



XI

**МЕЖДУНАРОДНАЯ
НАУЧНО-ПРАКТИЧЕСКАЯ
КОНФЕРЕНЦИЯ
«ИНТЕГРИРОВАННЫЕ МОДЕЛИ
И МЯГКИЕ ВЫЧИСЛЕНИЯ
В ИСКУССТВЕННОМ
ИНТЕЛЛЕКТЕ»**

ИММВ-2022



СБОРНИК НАУЧНЫХ ТРУДОВ

Том 1

**16-19 мая 2022 г.
Коломна**



XI Международная научно-практическая конференция

**«ИНТЕГРИРОВАННЫЕ МОДЕЛИ
И МЯГКИЕ ВЫЧИСЛЕНИЯ
В ИСКУССТВЕННОМ ИНТЕЛЛЕКТЕ**

(16-19 МАЯ 2022 г.)

СБОРНИК НАУЧНЫХ ТРУДОВ

Том 1

*Памяти вдохновителей и организаторов конференции
Валерия Борисовича Тарасова
и Эвклида Герасимовича Зафирова*

КОЛОМНА
2022

УДК 004.8 : 681.3.01
ББК 32.813

Организаторы конференции:

*Российская ассоциация искусственного интеллекта
Российская ассоциация нечетких систем и мягких вычислений
Федеральный исследовательский центр «Информатика и управление» РАН
Администрация Коломенского городского округа
Московский государственный технический университет им. Н.Э. Баумана
Коломенский институт (филиал) Московского политехнического университета
Институт компьютерных технологий и информационной безопасности
Южного федерального университета*

Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов XI Международной научно-практической конференции (ИММВ-2022, Коломна, 16-19 мая 2022 г.). В 2-х томах. Т.1. – М.: РАИИ, 2022. – 352 с.

ISBN 978-5-604-2802-5-6

Международная научно-практическая конференция посвящена вопросам построения интегрированных систем и реализации мягких вычислений в искусственном интеллекте.

В первом томе трудов публикуются пленарные доклады, а также доклады участников конференции, представленные на следующих секциях:

Секция 1 «Нечёткие модели, мягкие вычисления, измерения и оценки»,

Секция 2 «Машинное обучение, нейросетевые технологии»,

Секция 3 «Биоинспирированные подходы, эволюционное моделирование, генетические алгоритмы».

ISBN 978-5-604-2802-5-6

© Авторы, 2022
© РАИИ, 2022
© РАНСМВ, 2022



*Эвклид Герасимович Зафиров и Валерий Борисович Тарасов
на Международной научно-технической конференции,
посвященной 65-летию со дня рождения
Виктора Владимировича Емельянова
«Интеллектуальные системы
и компьютерно-интегрированные производства»
27 января 2014 г., МГТУ им. Н.Э. Баумана*

ПРОГРАММНЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

Сопредседатели

Борисов В.В., д.т.н., проф. (Россия, Смоленск, филиал НИУ МЭИ)

Кобринский Б.А., д.м.н., проф. (Россия, Москва, ФИЦ ИУ РАН)

Язенин А.В., д.ф.-м.н., проф. (Россия, Тверь, ТвГУ)

Заместители председателей

Веселов Г.Е., д.т.н., проф. (Россия, Таганрог, ЮФУ, ИКТИБ)

Ковалёв С.М., д.т.н., проф. (Россия, Ростов-на-Дону, РГУПС)

Члены программного комитета

Аверкин А.Н., к.ф.-м.н., доц. (Россия, Москва, ФИЦ ИУ РАН)

Алиев Р.А., д.т.н., проф. (Азербайджан, Баку, АЗГНА)

Городецкий В.И. д.т.н., проф. (Санкт-Петербург, СПб ФИЦ РАН)

Гладков Л.А., к.т.н., доц. (Россия, Таганрог, ЮФУ, ИКТИБ)

Грибова В.В., д.т.н., проф. (Россия, Владивосток, ИАПУ ДВО РАН)

Дудаков С.М., д.ф.-м.н., доц. (Россия, Тверь, ТвГУ)

Еремеев А.П., д.т.н., проф. (Россия, Москва, НИУ МЭИ)

Захаров В.Н., д.т.н., проф. (Россия, Москва, ФИЦ ИУ РАН)

Карпов В.Э., к.т.н., доц. (Россия, Москва, НИЦ КИ)

Колесников А.В., д.т.н., проф. (Россия, Калининград, БФУ им. И. Канта)

Колоденкова А.Е., д.т.н., доц. (Россия, Самара, СамГТУ)

Кузнецов О.П., д.т.н., проф. (Россия, Москва, ИПУ РАН)

Лебедев О.Б., д.т.н., доц. (Россия, Таганрог, ЮФУ, ИКТИБ)

Мисник А.Е., к.т.н., доц. (Беларусь, Могилев, БРУ)

Морозов А.В., д.т.н., проф. (Россия, Анапа, ВИТ «ЭРА»)

Осадчук А.В., к.т.н. (Россия, Москва, ГУИР МО РФ)

Палюх Б.В., д.т.н., проф. (Россия, Тверь, ТвГТУ)

Плесневич Г.С., к.ф.-м.н., доц. (Россия, Москва, НИУ МЭИ)

Редько В.Г., д.ф.-м.н., проф. (Россия, Москва, ФНЦ НИИСИ РАН)

Рыбина Г.В., д.т.н., проф. (Россия, Москва, НИЯУ МИФИ)

Стефанюк В.Л., д.т.н., проф. (Россия, Москва, ИППИ РАН)

Тельнов Ю.Ф., д.э.н., проф. (Россия, Москва, РЭУ им. Г.В. Плеханова)

Тулупьев А.Л., д.ф.-м.н., проф. (Россия, Санкт-Петербург, СПбГУ)

Федунов Б.Е., д.т.н., проф. (Россия, Москва, ФГУП ГосНИИАС)

Финн В.К., д.т.н., проф. (Россия, Москва, ФИЦ ИУ РАН)

Фоминых И.Б., д.т.н., проф. (Россия, Москва, НИУ МЭИ)

Юсупбеков Н.Р., академик АН РУз (Узбекистан, Ташкент, ТашГТУ)

Ющенко А.С., д.т.н., проф. (Россия, Москва, МГТУ им. Н.Э. Баумана)

Ярушкина Н.Г., д.т.н., проф. (Россия, Ульяновск, УлГТУ)

ОРГАНИЗАЦИОННЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

Сопредседатели

Гречищев А.В. (Россия, Коломна, Глава Коломенского городского округа)
Гаврюшин С.С., д.т.н., проф. (Россия, Москва, МГТУ им. Н.Э. Баумана)

