j = \overline{1, n}, \quad (1)$$

где функции $F_j()$ неизвестны, а τ – запаздывание по различным каналам многомерной системы, которое известно. В дальнейшем τ можно исключить, сделав сдвиг на величину τ в соответствующей матрице наблюдений. Рассматриваемая многомерная система, реализующая T -процесс, может быть представлена на рис. 1.

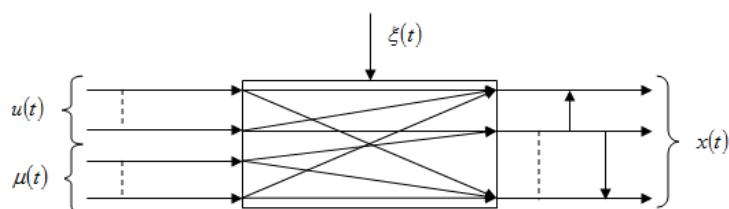


Рис. 1. Многомерный объект

На рис. 1 приняты следующие обозначения:

- $u = (u_1, \dots, u_m) \in \Omega(u) \subset R^m$ – m -мерный вектор входных управляемых переменных, где $\Omega(u)$ – некоторая подобласть, определяемая технологическим регламентом;
- $x = (x_1, \dots, x_n) \in \Omega(x) \subset R^n$ – n -мерный вектор выходных переменных;
- $\mu = (\mu_1, \dots, \mu_p) \in \Omega(\mu) \subset R^p$ – p -мерный вектор входных неуправляемых переменных;
- $\xi(t)$ – случайные помехи, действующие на процесс;
- вертикальные стрелки – зависимость выходных переменных [2].

Для каждого канала многомерной системы могут быть использованы некоторые компоненты векторов входных и выходных переменных \bar{u} , $\bar{\mu}$ и \bar{x} . Это приводит к необходимости использования составных векторов, которые включают эти компоненты, например, $u^{<3>} = (u_1, u_4, \mu_1)$, где $u^{<3>}$ – составной вектор. Составные векторы определяются исследователем из априорной информации.

T -модель рассматривается в виде следующей системы:

$$\hat{F}_j(u^{<j>}, x^{<j>}) = 0, \quad j = \overline{1, m}, \quad (2)$$

где функции $F_j()$ продолжают оставаться неизвестными.

Управление многомерными T -объектами рассматривается в условиях непараметрической неопределённости, т. е. в условиях, когда модель процесса с точностью до вектора параметров отсутствует полностью. При таких условиях известные приёмы не применимы [3] и следует использовать другие подходы для решения задачи. Рассмотрим схему непараметрической системы управления объектом.

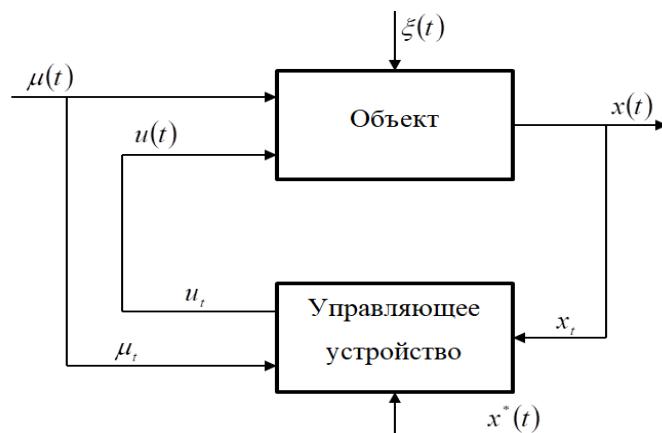


Рис. 2. Схема управления объектом

На рис. 2 $x^* = (x_1, \dots, x_n^*) \in \Omega(x^*) \subset R^n$ – задающее воздействие. Входная компонента $\mu(t)$ является неуправляемой, но контролируемой переменной. Например, это могут быть какие-нибудь технологические параметры, для измерения которых требуется физико-химическая технология.

Такая переменная существенно влияет на объект и на выходные переменные.

Непараметрический алгоритм управления T -объектом принимается в следующем виде. Входную переменную $u_1(t)$ берём произвольно из области $\Omega_1(u)$ и обозначаем её как $u_1^*(t)$. А для расчёта остальных входных переменных применяем следующий алгоритм:

$$u_k^{*s} = \frac{\sum_{i=1}^s u_k^i \prod_{k=1}^{k-1} \Phi\left(\frac{u_k^* - u_k^i}{c_{u_k}}\right) \prod_{j=1}^{<n>} \Phi\left(\frac{x_j^* - x_j^i}{c_{x_j}}\right) \prod_{v=1}^{<p>} \Phi\left(\frac{\mu_v^* - \mu_v^i}{c_{\mu_v}}\right)}{\sum_{i=1}^s \prod_{k=1}^{k-1} \Phi\left(\frac{u_k^* - u_k^i}{c_{u_k}}\right) \prod_{j=1}^{<n>} \Phi\left(\frac{x_j^* - x_j^i}{c_{x_j}}\right) \prod_{v=1}^{<p>} \Phi\left(\frac{\mu_v^* - \mu_v^i}{c_{\mu_v}}\right)}, \quad k = \overline{1, m}, \quad (3)$$

где $< n >$ – размерность составного вектора x_j , $< n > \leq n$; $< p >$ – размерность составного вектора μ_v , $< p > \leq p$; x_j^* – задающие воздействия, определяемые для каждой компоненты вектора выходных переменных из области $\Omega_j(x)$; μ_v^* – произвольные значения из области $\Omega_v(\mu)$. В алгоритме (3) для каждой последующей компоненты вектора управляющих воздействий учитываются предыдущие найденные значения управляющих воздействий. Настраиваемыми параметрами будут параметры размытости c_u , c_x , c_μ , которые находятся в соответствии со следующими формулами: $c_\mu = \alpha |\mu_v^* - \mu_v^i| + \eta$; $c_x = \beta |x_j^* - x_j^i| + \eta$; $c_u = \gamma |u_k^* - u_k^i| + \eta$, где α , β и γ – некоторые параметры больше 1, а $0 < \eta < 1$. Следует заметить, что выбор c_u , c_x , c_μ должен осуществляться на каждом такте управления. При этом если сначала определён c_u , то определение c_x , c_μ осуществляется с учётом этого факта. Однако может быть и наоборот: сначала, например, определяется c_x , а потом c_u , c_μ и т. д.

Приведём результат вычислительного эксперимента, используя алгоритм управления (3), для первой компоненты вектора выходных переменных. Рассматривался объект с четырьмя входными переменными и тремя выходными.

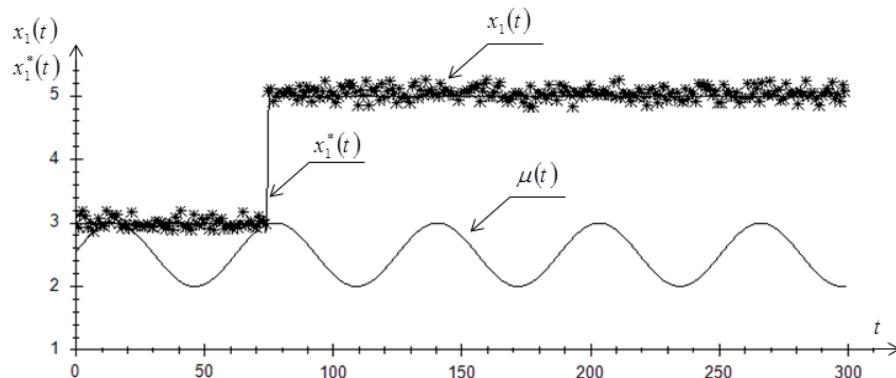


Рис. 3. Управление при задающем воздействии $x_1^*(t)$ в виде ступенчатой функции

Проведённый вычислительный эксперимент по управлению многомерной системой показал достаточно удовлетворительные результаты предлагаемого алгоритма управления (3). Следует отметить, что непараметрический алгоритм управления (3) может быть использован для управления многомерным объектом с запаздыванием в условиях непараметрической неопределённости.

Список литературы

1. Медведев А. В. Основы теории непараметрических систем: монография / А. В. Медведев. Красноярск: СибГАУ, 2018. 727 с.
2. Медведев А. В. Непараметрическое моделирование Т-процессов в условиях неполной информации / А. В. Медведев, Д. И. Ярешенко // Информационные технологии. 2019. № 10 (25). С. 579–584.
3. Цыпкин Я. З. Основы информационной теории идентификации / Я. З. Цыпкин. М.: Наука, 1984. 320 с.

РАЗРАБОТКА ПРОЕКТА РОБОТА-МАНИПУЛЯТОРА

Е. С. Ткачёв*

*Научный руководитель – Г. Б. Масальский,
кандидат технических наук, доцент
Сибирский федеральный университет
Политехнический институт
Кафедра робототехники и технической кибернетики*

В последнее время неотъемлемой частью современных производств стала промышленная автоматизация. Промышленная автоматизация – комплекс устройств, применяемых для автоматизации технологических процессов.

Промышленный робот-манипулятор является частью системы промышленной автоматизации. Он позволяет производить обслуживание технологического оборудования, выполнять операции механообработки, производить дуговую и точечную сварку, выполнять контроль качества и другие трудоёмкие операции, которые ранее выполнял человек.

Поэтому, чтобы не отставать от развивающихся технологий, был разработан проект промышленного робота-манипулятора широкого спектра применения. Актуальность данного проекта заключается в необходимости оснащать производства и учебные заведения промышленными роботами-манипуляторами отечественной разработки.

Робот-манипулятор (кинематическая схема изображена на рис. 1) предназначен для образовательных и производственных целей и обладает следующими характеристиками (табл. 1):

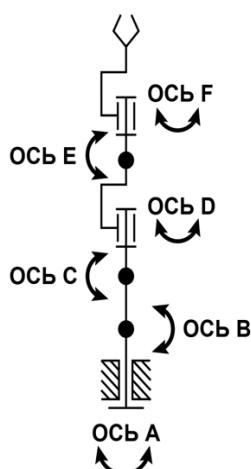


Рис. 1. Кинематическая схема робота-манипулятора

Таблица 1

Основные характеристики робота-манипулятора

Номинальная грузоподъёмность	7 кг
Максимальный радиус действия	1070 мм
Количество управляемых осей	6
Стабильность повторяемости позиционирования	не более $\pm 0,1$ мм
Вес	не менее 120 кг
Температура окружающей среды	от +5 до +45 °C

Исходя из основных характеристик робота-манипулятора (табл. 1) для выбора комплектующих были найдены моменты сил для каждой поворотной оси [1] робота-манипулятора (рис. 2) с учётом массы всех элементов. Результат вычислений представлен в табл. 2.

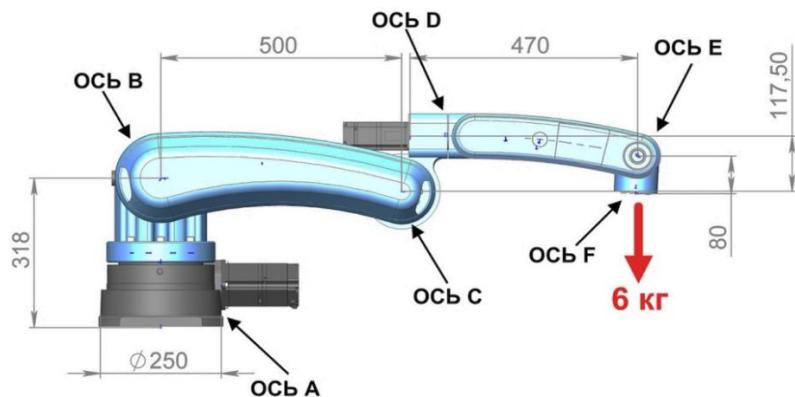


Рис. 2 Схема робота-манипулятора

Таблица 2

Характеристика сервоприводов робота-манипулятора

Наимено-вание оси	Характеристика привода	Крутящий момент мотора (Н×м)	Тип редуктора	Момент силы, действующий на ось (нагрузка 7 кг) (Н×м)
A	Сервопривод с обратной связью	4	Циклоидальный редуктор	210
B		3	Волновой редуктор	390
C		3		150
D		2		15
E		1,5		7
F		4,5		15

После расчёта и выбора комплектующих была спроектирована модель робота-манипулятора. За основу был взят ранее разработанный прототип робота-манипулятора [2] и зарубежные аналоги [3]. На рис. 3 представлена готовая модель робота-манипулятора.

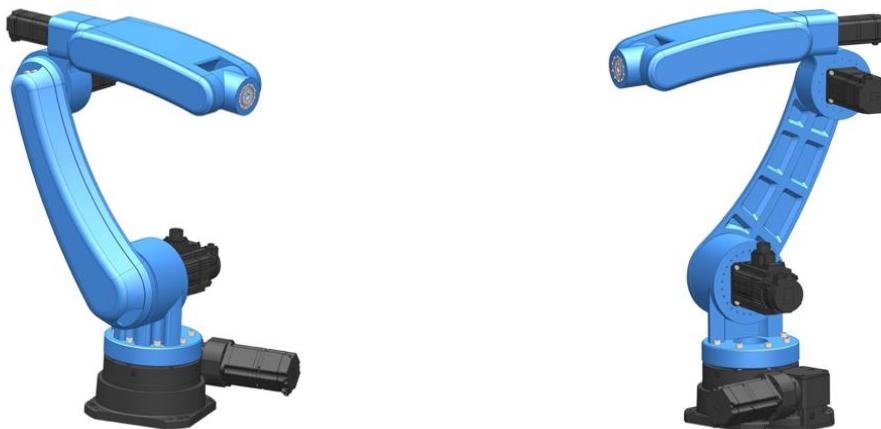


Рис. 3 Общая сборка робота-манипулятора

Для управления роботом-манипулятором было разработано программное обеспечение [4], которое имеет следующие возможности:

- 1) решение прямой задачи кинематики;
- 2) решение обратной задачи кинематики;
- 3) планирование траектории;
- 4) машинное зрение;
- 5) обмен данными между ПК и контроллером робота-манипулятора.

На данный момент разработки проекта робота-манипулятора выполнены следующие виды работ:

- разработана модель робота-манипулятора в *CAD*-системе;
- выбраны комплектующие робота-манипулятора;
- разработано программное обеспечение по управлению роботом-манипулятором.

Следующим этапом будет процесс закупки комплектующих и изготовление прототипа робота-манипулятора.

Список литературы

1. Юревич Е. И. Основы робототехники: учеб. пособие / Е. И. Юревич. 2-е изд. СПб.: БХВ, 2005. 416 с.
2. Ткачёв Е. С. Проектирование робота-манипулятора / Е. С. Ткачёв. Красноярск: СФУ, 2018. URL: <http://elib.sfu-kras.ru/>.
3. DirectIndustry: The online industrial exhibition. URL: <https://www.directindustry.com>.
4. Corke P. Robotics, Vision and Control. Fundamental Algorithms in MATLAB / P. Corke. Springer, 2011. 570 p.