Заместители председателей

Королева М.Н., к.т.н., доц. (Россия, Москва, МГТУ им. Н.Э. Баумана)
Мурзак Н.А., к.э.н., доц. (Россия, Коломна, Коломенский институт
(филиал) Московского политехнического университета)

Члены организационного комитета

Ефремова А.П., к.т.н. (Россия, Москва, АО «НИИАС»)
Луферов В.С., к.т.н. (Россия, Смоленск, филиал НИУ МЭИ)
Солопов Р.В., к.т.н., доц. (Россия, Смоленск, филиал НИУ МЭИ)
Федотова А.В., к.т.н., доц. (Россия, Москва, МГТУ им. Н.Э. Баумана)
Аникин М.А. (Россия, Москва, МГТУ им. Н.Э. Баумана)
Ахтямова М.Л. (Россия, Москва, МГТУ им. Н.Э. Баумана)
Мажуга В.М. (Россия, Коломна, ООО «Инлайт»)
Тычинская А.М. (Россия, Смоленск, филиал НИУ МЭИ)

ИНФОРМАЦИОННАЯ ПОДДЕРЖКА КОНФЕРЕНЦИИ

ООО «Лаборатория информационных технологий» (Россия, Смоленск)

ПРЕДИСЛОВИЕ

Перед Вами труды XI Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2022).

Конференции ИММВ на протяжении более чем двух десятилетий традиционно проводятся в старинном русском городе Коломна, по праву являющимся «колыбелью» отечественных научных сообществ по искусственному интеллекту и мягким вычислениям. С конца 80-х годов в Коломне неоднократно проводились научные семинары по искусственному интеллекту, нечетким множествам и их приложениям. Эти мероприятия организовывались основоположником искусственного интеллекта (ИИ) в нашей стране, д.т.н., профессором **Дмитрием Александровичем Поспеловым** и его учеником, талантливым учёным в области нечетких множеств, популяризатором ИИ, успешным предпринимателем, жителем Коломны, к.ф.-м.н. **Александром Филипповичем Блишуным**. В мае 1989 года по их инициативе в Коломне был проведён Учредительный съезд **Советской ассоциации искусственного интеллекта (САИИ)**, позднее ставшей Российской.

После трагического ухода из жизни А.Ф. Блишуна в 1991 году его друзья и коллеги продолжили проводить научные мероприятия в Коломне, которые постепенно переросли сначала в семинары, а затем и в конференцию «Интегрированные модели и мягкие вычисления в искусственном интеллекте».

Первый научный семинар ИММВ состоялся 17-18 мая 2001 года в Коломне под председательством известного учёного, ведущего специалиста в области интеллектуального имитационного моделирования, д.т.н., профессора, заведующего кафедрой «Компьютерные системы автоматизации производства» (РК-9) МГТУ им. Н.Э. Баумана **Виктора Владимировича Емельянова**. Соорганизатором первого семинара выступил выпускник МГТУ им. Н.Э. Баумана, аспирант В.В. Емельянова, предприниматель **Эвклид Герасимович Зафиров**, будущий руководитель Коломенского завода РТИ.

После безвременного ушедшего в 2004 году В.В. Емельянова главным организатором и вдохновителем конференции ИММВ как со стороны кафедры РК-9 МГТУ им. Н.Э. Баумана, так и со стороны РАИИ стал великодушный учёный, одним из ведущих отечественных специалистов в области ИИ, к.т.н., доцент **Валерий Борисович Тарасов**.

Благодаря В.Б. Тарасову и Э.Г. Зафирову, их неутомимой энергии, Коломенская конференция стала центром притяжения ученых и специалистов в области интеллектуальных систем, нечетких множеств, мягких вычислений, неклассических логик и других.

Многие годы конференция радует своих участников не только гостеприимством, неповторимой атмосферой, уважительным, даже бережным отношением к науке и ученым, но и прекрасной погодой, пьянящим ароматом цветущих яблонь и сирени. Общение в Коломне вызывает порой полярные эмоции, но всегда заканчивается единодушным общением в русле полезной научной дискуссии.

В прошлом году состоялась юбилейная 10-я конференция. Было много планов на будущее, но не всем суждено сбыться – 22 июля 2021 года после непродолжительной болезни умер В.Б. Тарасов, а 9 декабря 2021 года в автомобильной аварии трагически погиб Э.Г. Зафиров.

К сожалению, встречи с этими яркими, живыми, потрясающими людьми остались в прошлом. К счастью, они остались в наших воспоминаниях.

Мы посвящаем 11-ю конференцию ИММВ и издание этих трудов памяти безвременно ушедших за последний год наших друзей и коллег **Валерия Борисовича Тарасова** и **Эвклида Герасимовича Зафирова**, игравших определяющую роль в организации и проведении конференций ИММВ.



*Валерий Борисович Тарасов и Эвклид Герасимович Зафиров
на X Международной научно-практической конференции
«Интегрированные модели и мягкие вычисления
в искусственном интеллекте»*

20 мая 2021 г., Конькобежный центр Московской области «Коломна»

Валерий Борисович Тарасов был выдающимся ученым в области искусственного интеллекта, автором более 200 значимых работ по различным направлениям, включая синергетический искусственный интеллект, семиотическое моделирование, теорию агентов и много-агентных систем, теорию нечетких множеств, мягких вычислений и когнитивных измерений. Его научному наследию посвящен первый пленарный доклад этого сборника.

В начале 90-х годов в квартире Валерия Борисовича Тарасова в Варсонофьевском переулке в доме № 6 собирались участники московского семинара по нечетким множествам, проходившем совсем рядом в Обществе А.С. Попова на улице Кузнецкий мост. На этих же заседаниях сформировался авторский коллектив книги «Нечеткие множества в системах управления и искусственного интеллекта» во главе с Д.А. Пospelовым.

В стихотворении А.Н. Аверкина, посвященном памяти В.Б. Тарасова, передана атмосфера того удивительного времени.

*Картины смотрят с древних стен,
И бесконечные диваны,
И резкий запах перемен
Немножко в этом мире странный.*

*Рождалась мысль, рождался свет
Средь книг, рассыпанных обильно,
Средь варсонофьевских коллег,
Знакомых власти пофамильно.*

*Рождались книги и тосты,
Как вера запрещенной секты,
Как синергетика мечты
И роевого интеллекта.*

*Но смерть назначена была
Коварной и не очень старой,
И Сириус трагично стал
Твоим свиданием в Самарре.*

*Не все успелось написать –
Ты был пророком слишком скромным,
Но вечно будешь каждый май
Ты собирать друзей в Коломне.*

*И много месяцев подряд
Мне снишься ты в объятьях плета.
И рукописи не горят,
И не кончается беседа.*

Валерий Борисович был не просто движущей силой и основным организатором конференций ИММВ, он активно привлекал молодежь. В разные годы в организационный комитет входили студенты, аспиранты и ученики В.Б. Тарасова из МГТУ им. Н.Э. Баумана – Павел Владимирович Афонин, Андрей Витальевич Урусов, Анастасия Петровна Ефремова (Калуцкая), Алёна Валериевна Федотова, Мария Николаевна Королева (Святкина), Александра Львовна Живова, Владислав Сергеевич Дюндюков, Юлия Владимировна Ядгарова, Сергей Сергеевич Колтунов, Никита Викторович Воронцов.