СЕГМЕНТАЦИЯ И КЛАССИФИКАЦИЯ ЛОКАЛЬНЫХ ОБЪЁМНЫХ ОБРАЗОВАНИЙ НА МЕДИЦИНСКИХ ИЗОБРАЖЕНИЯХ

М. С. Хабибуллоев, Л. А. Карнаухова*

*Научный руководитель – К. В. Симонов,
доктор технических наук, профессор
Сибирский федеральный университет*

В настоящее время заболеваемость органов дыхания занимает лидирующее положение, что ведёт к востребованности радиологических исследований. Рентгенограмма грудной клетки является наиболее часто используемым методом визуализации для диагностики различных заболеваний лёгких благодаря её низкому уровню радиации, отсутствию побочных эффектов и экономической целесообразности [1].

Целью исследований является разработка вычислительной методики и алгоритмического обеспечения для обработки и анализа рентгенографии грудной клетки (*CXR*). Решается задача повышения точности обнаружения границы лёгких, что приводит к более точной классификации объектов интереса на *CXR*-изображениях.

Предлагается вычислительная методика, которая включает набор вычислительных процедур для обработки изображений и для дальнейшего анализа. Наличие патологий определяется в рамках предложенной методики в три этапа. В большинстве случаев изображения *CXR* имеют заметный шум и разные уровни контрастности из-за технических характеристик устройства.

Подавление шума с помощью медианного фильтра и усиление контраста проводятся на первом этапе [2]. На втором этапе проводится определение границ лёгких, выполняются обработка по методу Оцу и формирование выпуклой оболочки точек внешнего контура. Третий этап посвящён классификации объектов интереса на изображении (выявление патологии, воспаления и др.).

Выделение признаков основано на методе *GLCM*, а классификация выделенных объектов на *CXR*-изображениях производится классификатором вероятностной нейронной сети *PNN*. Метод *PNN* – это нейронная сеть с прямой связью, основанная на байесовских правилах и анализе дискриминанта ядра Фишера. Алгоритмическое обеспечение реализовано с использованием языка программирования *Matlab* (*Matlab 2017Ra*, программные средства).

* © Хабибуллоев М. С., Карнаухова Л. А., 2020

Основной задачей предварительной обработки является улучшение качества объектов интереса на CXR-изображении. Этот вычислительный этап помогает улучшить определённые параметры CXR-изображений (например, соотношение сигнал/шум), а также улучшить визуальный вид за счёт удаления несущественных шумов и нежелательных частей фона [2].

Сегментация изображений используется для выделения объектов интереса и определения их границ на изображениях. Результатом сегментации изображения является набор областей, которые в совокупности охватывают всё изображение, или набор контуров, извлечённых из изображения. Каждый из пикселей в области имеет какое-либо характеристическое или вычисленное свойство, такое как цвет, яркость или текстура. Метод пороговой сегментации (Оцу) разделяет CXR-изображения на двоичное разбиение яркостей изображения [3].

Классификатор используется для обнаружения аномальных изображений CXR (изображений с патологией). Чтобы провести классификацию изображений, используем вероятностную нейронную сеть. *PNN* относительно нечувствительны к выбросам и генерируют точно предсказанные целевые оценки вероятности. Нейросеть *PNN* состоит из 4 уровней: входного уровня, уровня шаблона, уровня суммирования и уровня принятия решения [4].

Экспериментальное исследование основано на двух наборах данных CXR: наборе округа Монтгомери (*MC*) и наборе больницы Шэньчжэня (*SH*) [5]. Набор *MC* содержит 138 рентгеновских снимков грудной клетки из программы скрининга на туберкулёз округа Монтгомери, а также 80 нормальных и 58 случаев с проявлениями туберкулёза. Набор *SH* содержит 662 рентгенограммы грудной клетки (326 нормальных случаев и 336 случаев с проявлениями туберкулёза). Надёжность и точность определения границ лёгких оценивалась по следующим показателям: показатель качества (*FOM*), коэффициент подобия Жакара (Ω), чувствительность к коэффициенту подобия (*DSC*), среднее расстояние до контура (*ACD*) и точность. Для проверки точности классификатора *PNN* было проведено обучение на совместном наборе изображений CXR из наборов данных *MC* и *SH*. Учебный набор содержит 40 изображений CXR с патологиями и 40 изображений CXR без патологий. Пример расчёта приведен на рис. 1.

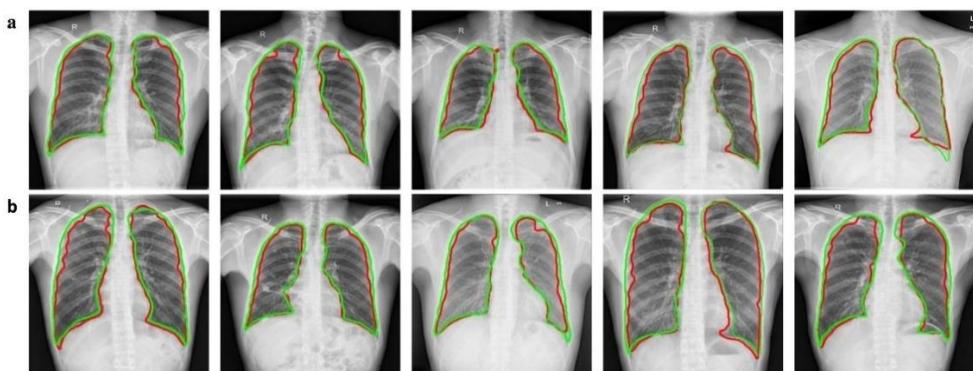


Рис. 1. Пример расчёта: а) обнаружение области лёгких на изображениях CXR без патологии; б) обнаружение области лёгких на изображениях CXR с патологией

Предложена вычислительная методика, которая позволяет выполнять процедуры обнаружения границ лёгких и классификации объёмных локальных объектов интереса на изображениях *CXR*. Методика состоит из трёх этапов: улучшение изображения *CXR*; обнаружение области лёгких; классификация *CXR*-изображения (с патологией или без патологии) с использованием классификатора *PNN*. Средняя точность классификации составляет от 94,98 до 95,77 % и зависит от набора данных.

Список литературы

1. Ahmad W. Classification of Infection and Fluid Regions in Chest X-Ray Images / W. Ahmad, W. S. H. Munirah, W. M. Diyana, W. Zaki, M. Faizal, A. Fauzi, W. H. Tan // IEEE International Conference on Digital Image Computing: Techniques and Applications (DICTA). 2016. P. 1–5.
2. Cadena L. Enhancement of Medical Image using Spatial Optimized Filters and OpenMP Technology / L. Cadena, A. Zotin, F. Cadena // Lecture Notes in Engineering and Computer Science: Proceedings of The International Multi-Conference of Engineers and Computer Scientists. 2018. P. 324–329.
3. Saad M. N. Image segmentation for lung region in chest X-ray images using edge detection and morphology / M. N. Saad, Z. Muda, N. S. Ashaari, H. A. Hamid // IEEE International Conference on Control System, Computing and Engineering (ICCSCE). 2014. P. 46–51.
4. Nandhagopal N. Probabilistic Neural Network Based Brain Tumor Detection and Classification System / N. Nandhagopal, K. R. Gandhi, R. Sivasubramanian // Research Journal of Applied Sciences, Engineering and Technology. 2015. Vol. 10 (12). P. 1347–1357.
5. Jaeger S. Two public chest X-ray datasets for computer-aided screening of pulmonary diseases / S. Jaeger, S. Candemir, S. Antani, Y.-X. J. Wang, P.-X. Lu, G. Thoma // Quantitative imaging in medicine and surgery. 2014. Vol. 4 (6). P. 475–486.