Эвклид Герасимович Зафиров, являясь председателем совета директоров ОАО «Коломенский завод РТИ», депутатом Московской областной Думы, был инициатором, вдохновителем и попатителем всех предыдущих конференций «Интегрированные модели и мягкие вычисления в искусственном интеллекте». Он обладал удивительной способностью помогать, не привлекая к себе внимания, и постоянно прилагал усилия к объединению ученых, специалистов и практиков для реализации различных идей, в том числе в области искусственного интеллекта. По сути, он являлся полпредом РАИИ в Коломне.

Коломенская конференция пронизана сердечной, неповторимой атмосферой. Она стала местом судьбоносных встреч, важных знакомств, положивших начало дружеским отношениям, формированию новых идей, совместным научным работам.

Особые чувства вызывает то, что на конференцию участники приезжают порой без докладов, чтобы проникнуться уникальным духом конференции, послушать коллег, поучаствовать в дискуссиях и почувствовать радость душевного общения. Приехав первый раз, многие из них возвращаются в Коломну вновь.

Как и ранее, организаторами конференции являются: Российская ассоциация искусственного интеллекта, Российская ассоциация нечетких систем и мягких вычислений, Администрация Коломенского городского округа, Московский государственный технический университет им. Н.Э. Баумана. В этом году в состав организаторов также вошли: Федеральный исследовательский центр «Информатика и управление» РАН, Коломенский институт (филиал) Московского политехнического университета, Институт компьютерных технологий и информационной безопасности Южного федерального университета. Мы надеемся сохранить дух конференции и преемственность в её организации.

Тематика конференции охватывает следующие актуальные научные и практические направления в области искусственного интеллекта:

- Синергетический искусственный интеллект, гибридные интеллектуальные системы, объяснимый ИИ.

- Мягкие вычисления, измерения и оценки. Интеллектуальные вычисления. Вычисления, инспирированные природой.
- Моделирование НЕ-факторов в интеллектуальных системах.
- Неклассические логики и семантики в ИИ. Многочленные и модальные логики в ИИ.
- Нечеткие множества, нечеткие отношения, нечеткие графы и нечеткие алгоритмы.
- Нечеткие логики и приближенные рассуждения.
- Нечеткое управление, нечеткая оптимизации, нечеткие модели принятия решений.
- Когнитивные модели в искусственном интеллекте.
- Грануляция информации, гранулярные вычисления, вычисления со словами и фигурами.
- Вероятностные модели в информационных технологиях и искусственном интеллекте. Байесовские сети.
- Машинное обучение.
- Нейросетевые технологии. Нейрологические модели. Нечеткие нейронные сети.
- Бионические подходы, эволюционное моделирование, генетические алгоритмы.
- Автономные агенты и многоагентные системы, распределенный и децентрализованный ИИ.
- Интеллектуальные, коллаборативные, «понимающие» роботы.
- Искусственная жизнь и адаптивное поведение. Роевое познание. Роевая робототехника.
- Интеллектуальные среды и их компоненты.
- Технологии искусственного интеллекта двойного и специального назначения.
- Инженерия знаний на основе измерений. Когнитивные измерения. Сенсорные сети.
- Интеллектуальные производства, интернет вещей, интеллектуальные мехатронные и киберфизические системы.
- Интеллектуальное управление жизненным циклом продукции.

После рецензирования на конференцию было принято 64 доклада. В результате сформирована программа конференции, включающая в себя 5 пленарных докладов ведущих ученых и специалистов, а также 59 секционных докладов, распределенным по 8 секциям.

В конференции принимают участие более 120 ученых, исследователей и специалистов из академических институтов, университетов, отраслевых институтов и организаций Москвы, Санкт-Петербурга,

Белгорода, Владивостока, Коломны, Новомосковска, Ростова-на-Дону, Самары, Смоленска, Таганрога, Твери, Ульяновска, из Азербайджана, Республики Беларуси и Сирийской Арабской Республики.

Выражаем особую благодарность Наталии Александровне Мурзак, Анастасии Петровне Ефремовой, Марии Николаевне Королевой, Наталии Викторовне Ярошевич, Владимиру Михайловичу Мажуге, Анастасии Михайловне Тычинской, Антону Павловичу Жаркову за активное участие в подготовке к печати трудов, в организации и проведении конференции.

На протяжении многих лет наша конференция пользуется вниманием и всесторонней поддержкой Администрации Коломенского городского округа. В этой связи хотелось бы выразить признательность председателю Оргкомитета конференции – главе Коломенского городского округа **Александру Владимировичу Гречищеву**.

Программный и организационный комитеты ИММВ-2022

ПЛЕНАРНЫЕ ДОКЛАДЫ

УДК 004.8

О НАУЧНОМ НАСЛЕДИИ В.Б. ТАРАСОВА*

*Аверкин А.Н., к.ф.-м.н., в.н.с.
ФИЦ «Информатика и управление» РАН
Лаборатория ИИ, нейротехнологий и бизнес-аналитики
РЭУ им. Г.В. Плеханова
Центр искусственного интеллекта Университета «Дубна»
e-mail: averkin2003@inbox.ru*

Валерий Борисович Тарасов являлся одним из ведущих отечественных специалистов в области искусственного интеллекта (ИИ). Им получены значимые результаты в области методологии ИИ, синергетического ИИ, семиотического моделирования, теории агентов и много-агентных систем, теории нечетких множеств, мягких вычислений и измерений. Он активно проводил исследования по направлениям интеллектуальных систем автоматизированного проектирования, организационного проектирования, виртуальных предприятий, моделирования и реинжиниринга сетевых и интеллектуальных предприятий, управления знаниями. Им были предложены инновации в сфере инженерного образования на основе интеллектуальных и сетевых технологий, открытого образования и виртуальных кафедр.

Переводы В.Б. Тарасовым под редакцией С.А. Орловского монографий ведущих европейских учёных в области нечётких систем [1] и теории возможности [2] позволили ему достаточно быстро выйти на высокий уровень понимания методов и моделей теории нечётких множеств.

* Работа выполнена при финансовой поддержке РФФИ, проект №20-07-00770.

Он является соавтором одной из первых отечественных книг по нечётким множествам и их приложениям «Нечеткие множества в системах управления и искусственного интеллекта» [3], автором первой монографии на русском языке в области теории агентов, многоагентных систем и интеллектуальных организаций «От многоагентных систем к интеллектуальным организациям» [4], получившей в 2004 году премию Российской ассоциации искусственного интеллекта за лучшую монографию. В этой книге описана концепция синергетического искусственного интеллекта, развивающая идеи Д.А. Поспелова и появившаяся на несколько лет раньше направления Artificial General Intelligence (искусственный общий интеллект).

В 2007 году В.Б. Тарасов вошёл в состав авторского коллектива первой российской монографии по нечётким гибридным системам [5], посвященной основам мягких вычислений, в которой им проведен анализ и классификация НЕ-факторов в информатике и искусственном интеллекте, указаны способы их формализации на базе нечётких множеств и родственных моделей, представлены математические операции для описания логико-лингвистических связей и модификаторов, изложены теоретические основы гибридных систем, проанализированы варианты построения гибридных интеллектуальных систем и интеллектуальных вычислений, рассмотрены нечёткие эволюционные и генетические гибриды, нечёткие нейронные сети и распределённые нечёткие системы.

В 2001 году под редакцией В.В. Голенкова и В.Б. Тарасова вышла в свет книга «Интеллектуальные обучающие системы и виртуальные учебные организации» [6], где впервые были даны теоретические основы построения и функционирования сетевых, виртуальных организаций в сфере высшего образования.

Интерес В.Б. Тарасова к математической психологии и научные связи с Институтом психологии Академии наук обусловили его участие в коллективной монографии «Математическая психология», посвящённой развитию работ школы В.Ю. Крылова [7], где изложены современные методологические основы математической психологии, особенности психологических измерений, математические модели психических явлений и процессов принятия решений.

Психологические исследования форм, видов, операций мышления, взаимосвязей между мышлением, интеллектом и психикой послужили обоснованием концепции грануляции информации как базовой характеристики мышления и когнитивных процессов в целом, а также предпосылки создания прикладных интеллектуальных систем новых поколений были сформулированы в коллективной монографии «Подходы к моделированию мышления» [8].

Результатом совместных работ В.Б. Тарасова с коллегами из Казахстана и Вьетнама стала монография «Методы и языки для онтологического моделирования» [9], а его вклад в развитие мягких вычислений и измерений отражён в монографии [10].

После выхода монографии [3] В.Б. Тарасов опубликовал статьи, посвящённые применению нечётких отношений в системном анализе [11] и принятии решений [12], а также построению моделей когнитивных процессов на основе нечётких метрик и топологий [13].

Во второй половине 1990-х годов был написан первый цикл статей В.Б. Тарасова по вопросам методологии ИИ [14–21]. В [14] им изложена общая концепция наук об искусственном, описана смена парадигмы ИИ, которая заключается в сдвиге от инженерии знаний к моделированию целостного интеллекта и развитию «социального» подхода, связанного с исследованием и проектированием многоагентных систем и их сообществ, проанализированы особенности гибридных интеллектуальных систем типа мягких вычислений и отмечена роль технологий виртуальной реальности в разработке искусственных сред, проанализированы истоки и перспективы междисциплинарной научно-технической области «искусственная жизнь».

В 1998 году В.Б. Тарасов опубликовал ключевую работу «Моделирование психических образов: как совместить дискретное и непрерывное?» [15].

В статьях [15, 16] им представлен системно-организационный подход в ИИ, акцентирующий внимание на идее моделирования интеллекта как открытой системы, в публикациях предложена концепция синергетического искусственного интеллекта [17, 18]. Тогда же были опубликованы работы по эволюционной семиотике [19], теории агентов и многоагентным системам [20], а также их логическому моделированию [21]. Завершением этого цикла стала монография [4].

В.Б. Тарасов являлся инициатором дискуссии «Моделирование НЕ-факторов: ключевое направление ИИ в начале XXI-го века» [22]. Среди его работ по нечётким множествам и нечётким логикам в 2000-е годы следует отметить исследование видов и свойств операций отрицания в многозначных и нечётких логиках [23], нечёткие и бирешётчатые расширения логических матриц [24], а также предложенный им единый подход к описанию нестандартных нечётких множеств (нечёткие множества уровня, интервальнозначные нечёткие множества, интуиционистские нечёткие множества, нечёткие мультимножества) на базе параметризованных нечётких множеств и бирешёток [25].

В.Б. Тарасов находил время для написания обзорных докладов и презентаций, которые включали:

- результаты анализа и моделирования НЕ-факторов на биполярных шкалах [26];
- модели нестандартных логических семантик [27], в том числе многомерных и интервальнозначных, интуиционистских и паранепротиворечивых семантик, а также описание аппарата бирешёток [28], обеспечивающего единый подход к построению этих семантик;
- обоснование перехода от монологических к диалогическим моделям в русле синергетического ИИ [29] и разработанные им логики диалогов и переговоров между агентами [30];
- модели нечёткой грануляции информации на основе неклассических множеств, нестандартных и гибридных нечётких множеств [31];
- формализации обобщённых истинностных значений, расширенных логических матриц, логических миров и пространств в контексте моделирования когнитивных процессов на основе универсальной логики и гранулярных логических структур [32].

В конце 1990-х годов В.Б. Тарасов заинтересовался вопросами применения методов и моделей теории агентов и многоагентных систем в реинжиниринге процессов, теории организаций и организационном проектировании, включая построение виртуальных и интеллектуальных предприятий [33–35]. Им был сделан ряд докладов: по биореинжинирингу предприятий [36], инфраструктуре виртуальных организаций [37], управлению знаниями в сетях предприятий [38], агентно-ориентированному подходу к проектированию виртуальных и интеллектуальных предприятий [39], методологии синергетических организаций [40]. Завершением этой серии работ стала статья «Реинжиниринг процессов, виртуальные организации, управление знаниями, что дальше?» [41]. В ней автор дал ряд прогнозов, касающихся перехода к имитационному моделированию процессов и продуктов предприятий («цифровых двойников»), сдвига от управления к сотрудничеству, подъёма виртуальных организаций, как мы сейчас видим – прогнозы оправдались.