**ПРОГРАММНЫЙ КОМПЛЕКС
ЧИСЛЕННОГО ИССЛЕДОВАНИЯ ЖЁСТКИХ ЗАДАЧ**

Д. В. Хоров, В. Д. Перехрест*

*Научный руководитель – М. В. Рыбков,
старший преподаватель кафедры МОДУС
Сибирский федеральный университет*

Решение жёстких задач, возникающих при моделировании процессов в электрических цепях, химических реакциях и других областях, является актуальной задачей [1]. При этом при разработке методов часто стоит задача их отладки, тестирования на большом объёме задач и разными входными значениями. Здесь представлена программа по разработке и тестированию алгоритмов расчёта жёстких задач систем обыкновенных дифференциальных уравнений.

В выборе инструмента разработки отталкивались от следующих параметров: гибкость среды разработки, стоимость, кроссплатформенность, возможность быстро делать UI-шаблоны программы.

Третий параметр (кроссплатформенность) значительно уменьшил выборку доступных сред разработки, т. к. не все инструменты позволяют без затруднений запускать программы на разных платформах (семейство операционных систем *Linux* или *Windows*).

В итоге выбор остановился на *IDE QT* и языке программирования *C++*. Среда разработки обладает следующими преимуществами:

- кроссплатформенность «из коробки»: достаточно просто перекомпилировать и собрать программу на другой ОС, и ваш исходный код запустится на другой платформе;
- возможность использовать разные версии компиляторов;
- удобные средства отладки приложений;
- удобный редактор форм интерфейса;
- условная бесплатность (для разработки некоммерческих продуктов).

Перед началом разработки необходимо было развернуть рабочее окружение – это все необходимые инструменты, собранные в одной среде и настроенные для конкретной задачи.

Одна из главных задач – подключение библиотеки *GMP* и настройка среды разработки для работы с этой библиотекой.

Реализована программа построения адаптивных алгоритмов интегрирования. Программа позволяет пользователю контролировать параметры алгоритмов. Это значительно облегчает подход к расчётам жёстких си-

* © Хоров Д. В., Перехрест В. Д., 2020

стем, т. к. пользователь может изменять параметры перед запуском расчётов и в реальном времени анализировать полученные результаты.

Отдельное внимание уделено контролю устойчивости. Недостаточная точность встроенных типов данных накладывает ограничения на величину шага интегрирования, что приводит к зацикливанию алгоритмов.

Решение данной проблемы заключается в контроле устойчивости. С этой целью были использованы методы первого порядка, но с расширенными областями устойчивости и высоким числом стадий метода.

Для быстрой проверки графика многочлена устойчивости [2] на данные требования был написан отдельный модуль, который по вычисленным коэффициентам многочлена строит область устойчивости на комплексной плоскости.

Таким образом, разработана программа расчёта жёстких задач систем ОДУ алгоритмами на основе явных методов [3] и дополнительный модуль отрисовки областей устойчивости по вычисленным коэффициентам многочлена устойчивости. Данный программный комплекс обладает элементами мониторинга статистической информации в реальном времени и гибкими инструментами настройки вычислений. Кроме того, проведено тестирование программы, которое показало, что построенные алгоритмы значительно повышают эффективность расчётов – как по числу возвратов, так и по числу вычислений правой части.

Список литературы

1. Новиков Е. А. Явные методы для жёстких систем / Е. А. Новиков. Новосибирск: Наука, 1997. 197 с.
2. Новиков Е. А. Численный алгоритм построения многочленов устойчивости методов первого порядка / Е. А. Новиков, М. В. Рыбков // Вестник БГУ. 2014. № 9–2. С. 80–85.
3. Rybkov M. V. First-Order Methods With Extended Stability Regions for Solving Electric Circuit Problems / M. V. Rybkov, L. V. Knaub, D. V. Khorov // J. Sib. Fed. Univ. Math. Phys. 2020. № 13 (2). P. 242–252. DOI: 10.17516/1997–1397–2020–13–2–242–252.

РАЗРАБОТКА АЛГОРИТМА ОЦЕНКИ
УГЛА АЭРОДИНАМИЧЕСКОГО СКОЛЬЖЕНИЯ
БЕСПИЛОТНОГО ЛЕТАТЕЛЬНОГО АППАРАТА

А. А. Шабашов*

*Научный руководитель – А. Ю. Мазуров,
кандидат физико-математических наук, доцент
Арзамасский политехнический институт (филиал)
ФГБОУ ВО «НГТУ им. Р. Е. Алексеева», г. Арзамас*

Стабилизация короткопериодического движения беспилотного летательного аппарата (БПЛА), имеющего статическую малую устойчивость или неустойчивость и низкую эффективность управления в боковом канале, является достаточно трудной задачей. В некоторых случаях для её решения используются перегрузочные контуры стабилизации с введением дополнительных сигналов. Одним из таких сигналов является угол аэродинамического скольжения $\beta_{ск}$.

Вычислить угол аэродинамического скольжения можно достаточно легко, используя информацию о векторе воздушной скорости от системы воздушных сигналов (СВС). Бывают случаи, когда в состав бортовой системы СВС не входит. Бесплатформенная инерциальная навигационная система (БИНС) при этом даёт информацию о векторе земной скорости. При отсутствии ветра эти скорости будут совпадать, но также следует учитывать ошибки измерения инерциальной системой угла курса, который необходим для перевода вектора скорости из земных осей в связанные.

Универсальным способом может стать оценка угла аэродинамического скольжения с использованием наблюдающего устройства на основе оптимального фильтра Калмана (ФК) и достаточно точной априорной информации об аэродинамических характеристиках (АДХ) БПЛА и доступных измерениях БИНС.

Линеаризованные уравнения бокового возмущённого движения имеют вид:

$$\begin{aligned}\dot{\beta}_{ск} &= Z^\beta \beta_{ск} + \omega_x \sin \alpha + \omega_y \cos \alpha + \frac{g}{V} \gamma \cos \vartheta + \Delta Z, \\ \dot{\omega}_x &= M_x^\beta \beta_{ск} + M_x^{\omega_x} \omega_x + M_x^{\omega_y} \omega_y + \Delta M_x, \\ \dot{\omega}_y &= M_y^\beta \beta_{ск} + M_y^{\omega_x} \omega_x + M_y^{\omega_y} \omega_y + \Delta M_y, \\ \dot{\gamma} &= \omega_x - \omega_y \tan \vartheta,\end{aligned}\tag{1}$$

где $\Delta Z = Z^{\delta_h} \delta_h + Z^{\delta_e} \delta_e$ – составляющая боковой силы, вызванная рулями направления и элерона; $\Delta M_x = M_x^{\delta_h} \delta_h + M_x^{\delta_e} \delta_e$ – составляющая момента крена, вызванная рулями направления и элерона; $\Delta M_y = M_y^{\delta_h} \delta_h + M_y^{\delta_e} \delta_e$ – составляющая момента рыскания, вызванная рулями направления и элерона; ω_x, ω_y – угловые скорости крена и рыскания соответственно; $\alpha, \gamma, \vartheta$ – углы атаки, крена и тангенса соответственно; g – ускорение свободного падения; V – модуль земной скорости; Z^β – частная производная боковой силы по углу скольжения; $M_x^\beta, M_x^{\omega_x}, M_x^{\omega_y}, M_y^\beta, M_y^{\omega_x}, M_y^{\omega_y}$ – частные производные моментов крена и рыскания по соответствующим величинам.

В уравнениях состояния система (1) запишется в виде:

$$\begin{aligned}\dot{x} &= Ax + Bu, \\ y &= Cx + v,\end{aligned}\tag{2}$$

где $x = \begin{bmatrix} \beta_{\text{ск}} \\ \omega_x \\ \omega_y \\ \gamma \end{bmatrix}$ – вектор состояния; $A = \begin{bmatrix} Z^\beta & \sin \alpha & \cos \alpha & \frac{g}{V} \cos \vartheta \\ M_x^\beta & M_x^{\omega_x} & M_x^{\omega_y} & 0 \\ M_y^\beta & M_y^{\omega_x} & M_y^{\omega_y} & 0 \\ 0 & 1 & -\tan \vartheta & 0 \end{bmatrix}$ – матрица системы; $B = \begin{bmatrix} Z^{\delta_h} & Z^{\delta_e} \\ M_x^{\delta_h} & M_x^{\delta_e} \\ M_y^{\delta_h} & M_y^{\delta_e} \\ 0 & 0 \end{bmatrix}$ – матрица управления; $C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ – матрица измерений; $u = \begin{bmatrix} \delta_h \\ \delta_e \end{bmatrix}$ – вектор управляющих сигналов; v – вектор случайного шума измерений с распределением $N(0, \sigma^2)$.

Запишем систему (2) в дискретном виде:

$$\begin{aligned}x_{t+1} &= F_t x_t + G_t u_t + w_t, \\ y_t &= C x_t + v_t,\end{aligned}\tag{3}$$

где $F_t = I + A_t \times \Delta t$; $G_t = B_t \times \Delta t$; I – единичная матрица; Δt – шаг дискретизации; w_t – вектор случайного шума системы с распределением $N(0, \sigma^2)$.

Нижние индексы $t + 1, t, t - 1$ в выражениях означают следующий, текущий и предыдущий шаги соответственно.

Перейдём к построению дискретного оптимального ФК [1]. Начальными условиями являются:

- \hat{x}_0 – начальная оценка состояния;
- P_0 – ковариационная матрица ошибки оценивания;

- $Q = \begin{bmatrix} \sigma_{\beta_{\text{ск}}}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ – ковариационная матрица w_t ;
- $R = \begin{bmatrix} \sigma_{\omega_x}^2 & 0 & 0 \\ 0 & \sigma_{\omega_y}^2 & 0 \\ 0 & 0 & \sigma_{\gamma}^2 \end{bmatrix}$ – ковариационная матрица v_t ;
- $\sigma_{\beta_{\text{ск}}}$ – среднеквадратическое отклонение угла аэродинамического скольжения;
- $\sigma_{\omega_x}, \sigma_{\omega_y}, \sigma_{\gamma}$ – погрешности измерения координат состояния датчиками, имеющимися на борту БПЛА.

Оценка угла аэродинамического скольжения может быть теперь получена с помощью следующего алгоритма:

1. Этап наблюдения:

По данному наблюдению на шаге t вычисляются:

- $v_t = y_t - C\hat{x}_{t|t-1}$ – ошибка обновления;
- $K_t = P_{t|t-1}C_t^T(C_tP_{t|t-1}C_t^T + R)^{-1}$ – коэффициент усиления ФК;
- $\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t v_t$ – уточнённое состояние;
- $P_{t|t} = (I - K_t C_t)P_{t|t-1}$ – ковариация ошибки состояния.

2. Этап прогноза:

- $\hat{x}_{t+1|t} = F_t \hat{x}_{t|t} + B_t u_t$ – прогноз состояния;
- $P_{t+1|t} = F_t P_{t|t} F_t^T + Q$ – ковариация ошибки прогноза состояния.

Математическое моделирование

Математическое моделирование проводилось в среде *Matlab* с помощью решения методом Эйлера замкнутых линеаризованных уравнений бокового возмущённого движения с учётом воздействия ветра. Шаг дискретизации $\Delta t = 0,02$.

За рассматриваемый режим принимается горизонтальный полёт на заданной высоте с постоянной скоростью и нулевыми углами атаки и тангажа. В данном случае мы не учитываем влияние вертикального канала. Рули направления и элерона являются измеряемыми величинами и на входы наблюдателей передаются из математической модели.

Проверка наблюдателя производилась при воздействии на систему знакопеременного порыва ветра с амплитудой 10 м/с и интенсивностью нарастания/убывания 2 м/с.

Результаты моделирования представлены на рис. 1. Параметры с дополнительным нижним индексом « ϕ » соответствуют вектору состояния наблюдателя, а без индекса – модели.

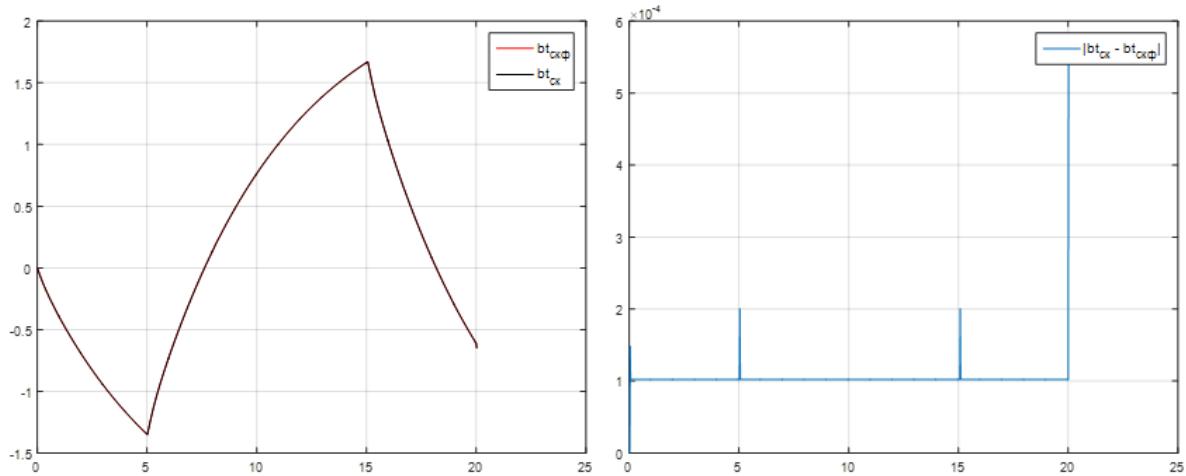


Рис. 1. Углы аэродинамического скольжения (слева) и ошибка оценивания (справа), [град]

Разработанный алгоритм по результатам математического моделирования обеспечил высокую точность оценки угла аэродинамического скольжения в условиях полёта в турбулентной атмосфере.

Список литературы

1. Ледерман У. Справочник по прикладной статистике. Т. 2 / У. Ледерман; пер. с англ. под ред. С. А. Айвазяна и Ю. Н. Тюрина. М.: Финансы и статистика, 1990. 526 с.

СРЕДСТВА ПОВЫШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ JAVASCRIPT-КОДА

А. А. Шатоба*

*Научный руководитель – Д. А. Кузьмин,
кандидат технических наук, доцент
Сибирский федеральный университет*

В статье рассматривается проблема повышения эффективности программного кода на языке программирования *JavaScript*, а именно оптимизация времени выполнения. Как правило, вопросы производительности решаются программными средствами, используемыми в браузерах, однако они не всегда справляются с поставленной задачей, т. к. производительность во многом зависит и от качества написанного разработчиками кода.

Ключевые слова: оптимизация кода, производительность, программный продукт, *JavaScript*, *V8*, *Google*.

Процессом оптимизации кода занимается как разработчик, так и браузер. Среди существующих программных средств по браузерной оптимизации можно выделить движок *V8*, написанный командой *Google* в 2008 г.