Следующий цикл работ был посвящён анализу влияния новых технологических инициатив и моделей знаний на формирование новых экономических концепций. В.Б. Тарасовым были опубликованы работы о роли НЕ-факторов в экономике знаний и сетевых организациях [42], переходе от информационного общества и накопления интеллектуального капитала в сетях организаций к интеллектуальной экономике [43], построении новой экономики в смарт-городах на основе развития интеллектуальных сред [44]. Особое место в этом ряду занимает работа «Экономический кризис и синергетические организации» [45], написанная по следам мирового кризиса 2008 года.

Начиная с 2013 года В.Б. Тарасовым был сделан ряд ключевых докладов по инжинирингу предприятий и управлению знаниями в контексте инициативы Индустрия 4.0 [46–51].

Ещё одна важная сторона научной деятельности В.Б. Тарасова – постоянный интерес к истории развития системологии, теории нечётких множеств и искусственного интеллекта в СССР и России и к этапам жизненного пути и научного наследия выдающихся учёных – А.А. Богданова [52], Л. Заде [53–57], Д.А. Поспелова [58–60], Э.В. Попова [61], В.В. Мартынова [62], А.С. Нариньяни [63] и других. По сути, В.Б. Тарасов являлся основным исследователем развития этих научных направлений.

В 2010-е годы он выступил в качестве инициатора и активного участника работ по ряду новых направлений ИИ, когнитивной науки, мягких вычислений и теории измерений, включая моделирование физических когнитивных агентов (роботов) на базе псевдофизических логик, опирающихся на распространение обобщённых ограничений [64], описание гранулярных структур измерений [65] и методов построения гранулярных логических значений и систем [66], разработку интеллектуальных систем на основе когнитивных измерений [67], развитие теоретических основ синергетического ИИ и построение бирешётки переговоров между агентами [68], формирование общей архитектуры деятельности агента [69].

В 2015 году с междисциплинарного анализа проблемы понимания [70, 71] начался цикл работ В.Б. Тарасова, посвящённых когнитивным сенсорам [72, 73] – информационно-измерительным устройствам, обеспечивающим грануляцию измерительной информации с интерпретацией результатов измерений на основе многозначных логических семантик средств когнитивной графики – и «понимающим» агентам [74, 75]. С 2018 года проблемы понимания для искусственных агентов стали рассматриваться им в русле коллаборативной робототехники, связанной с развитием концепций гибридного интеллекта применительно к системам «человек–коллаборативный робот», разработкой расширенного партнёрского интерфейса между человеком и роботом, и реализацией его компонентов [76–80].

В.Б. Тарасов руководил десятками исследовательских проектов, включающими следующие направления:

- теория и методология искусственного интеллекта и искусственной жизни, в том числе, развитие системного подхода в ИИ и синергетического ИИ, теория агентов, многоагентные системы и интеллектуальные организации, теория нечетких множеств и нечетких меры неопределенности и теория возможностей, мягкие вычисления и вычисления со словами, неклассические логики и небулевы алгебры, оппозиционные шкалы и семиотическое моделирование;

- прикладные интеллектуальные системы, в том числе, интеллектуальные системы автоматизированного проектирования и совмещенной разработки, компьютерно-интегрированные производства и виртуальные предприятия, организационное проектирование, моделирование и реинжиниринг предприятий, управление знаниями, создание интеллектуальных производств, сетевых и интеллектуальных предприятий;
- инновации в сфере инженерного образования и профессиональной подготовки персонала, в первую очередь, на основе интеллектуальных и сетевых технологий, открытое образование и виртуальные кафедры;
- психологические и эргономические аспекты искусственного интеллекта.

Его последний неоконченный проект РФФИ был посвящен разработке фундаментальных проблем построения «понимающих» когнитивных агентов, многоагентных систем и искусственных сообществ на основе подходов синергетического искусственного интеллекта, методов грануляции информации, динамических биполярных шкал и диалогических миров. Но научный коллектив, сформированный В.Б. Тарасовым, не позволил этому проекту прекратиться, работа над ним продолжается.

На юбилейной X Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2021) в Коломне В.Б. Тарасов выступил с фундаментальным докладом «От объяснимого искусственного интеллекта к «понимающим» когнитивным агентам» [81].

В его ближайших планах была поездка в Турцию с пленарным докладом на XI Международную конференцию по теории и применению мягких вычислений, вычислений со словами и искусственному интеллекту (ICSCCW-2021). Однако в трудах этой конференции была опубликована уже мемориальная статья про Валерия Борисовича [82].

В.Б. Тарасов – автор более 250 научных работ, включая монографии «Нечеткие множества в моделях управления и искусственного интеллекта», «Интеллектуальные обучающие системы и виртуальные учебные организации», «От многоагентных систем к интеллектуальным организациям», «Нечеткие гибридные системы. Теория и практика», «Методы и языки для онтологического моделирования». Он является одним из наиболее цитируемых российских авторов в области искусственного интеллекта. В статье [83], посвященной его 65-летнему юбилею, приведена лишь небольшая часть его библиографии.

Стиль жизни Валерия Борисовича он сам точно охарактеризовал двумя словами «собиратель» и «путешественник». Собиратель книг (у него была очень большая домашняя библиотека), собиратель «науч-

ных земель», не признающий жёстких границ между разными научными дисциплинами, собиратель людей науки на конференции и симпозиумы, где всегда царилла благожелательная и дружеская атмосфера, присущая его духу. И лёгкий на подъём путешественник – как в прямом, так и в переносном смысле. Для него необходимыми и чрезвычайно увлекательными являлись путешествия из одной научной области в другую: из инженерных наук в психологию, из психологии в сферу информационных технологий и искусственного интеллекта, из информатики, инженерии знаний и теории сетей – в экономику, и снова в новые технологии четвёртой промышленной революции, воплощающие идеи гибридного и синергетического интеллекта. И его поездки как докладчика на всероссийские и международные конференции: от Калининграда до Владивостока, от озера Байкал до Смоленска и Сочи, от Ростова-Дону и Таганрога до Казани и Ульяновска, от Ташкента до Канкуна, от Будапешта и Братиславы до Лиссабона и Рио-де-Жанейро для получения нового заряда творческой энергии.