V8 – это *JavaScript*-движок с открытым исходным кодом, разработанный компанией *Google*, написанный на *C++* [1]. Движок используется в браузере *Google Chrome*, *Google Chrome OS*, *Chromium*. Движок *V8* компилирует *JavaScript* в машинные инструкции при исполнении программы, реализуя механизм динамической компиляции, так делает большинство современных *JavaScript*-движков: *SpiderMonkey* и *Rhino* (*Mozilla*). Главное отличие *V8* от этих движков состоит в том, что *V8* при исполнении кода не использует байт-код или какой-либо другой промежуточный код [2].

Внутри движка *V8* работает несколько потоков. Главный поток работает с исходным кодом: читает его, компилирует и выполняет. Поток компиляции оптимизирует код во время выполнения главного потока. Поток профилировщика сообщает системе о проблемных участках кода, о тех, где тратится больше времени на выполнение. Есть несколько потоков, которые занимаются сборкой мусора [3, 4].

Для решения поставленной проблемы была создана система, которая способна предоставить более производительный код на основе предоставленного. Система построена на основе браузерного движка от *Google* – *V8*.

Система работает следующим образом: на основе полученного кода она строит абстрактное синтаксическое дерево, после чего каждый узел

* © Шатоба А. А., 2020

проверяется на наличие ресурсоёмких операций (профилирование), и если таковой найден, то данный узел модифицируется путём использования более экономичных конструкций, после чего код подаётся на вход парсеру, а на выходе получается оптимизированный код.

Разработанный программный продукт использует фреймворк *Angular*, поскольку этот инструмент обеспечивает всем необходимым, и язык программирования *TypeScript*. Реализация программного продукта содержит в себе два слоя:

- 1) компонентов;
- 2) сервисов.

UI-компоненты получают от пользователя код, над которым будут проводиться манипуляции, а также выводят результат и отображают статистические данные.

Сервисы содержат логику работы с кодом: построение бинарного дерева, профилирование, замена конструкций, которые требуют больших ресурсов компьютера, построение кода из абстрактного дерева при помощи парсера.

Ниже представлен код, который подаётся на вход программе, и код, который получился после манипуляций над ним, а также временная оценка производительности обоих вариантов кода.

Таблица 1
Пример работы системы, обработка входного кода

Участок исходного кода	Результат работы системы
array.forEach(el => {count += el;});	for (var i = 0; i < array.length; i++) {count += array[i];}

Таблица 2
Временная оценка производительности кода

Количество элементов массива	Время работы исходного кода, ms	Время работы полученного кода, ms
10	0,0309999953992665	0,04699999922886491
100	0,01499999944120646	0,0279999975599348
1000	0,0820000082142651	0,5540000005159527
10 000	1,7219999993685633	0,7359999995678663
100 000	10,405999999167397	7,663999999081716
1 000 000	43,35200000135228	20,775000000093133

Таким образом, разработанная система показала, что на больших количествах данных способна существенно увеличить производительность кода. Кроме того, приведённый пример является самым простым, и, несомненно, разница в производительности будет более ощутима, если в коде будет присутствовать сложная логика и будут обрабатываться большие объёмы данных.

Список литературы

1. V8 под капотом. URL: <https://habr.com/ru/company/jugru/blog/338930/>.
2. How JavaScript works: inside the V8 engine + 5 tips on how to write optimized code. URL: <https://blog.sessionstack.com/how-javascript-works-inside-the-v8-engine-5-tips-on-how-to-write-optimized-code-ac089e62b12e>.
3. Основы движков JavaScript: общие формы и Inline кэширование.
- Ч. 1. URL: <https://habr.com/ru/company/otus/blog/446446/>.
4. Как работает JavaScript. URL: <https://nuancesprog.ru/p/4553/>.

СРАВНЕНИЕ ТЕХНОЛОГИЙ БЕСПРОВОДНОЙ СВЯЗИ, ИСПОЛЬЗУЕМЫХ В IoT-ПРОЕКТАХ

Д. О. Шкуринский, А. А. Соколов*

Научный руководитель – А. В. Хныкин,
кандидат технических наук, доцент
Сибирский федеральный университет

Необходимость в межмашинном взаимодействии появилась одновременно с доступной цифровой техникой и бурным ростом проектов, использующих её. Первые попытки разработать системы межмашинного взаимодействия относятся к 70-м годам прошлого века, но настоящий расцвет эта область увидела с появлением доступной мобильной связи и иных технологий беспроводной передачи данных, что сделало возможным разработку систем позиционирования и иных систем с использованием данного вида связи. Вместе с появлением новых стандартов и технологий, в том числе *IEEE 802.11n* (более известного как *Wi-Fi*), росло количество устройств, подключённых к беспроводным сетям. Одной из наиболее популярных парадигм межмашинного взаимодействия является концепция интернета вещей.

Интернет вещей (англ. *Internet of Things, IoT*) – концепция сети физических предметов, взаимодействующих между собой и с окружающей средой. Концепция сформулирована в 1999 г. как осмысление перспектив широкого применения средств беспроводной связи. Несмотря на постоянно растущую популярность данной концепции, имеет место недостаток сведений прикладного характера, что может быть критично при работе над проектами, использующими Интернет вещей.

Целью нашей работы было провести анализ и сравнение технологий беспроводной связи, наиболее часто используемых в IoT-проектах, и на основе собранных данных выявить те из них, что подходят для небольших проектов. Объектами сравнения служили следующие технологии: *Bluetooth* и *Bluetooth Low Energy (BLE)* [1], *DASH7*, *LoRa-WAN*, *NB-IoT*, *Weightless*, *Wi-Fi*, *ZigBee*, *Z-Wave*, а также стандарты мобильной связи *4G* и *GSM*. Сравнение производилось по следующим критериям: скорость передачи данных, энергопотребление, максимальное количество «ведомых» устройств и используемый диапазон частот. Результаты исследования представлены в табл. 1.

* © Шкуринский Д. О., Соколов А. А., 2020

Таблица 1

Сравнение технологий беспроводной связи

Название	Скорость передачи данных	Энерго-потребление	Количество «ведомых» устройств	Диапазон частот
4G-связь	до 150 Мбит/с	высокое	до 2000 [4]	зависит от страны
<i>Bluetooth</i> и <i>BLE</i>	до 3 Мбит/с	низкое	до 8	2,4–2,4835 ГГц
<i>DASH7</i>	до 200 Кбит/с	низкое	неизвестно	433 / 868 / 915 МГц
<i>GSM</i> -связь	до 454 Кбит/с (с использованием <i>EDGE</i>)	низкое	неизвестно	850 / 900 / 1800 / 1900 МГц
<i>LoRa-WAN</i>	до 50 Кбит/с	низкое	до 1000000	зависит от страны [3]
<i>NB-IoT</i>	до 250 Кбит/с	очень низкое	неизвестно	453–2690 МГц
<i>NRF24</i>	до 2 Мбит/с	низкое	до 6	2,4–2,4835 ГГц
<i>Weightless</i>	до 100 Кбит/с	низкое	неизвестно	169–923 МГц
<i>Wi-Fi</i>	до 11 Гб/с	высокое	до 255	2,4 / 5 ГГц
<i>ZigBee</i>	до 250 Кбит/с	низкое	до 16	2,4–2,48 ГГц
<i>Z-Wave</i>	до 100 Кбит/с	низкое	до 232	зависит от страны

Принимая во внимание то, что для *IoT*-проектов одним из наиболее важных аспектов является автономность устройств, технологии с низким расходом электроэнергии будут предпочтительнее. К сожалению, некоторые из них (в частности *GSM*-связь и *NB-IoT*) требуют лицензии для использования их рабочего радиодиапазона. Также стоит принимать во внимание то, что небольшие проекты, как правило, работают с небольшим количеством устройств.