Литература

1. Негойцэ К. Применение теории систем к проблемам управления: пер. с англ. В.Б. Тарасова / под ред. С.А. Орловского. – М.: Мир, 1981.
2. Дюбуа Д., Прад А. Теория возможностей. Приложения к представлению знаний в информатике: пер. с франц. В.Б. Тарасова / под ред. С.А. Орловского. – М.: Радио и связь, 1990.
3. Аверкин А.Н., Батыршин И.З., Блишун А.Ф., Силов В.Б., Тарасов В.Б. Нечёткие множества в моделях управления и искусственного интеллекта / под ред. Д.А. Поспелова. – М.: Наука, 1986.
4. Тарасов В.В. От многоагентных систем к интеллектуальным организациям. – М.: Эдиториал УРСС, 2002.
5. Батыршин И.З., Недосекин А.О., Стецко А.А., Тарасов В.Б., Язенин А.В., Ярушкина Н.Г. Нечёткие гибридные системы. Теория и практика / под ред. Н.Г. Ярушкиной. – М.: Физматлит, 2007.
6. Интеллектуальные обучающие системы и виртуальные учебные организации / под ред. В.В. Голенкова и В.Б. Тарасова. – Мн.: БГУИР, 2001.
7. Математическая психология: Школа В.Ю. Крылова / под ред. А.Л. Журавлёва, Т.Н. Савченко, Г.М. Головиной. М.: Институт психологии РАН, 2010.
8. Подходы к моделированию мышления / под ред. В.Г. Редько. – М.: ЛЕНАНД, 2014.
9. Плесневич Г.С., Тарасов В.Б., Карабеков Б.С., Нгуен Тхи Минь Ву. Методы и языки для онтологического моделирования. – Алматы: ИИВТ, 2017.
10. Прокопчина С.В., Тарасов В.Б., Лазарев В.Л., Горохов В.Л., Куприянов М.М. Мягкие вычисления и измерения. Т. 1. Теоретические основы и методы / под ред. С.В. Прокопчиной. – М.: ИД «Научная библиотека», 2017.
11. Тарасов В.Б. Нечеткие отношения и моделирование открытых иерархических систем // Математические методы оптимизации и управления в сложных системах. – Калинин: КГУ, 1986. – С. 3-17.

12. Тарасов В.Б. Моделирование предпочтений в задачах принятия решений параметризованными нечеткими отношениями // *Нечеткие системы: моделирование структуры и оптимизация*. – Калинин: КГУ, 1987. – С. 17-31.
13. Тарасов В.Б., Шостак А.П. Нечеткая топология в моделировании когнитивных процессов // *Нечеткие системы: модели и программные средства*. – Тверь: ТГУ, 1991. – С. 43-48.
14. Тарасов В.Б. От искусственного интеллекта к искусственной жизни: новые направления в науках об искусственном // *Новости искусственного интеллекта*. – 1995. – № 4. – С. 93-117.
15. Тарасов В.В. Моделирование психических образов: как совместить дискретное и непрерывное? // *Новости искусственного интеллекта*. – 1998. – № 3. – С. 86-100.
16. Тарасов В.Б. Системно-организационный подход в искусственном интеллекте // *Программные продукты и системы*. – 1997. – № 3. – С. 6-13.
17. Тарасов В.Б. Синергетические проблемы в искусственном интеллекте // *Труды международной научно-практической конференции «Знание – диалог – решение»*. Т. 2 (KDS-2001, Санкт-Петербург, 19-22 июня 2001 г.). – СПб: Лань, 2001. – С. 594-602.
18. Тарасов В.Б. Синергетический искусственный интеллект: истоки и перспективы // *Труды международного конгресса «Искусственный интеллект в XXI веке» (ICAI'2001, Дивноморск, Россия, 3-8 сентября 2001 г.)*. – М.: Наука, Физматлит, 2001. – С. 559-570.
19. Тарасов В.Б. Эволюционная семиотика и нечеткие многоагентные системы – основные теоретические подходы к построению интеллектуальных организаций // *Информационные технологии и вычислительные системы*. – 1998. – № 1. – С. 54-68.
20. Тарасов В.Б. Агенты, многоагентные системы, виртуальные сообщества: стратегическое направление в информатике и искусственном интеллекте // *Новости искусственного интеллекта*. – 1998. – № 2. – С. 5-63.
21. Тарасов В.Б. Нетрадиционные и гибридные логики в моделировании интеллектуальных агентов. I. Искусственные деятели, интенциональные характеристики и пути их моделирования // *Известия РАН: Теория и системы управления*. – 2000. – № 5. – С. 5-17.
22. Тарасов В.Б. НЕ-факторы: от семиотического анализа к методам формализации // *Новости искусственного интеллекта*. – 2004. – № 2. – С. 95-114.
23. Тарасов В.Б. Операции отрицания в многозначных и нечетких логиках // *Нечеткие системы и мягкие вычисления: Труды Всероссийской научной конференции*. – М.: Физматлит, 2006. – С. 13-30.
24. Тарасов В.Б. От алгебраической модели Тарского к логико-семиотической матрице // *Труды IX Национальной конференции по искусственному интеллекту КИИ-2004 (Тверь, 28 сентября – 2 октября 2004 г.)*. Т. 1. – М.: Физматлит, 2004. – С. 311-322.
25. Тарасов В.Б. От параметризованных нечетких множеств к нечетким мультимножествам уровня // *Нечеткие системы и мягкие вычисления: Сборник научных трудов II-й Всероссийской научной конференции (НСМВ-2008, Ульяновск, 27–29 октября 2008 г.)*. – Ульяновск: УлГТУ, 2008. – С. 40-48.

26. Тарасов В.Б. Анализ и моделирование НЕ-факторов на полярных шкалах // Интегрированные модели и мягкие вычисления в искусственном интеллекте: Сборник трудов Международного научно-практического семинара (ИММВ-2001, Коломна, 17–18 мая 2001 г.). – М.: Физматлит, 2001. – С. 65-71.
27. Тарасов В.Б. О путях построения нестандартных и гибридных нечетких логик // Сборник трудов IV Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2007, Коломна, 28–30 мая 2007 г.). Т. 1. – М.: Физматлит, 2007. – С. 101-113.
28. Тарасов В.Б. Гибридные операции на бирешетках // Сборник трудов III Международного научно-практического семинара «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2005, Коломна, 15–17 мая 2005 г.). – М.: Физматлит, 2005. – С. 63-68.
29. Тарасов В.Б. От монологических к диалогическим подходам в искусственном интеллекте // Сборник трудов V Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2009, Коломна, 28-30 мая 2009 г.). Т. 1. – М.: Физматлит, 2009. – С. 149-162.
30. Тарасов В.Б. От логических к диалогическим бирешеткам: использование в теории агентов // Вестник Ростовского государственного университета путей сообщения. – 2011. – № 3. – С. 127-139.
31. Тарасов В.Б. Грануляция информации, нестандартные и гибридные нечеткие множества // Сборник трудов VI Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2011, Коломна, 16–19 мая 2011 г.). Т. 1. – М.: Физматлит, 2011. – С. 35-49.
32. Тарасов В.Б. Универсальная логика и грануляция информации: новые подходы к моделированию когнитивных процессов и систем // Сборник научных трудов VIII Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2015, Коломна, 18–20 мая 2015 г.). Т. 1. – М.: Физматлит, 2015. – С. 107-122.
33. Тарасов В.Б. Новые стратегии реорганизации и автоматизации предприятий: на пути к интеллектуальным предприятиям // Новости искусственного интеллекта. – 1996. – № 4. – С. 40-84.
34. Тарасов В.Б. Предприятия XXI-го века: проблемы проектирования и управления // Автоматизация проектирования. – 1998. – № 4 (10). – С. 45-52.
35. Тарасов В.Б., Шильников П.С. Виртуальные предприятия: свойства, технология создания, компоненты инфраструктуры // Информационные технологии. – 2000. – № 9. – С. 2-7.
36. Тарасов В.Б. От реинжиниринга процессов к биореинжинирингу предприятий: системно-эволюционное проектирование организаций // Сборник научных трудов II Российской научно-практической конференции «Реинжиниринг бизнес-процессов на основе современных информационных технологий» (Москва, 19–20 мая 1998 г.). – М.: МЭСИ, 1998. – С. 68-82.
37. Тарасов В.Б. Виртуальные предприятия: организационные характеристики и компоненты инфраструктуры // Сборник научных трудов IV Российской научно-практической конференции «Реинжиниринг бизнес-процессов на основе современных информационных технологий» (Москва, 11–12 апреля 2000 г.). – М.: МЭСИ, 2000. – С. 21-33.