Технологии *DASH7* и *Weightless*, несмотря на формальное соответствие требованиям, были отсеяны из-за невозможности вынести объективную оценку на основе скучных сведений о них. Таким образом, были отобраны три технологии, подходящие под указанные критерии: *Bluetooth* и *BLE*, *NRF-24* и *ZigBee*.

Список литературы

1. 12 Wireless Options for IoT/M2M: Diversity or Dilemma? URL: electronicdesign.com/technologies/iot/article/21801725/12-wireless-options-for-iotm2m-diversity-or-dilemma (дата обращения: 09.04.2020).
2. Z-Wave vs ZigBee, WiFi, Thread, Bluetooth BLE: выбираем протокол управления умным домом. URL: moy-domovoy.ru/z-wave-vs-zigbee-wifi-thread-bluetooth-ble-vybiraem-protokol-upravleniya-umnym-domom (дата обращения: 09.04.2020).
3. Плотников О. Практический опыт эксплуатации сети LoRa-WAN. Заметки IoT-провайдера / О. Плотников. URL: <https://nag.ru/articles/article/101403/prakticheskiy-opyit-ekspluatatsii-seti-lorawan-zametki-iot-provaydera.html> (дата обращения: 09.04.2020).

4. What is the maximum number of mobile users that can be served by single 3G node / 4G enode at the same time? URL: quora.com/What-is-the-maximum-number-of-mobile-users-that-can-be-served-by-single-3G-node-4G-enode-at-the-same-time (дата обращения: 09.04.2020).

ГИБРИДНЫЙ МЕТОД ОБУЧЕНИЯ НЕЙРОННОЙ СЕТИ С ПОМОЩЬЮ ДИФФЕРЕНЦИАЛЬНОЙ ЭВОЛЮЦИИ И ОБРАТНОГО РАСПРОСТРАНЕНИЯ ОШИБКИ

В. Г. Юршин*

*Научный руководитель – В. В. Становов,
кандидат технических наук, доцент
Сибирский федеральный университет*

Обучение искусственной нейронной сети сводится к задаче оптимизации функции потерь на выходе из этой сети [1]. Рассмотрены два метода оптимизации: обратное распространение ошибки (градиентный метод) и дифференциальная эволюция. Оптимизация с помощью дифференциальной эволюции показывает преимущества над градиентным методом, т. к. лучше ищет глобальный экстремум функции. В свою очередь, градиентный метод лучше ищет локальный экстремум.

В данной работе представлен гибридный метод обучения нейронной сети с помощью дифференциальной эволюции для поиска глобального минимума и с применением обратного распространения ошибки для каждого индивида популяции.

В дифференциальной эволюции в первую очередь инициализируется популяция, состоящая из n числа индивидов (векторов). Каждый индивид в данной работе является массивом весов нейронной сети (начальное значение весов сгенерировано случайным образом по нормальному закону при среднем значении 0 и дисперсии 0,05). Далее идёт основной цикл алгоритма: мутация, скрещивание, селекция.

Используется мутация с разницей между векторами:

$$\vec{V}_{i,G} = \vec{X}_{r_1^i,G} + F^* \left(\vec{X}_{r_2^i,G} - \vec{X}_{r_3^i,G} \right), \quad (1)$$

где $\vec{V}_{i,G}$ – донор-вектор; $\vec{X}_{r_1^i,G}, \vec{X}_{r_2^i,G}, \vec{X}_{r_3^i,G}$ – случайные векторы, не равные друг другу; F – масштабирующий фактор.

При каждой итерации цикла после скрещивания данные в каждом векторе популяции корректируются с помощью алгоритма обратного распространения ошибки.

Через нейронную сеть с весовыми коэффициентами, полученными после скрещивания в прямом распространении, пропускаются нормализованные данные. Далее от последнего слоя к первому рассчитываются погрешности нейронов и корректируются весовые коэффициенты (3). Также для

* © Юршин В. Г., 2020

лучшей работы алгоритма используется метод моментов (2). Момент накапливает значения градиентов и ускоряет сходимость к экстремуму [2].

$$z^{k+1} = \beta z^k + \nabla f(w^k). \quad (2)$$

$$w^{k+1} = w^k + \alpha z^{k+1}. \quad (3)$$

где z – момент; α – скорость обучения; β – импульс; $\nabla f(w^k)$ – градиент; w – вес.

По завершении работы алгоритма обратного распространения ошибки производится селекция. Селекция определяет, улучшилось ли значение функции потерь полученного вектора по сравнению с целевым. Таким образом, если значение функции потерь полученного вектора меньше, чем значение функции целевого индивида, он заменяет этот целевой вектор, в противном случае – отбрасывается.

Также при работе алгоритма дифференциальной эволюции важным является выбор параметров F и Cr . Коэффициент масштаба F определяет максимально возможное расстояние, на которое может расширяться область поиска донор-вектора. Cr показывает, с какой вероятностью наследуется признак от донор-вектора.

Для обратного распространения ошибки также важны параметры α и β . Ниже в табл. 1 приведена средняя точность на разных наборах данных при 10 запусках алгоритма с разными значениями момента и скорости обучения. Cr и F фиксировались на значении 0,5. Количество скрытых слоёв нейронной сети равнялось трём, на первом слое – 5 нейронов, на втором – 6, на третьем – 5.

С помощью обученной нейронной сети классифицировались данные, взятые с сайта KEEL [3], и выбирались оптимальные значения параметров α и β .

Таблица 1

Средняя точность для нейронной сети

$\alpha \backslash \beta$	0,1	0,3	0,5	0,7	0,9	Выборка
0,01	0,817	0,900	0,882	0,858	0,833	<i>Iris</i>
0,04	0,895	0,791	0,847	0,852	0,855	
0,07	0,890	0,882	0,869	0,867	0,867	
0,1	0,865	0,885	0,860	0,787	0,763	
0,01	0,753	0,763	0,751	0,770	0,730	<i>Phoneme</i>
0,04	0,751	0,733	0,725	0,725	0,718	
0,07	0,717	0,718	0,725	0,721	0,729	
0,1	0,723	0,725	0,724	0,722	0,718	
0,01	0,763	0,769	0,768	0,770	0,770	<i>Titanic</i>
0,04	0,770	0,769	0,769	0,770	0,741	
0,07	0,762	0,770	0,771	0,770	0,720	

$\alpha \backslash \beta$	0,1	0,3	0,5	0,7	0,9	Выборка
α	0,755	0,754	0,735	0,742	0,706	
0,1						

Из полученных данных следует, что точность на больших объемах данных возрастает при α , близких к 0,1, и средних значениях момента β .

Для гибридного алгоритма обучения фиксируем параметры α и β на значениях 0,07 и 0,4 соответственно. С полученными параметрами обратного распространения ошибки запустим гибридный алгоритм с разными значениями Cr и F .

Таблица 2

Средняя точность гибридного алгоритма

$F \backslash Cr$	0,2	0,4	0,6	0,8	Выборка
0,2	0,360	0,432	0,535	0,687	<i>Iris</i>
0,4	0,587	0,565	0,658	0,890	
0,6	0,658	0,757	0,817	0,862	
0,8	0,658	0,725	0,732	0,777	
0,2	0,717	0,726	0,710	0,732	<i>Phoneme</i>
0,4	0,719	0,731	0,721	0,749	
0,6	0,714	0,722	0,731	0,726	
0,8	0,740	0,727	0,741	0,736	
0,2	0,763	0,770	0,755	0,770	<i>Titanic</i>
0,4	0,756	0,769	0,769	0,770	
0,6	0,769	0,762	0,763	0,763	
0,8	0,770	0,763	0,755	0,770	

Из табл. 2 следует, что при 30 поколениях и 10 индивидах значение средней точности на различных выборках в среднем больше 70 %. Наилучшие значения получаются при коэффициенте масштаба, равном 0,4, и вероятности скрещивания 0,8.