38. Тарасов В.Б. Управление знаниями на сетевых предприятиях // Сборник научных трудов VI Научно-практической конференции «Реинжиниринг бизнес-процессов на основе современных информационных технологий. Системы управления знаниями» (Москва, 19–20 марта 2002 г.). – М.: МЭСИ, 2002. – С. 47-53.
39. Тарасов В.Б. Агентно-ориентированный подход к формированию интеллектуальных и виртуальных предприятий // Сборник научных трудов Международной научно-практической конференции «Реинжиниринг бизнес-процессов на основе современных информационных технологий. Системы управления знаниями» (РБП-СУЗ-2005, Москва, 9–10 июня 2005 г.). – М.: МЭСИ, 2005. – С. 37-44.
40. Тарасов В.Б. Синергетические организации в сетевой экономике знаний // Мир электронного обучения. – 2006. – № 4. – С. 68-76.
41. Тарасов В.Б. Реинжиниринг процессов, виртуальные организации, управление знаниями – что дальше? // Сборник научных трудов X Российской научно-практической конференции «Реинжиниринг бизнес-процессов на основе современных информационных технологий. Системы управления знаниями» (Москва, 17–18 апреля 2007 г.). – М.: МЭСИ, 2007. – С. 3-12.
42. Тарасов В.Б. НЕ-факторы в экономике знаний и сетевых организациях // Сборник научных трудов XI Российской научно-практической конференции «Реинжиниринг бизнес-процессов на основе современных информационных технологий. Системы управления знаниями» (РБП-СУЗ-2008, Москва, МЭСИ, 23-24 апреля 2008 г.). – М.: МЭСИ, 2008. – С. 247-253.
43. Тарасов В.Б. От информационного общества к интеллектуальной экономике // Сборник научных трудов XIV Научно-практической конференции «Реинжиниринг бизнес-процессов на основе современных информационных технологий. Системы управления знаниями» (РБП-СУЗ-2011, Москва, МЭСИ, 28–29 апреля 2011 г.). – М.: МЭСИ, 2011. – С. 286-298.
44. Тарасов В.Б. Интеллектуальные среды: технологическая основа новой экономики в смарт-городах // Сборник научных трудов XV Научно-практической конференции «Реинжиниринг бизнес-процессов на основе современных информационных технологий. Системы управления знаниями» (РБП-СУЗ-2012, Москва, МЭСИ, 26–27 апреля 2012 г.). – М.: МЭСИ, 2012. – С. 180-194.
45. Тарасов В.Б. Экономический кризис и синергетические организации // Экономика, статистика и информатика // Вестник УМО. – 2009. – № 3. – С. 29-35.
46. Тарасов В.Б. Управление жизненными циклами продукции и предприятия – ключевой аспект инжиниринга сетевых предприятий // Сборник научных трудов XVII Научно-практической конференции «Инжиниринг предприятий и управление знаниями» (ИП&УЗ-2014, Москва, МЭСИ, 24–25 апреля 2014 г.). – М.: МЭСИ, 2014. – С. 245-255.
47. Тарасов В.Б. Инжиниринг предприятий и организационные онтологии // Сборник научных трудов XVIII Научно-практической конференции «Инжиниринг предприятий и управление знаниями» (ИП&УЗ, Москва, МЭСИ, 21–24 апреля 2015 г.). – М.: МЭСИ, 2015. – С. 25-41.

48. Тарасов В.Б. Стратегический инжиниринг предприятий будущего: массовое сотрудничество, интернет вещей, инициатива «Индустрия 4.0», что дальше? // Сборник научных трудов XIX Научно-практической конференции «Инжиниринг предприятий и управление знаниями» (ИП&УЗ-2016, Москва, ФГБОУ ВО «РЭУ им. Г.В. Плеханова», 26–27 апреля 2016 г.). – М.: РЭУ им. Г.В. Плеханова, 2016. – С. 57-68.
49. Тарасов В.Б., Овсянников М.В. Развитие технологий Индустрии 4.0: от цифрового производства и интернета вещей до коллаборативных роботов // Сборник научных трудов XIX Научно-практической конференции «Инжиниринг предприятий и управление знаниями» (ИП&УЗ-2018, Москва, ФГБОУ ВО «РЭУ им. Г.В. Плеханова», 26–28 апреля 2018 г.). Т. 1. – М.: РЭУ им. Г.В. Плеханова, 2018. – С. 52-60.
50. Тарасов В.Б. Технологии Индустрии 4.0: от цифрового производства и интернета вещей до коллаборативных роботов и интеллектуального имитационного моделирования // Мягкие измерения и вычисления. – 2018. – Т. 2, № 12. – С. 4-19.
51. Tarassov V.B. Enterprise total agentification as a way to industry 4.0: Forming artificial societies via goal-resource networks // Advances in Intelligent Systems and Computing. – 2019. – Vol. 874. – P. 26-40.
52. Тарасов В.Б. Тектология А.А. Богданова и неоклассическая теория организаций – предвестники эры реинжиниринга // Проблемы теории и практики управления. – 1998. – № 4. – С. 67-72.
53. Аверкин А.Н., Батыршин И.З., Тарасов В.Б. Лотфи Заде и Россия // Новости искусственного интеллекта. – 2001. – № 2-3. – С. 16-20.
54. К 90-летию Лотфи Заде // Нечёткие системы и мягкие вычисления. – 2011. – Т. 6, № 1. – С. 5-9.
55. К 97-летию со дня рождения Лотфи Заде // Мягкие измерения и вычисления. – 2018. – Т. 2, № 6. – С. 4-23.
56. Tarassov V. A panorama of fuzzy sets: Dedicated to the 100th anniversary of the birth of Lotfi Zadeh // CEUR Workshop Proceedings. – 2020. – Vol. 2782. – P. 1-10.
57. Aliev R.A., Tarassov V.B. The Man Who Changed the Scientific World: To the Centenary of the Birth of Lotfi Zadeh // Advances in Intelligent Systems and Computing. – 2021. – Vol. 1306. – P. 148-164.
58. Тарасов В.Б. Д.А. Поспелов – основоположник искусственного интеллекта в СССР и России // Политехнические чтения. Вып. 7. Искусственный интеллект – проблемы и перспективы. – М.: Политехнический музей – РАИИ, 2006. – С. 34-47.
59. Тарасов В.Б. Искусственный интеллект: от психоники и прикладной семиотики к синергетическому подходу. о научном наследии профессора Д.А. Поспелова // Развитие вычислительной техники в России, странах бывшего СССР и СЭВ: история и перспективы. Труды Пятой международной конференции. – М., 2020. – С. 296-303.
60. Тарасов В.Б. Искусственный интеллект: от прошлого к будущему. Этапы жизненного пути и научное наследие профессора Д.А. Поспелова // Искусственный интеллект и принятие решений. – 2020. – № 2. – С. 12-42.
61. Тарасов В.Б. Эдуард Викторович Попов: научное наследие и страницы биографии // ЦДУ РАН: Секция кибернетики, 10 февраля 2016 г.