Список литературы

1. Рутковская Д. Нейронные сети, генетические алгоритмы и нечеткие системы / Д. Рутковская, М. Пилиньский, Л. Рутковский, В. Радченко. М.: Горячая линия – Телеком, 2006. 385 с.
2. Das S. Differential Evolution: A Survey of the State-of-the-Art / S. Das, P. Suganthan // IEEE Transactions On Evolutionary Computation. 2011. Vol. 15. № 1.
3. Alcalá-Fdez. KEEL: A software tool to assess evolutionary algorithms for data mining problems / Alcalá-Fdez, L. Sánchez, S. García, M. J. del Jesus, S. Ventura, J. M. Garrell, J. Otero, C. Romero, J. Bacardit, V. M. Rivas, J. C. Fernández, F. Herrera // Soft Comput. 2009. Vol. 13. № 3. P. 307–318.

**ПОВЫШЕНИЕ КАЧЕСТВА
ИСХОДНЫХ МЕТЕОРОЛОГИЧЕСКИХ ДАННЫХ
КАК СПОСОБ УЛУЧШЕНИЯ ПРОГНОЗИРОВАНИЯ ПОГОДЫ**

К. Г. Ященков*

*Научный руководитель – А. В. Хныкин,
кандидат технических наук, доцент
Сибирский федеральный университет*

Последние десятилетия службы метеорологических наблюдений работают без активного использования информационных технологий. Каждый день метеоролги формируют несколько десятков сообщений, содержащих информацию о текущей метеорологической обстановке в закодированном виде. Передача данных при этом долгое время производилась при помощи телефонной связи. Такой подход сохранялся долгое время, но в какой-то момент перестал удовлетворять требованиям современного мира.

Для работы с данными метеорологических наблюдений были разработаны программы для составления и обмена метеорологическими сообщениями с использованием возможностей интернета. Пропускная способность каналов связи значительно выросла, что позволило также повысить и количество выпускаемых сообщений для более точного прогнозирования метеобстановки.

Со временем стала ярко проявляться проблема большого количества ошибок, допускаемых в прогнозах. Одной из причин появления ошибок является высокая загруженность метеорологов, которые в короткие промежутки времени должны производить выпуск сразу нескольких сообщений разных типов.

На сегодняшний день прогнозы, предоставляемые метеослужбами, часто содержат ошибки или оказываются неточными. В связи с этим появилась необходимость поиска и внедрения новых возможностей, которые не только позволяют оптимизировать работу метеорологов, но и открыть доступ к анализу отсылаемых данных с целью нахождения в них аномалий. Использование алгоритмов для поиска, определения и исправления ошибочных данных позволило бы существенно уменьшить количество недостоверной информации, поступающей от каждой из нескольких десятков метеостанций ежедневно, и, как следствие, повысить качество прогнозной информации. Следовательно, использование компьютерных технологий и высокопроизводительных алгоритмов обработки поступающей информации с метеостанций становится необходимым.

* © Ященков К. Г., 2020

Объектом исследования является обработка и анализ метеосообщений. Предметом исследования является поиск некорректных и аномальных данных в метеосводках. Основной целью является уменьшение трудозатрат при формировании метеорологических сообщений и повышение точности формируемых прогнозов. Достижение цели заключается в создании программного обеспечения, способного анализировать метеорологические сообщения на предмет ошибочных данных и производить анализ существующих данных для выделения аномалий среди всех сообщений.

Применение программного обеспечения для передачи информации в других странах уже широко распространено. Обмен данными производится по общепринятым и стандартизированному каналу *UNIMAS* [1]. Однако средства формирования и передачи сообщений могут быть разнообразными. Отсутствие обработки ошибок в сообщениях других стран подтверждается большим количеством неточной информации, поступающей от иностранных метеостанций.

Формируемая информация метеосводок представляет собой большую ценность также и для обеспечения безопасности полётов всех воздушных судов, в том числе и для пролетающих над территорией Красноярского края. На основе показателей, полученных из данных метеосообщений, сотрудниками центра управления полётами ежедневно формируются карты воздушных коридоров.

Для решения проблем, возникающих в ходе формирования метеосводок, было принято решение о разработке программного обеспечения, способного значительно облегчить и повысить качество работы метеорологов. Результаты проведённого литературного обзора говорят о том, что разрабатываемая программа с применением алгоритма выявления и обработки ошибок является уникальной, не имеющей аналогов в нашей и других странах. Основными пользователями данного продукта будут метеослужбы, а тестовая эксплуатация планируется на базе организации «РосГидромет» [2].

Для разработки программного комплекса был выбран язык программирования *C++* и среда разработки *.NET Core* [3]. В качестве обработчика текста было решено использовать встроенный движок регулярных выражений под названием *GOLD Parsing System* [4]. Это мощное средство написания собственных выражений и их интеграции в существующие программные комплексы.

Для извлечения информации из входных данных, которые представлены введённой пользователем строкой, применяется лексический и синтаксический анализ на основе регулярных выражений. Для каждого из типов сообщений разрабатывается отдельный парсер, который состоит из набора декодеров – один декодер на одно слово сообщения.

Применение данного программного обеспечения также позволит производить сбор информации о наиболее часто встречающихся ошибках в работе с метеоданными. Ещё одной немаловажной особенностью программы может стать возможность анализа уже отправленных сообщений на предмет

аномальных значений метеоданных. Эта функция открывает новые возможности для немедленного реагирования на внезапно возникающие природные катаклизмы и для анализа факторов, способных привести к появлению необычных явлений погоды в различных регионах края.

Появление описанного программного обеспечения для анализа и предсказания воздушной обстановки позволит в значительной степени повысить точность формируемых прогнозов, а также производить контроль качества работы персонала метеостанций.

В заключение хочется отметить, что внедрение методов и алгоритмов анализа метеоданных позволит повысить точность прогнозирования погоды, что в конечном итоге может привести к снижению ущерба от несвоевременного реагирования на предстоящую погоду.

Список литературы

1. Универсальный метеорологический телекоммуникационный комплекс маршрутизации сообщений и файлов. URL: <https://docplayer.ru/26230235-Universalnyy-meteorologicheskiy-telekommunikacionnyy-kompleks-marshrutizacii-soobshcheniy-i-faylov.html> (дата обращения: 11.04.2020).
2. Федеральная служба по гидрометеорологии и мониторингу окружающей среды. URL: <http://www.meteorf.ru/> (дата обращения: 11.04.2020).
3. .NET Core. URL: <https://dotnet.microsoft.com/download/dotnet-core> (дата обращения: 11.04.2020).
4. GOLD Parsing System: A Free, Multi-Programming Language, Parser Generator. URL: <http://goldparse.org/> (дата обращения: 11.04.2020).

Учебное издание

ПРОСПЕКТ СВОБОДНЫЙ – 2020

Материалы Международной конференции молодых ученых

Красноярск, 20 апреля – 18 мая 2020 г.

Корректор *Л. В. Боец*
Компьютерная вёрстка *И. В. Владимировой*

Подписано в печать 31.08.2020. Печать плоская. Формат 60×84/16
Бумага офсетная. Усл. печ. л. 5,4. Тираж 100 экз. Заказ № 11782

Библиотечно-издательский комплекс
Сибирского федерального университета
660041, Красноярск, пр. Свободный, 82а
Тел. (391) 206-26-16; <http://bik.sfu-kras.ru>,
E-mail: publishing_house@sfu-kras.ru

Для заметок

Для заметок