62. Тарасов В.Б. От семантического кода к когнитивной лингвистике, семиотике и информатике: наследие В.В. Мартынова // Материалы IV Международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» (Минск, БГУИР, 20–22 февраля 2014 г.). – Мн.: БГУИР, 2014. – С. 39-48.
63. Рыбина Г.В., Тарасов В.Б., Липатов А.А. К 80-летию со дня рождения А.С. Нариньяни. 35 лет концепции недоопределённости // ЦДУ РАН: Секция кибернетики, 8 ноября 2017 г.
64. Калущая А.П., Тарасов В.Б. Моделирование взаимодействия искусственного агента с внешней средой на основе пространственных логик и обобщенных ограничений // Интеллектуальные системы. Вып. 4 / под ред. В.М. Курейчика. – М.: Физматлит, 2010. – С. 105-142.
65. Тарасов В.Б. Гранулярные структуры измерений в интеллектуальных средах: васильевские и белнаповские сенсоры и модели их взаимодействия // Информационно-измерительные и управляющие системы. – 2013. – № 5. – С. 65-74.
66. Тарасов В.Б. О методах построения гранулярных логических значений и структур // Труды VI Всероссийской научно-практической конференции «Нечёткие системы и мягкие вычисления» (Санкт-Петербург, СПИИРАН, 27–29 июня 2014 г.). Т. 2. – СПб: Политехника-сервис, 2014. – С. 33-44.
67. Тарасов В.Б., Святкина М.Н. Интеллектуальные системы на основе когнитивных измерений // Сборник научных трудов VII Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2013, Коломна, 20–22 мая 2013 г.). Т. 2. – М.: Физматлит, 2013. – С. 611-623.
68. Тарасов В.Б. Синергетический искусственный интеллект: моделирование переговоров между автономными агентами // Материалы II Международного Поспеловского симпозиума «Гибридные и синергетические интеллектуальные системы» (Светлогорск, Калининградская обл., 30 июня – 6 июля 2014 г.). – Калининград: БФУ им. И. Канта, 2014. – С. 340-358.
69. Тарасов В.Б. От спецификации когнитивных и инженерии интенций к обобщенной архитектуре деятельности агентов // Материалы III Всероссийской Поспеловской конференции «Гибридные и синергетические интеллектуальные системы» (Светлогорск, Калининградская область, 6 июня – 11 июня 2016 г.). – Калининград: БФУ им. И. Канта, 2016. – С. 94-114.
70. Тарасов В.Б. Проблема понимания: настоящее и будущее искусственного интеллекта // Материалы V Международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2015, Минск, БГУИР, 19–21 февраля 2015 г.). – Мн.: БГУИР, 2015. – С. 25-42.
71. Svyatkina M.N., Tarassov V.B. Toward Logical Models of Understanding: Axiology, Logical Pragmatics, Pragmatic Logics, What Else? // Handbook of the 5th World Congress and School on Universal Logic (UNILOG'2015, Istanbul, Turkey, June 20-30, 2015). – Istanbul University, 2015. – P. 260-262.
72. Святкина М.Н., Тарасов В.Б. Логико-алгебраические методы построения когнитивных сенсоров // Материалы VI Международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2016, Минск, БГУИР, 18–20 февраля 2016 г.). – Мн.: БГУИР, 2016. – С. 331-348.

73. Svyatkina, M.N., Tarassov, V.B., Dolgiy, A.I. Logical-algebraic methods in constructing cognitive sensors for railway infrastructure intelligent monitoring system // *Advances in Intelligent Systems and Computing*. – 2016. – Vol. 450. – P. 191-206.
74. Тарасов В.Б. Цветные ресурсно-целевые сети: приложение к задаче понимания поведения взаимодействующих агентов // Труды VII Всероссийской научно-практической конференции «Нечёткие системы, мягкие вычисления и интеллектуальные технологии» (НСМВИТ-2017, Санкт-Петербург, 3-7 июля 2017 г.). Т. 1. – СПб.: Политехника-сервис, 2017. – С. 146-159.
75. Тарасов В.Б. Биполярные шкалы, многозначные семантики и решётки модальностей в моделировании когнитивных агентов // Материалы IV Всероссийской Поспеловской конференции «Гибридные и синергетические интеллектуальные системы» (Светлогорск, Калининградская область, 14–19 мая 2018 г.). – Калининград: БФУ им. И. Канта, 2018. – С. 98-115.
76. Karpov V.E., Tarassov V.B. Synergetic artificial intelligence and social robotics // *Advances in Intelligent Systems and Computing*. – 2018. – Vol. 679. – P. 3-15.
77. Карпов В.Э., Тарасов В.Б. От коллаборативной робототехники к социальным роботам для поддержки людей с ограниченными возможностями: новые направления разработки использования интеллектуальных агентов // Интеллектуальные технологии и средства реабилитации и абилитации людей с ограниченными возможностями (ИТССР-2018). Труды III международной конференции. – М.: МГЭУ, 2018. – С. 20-29.
78. Tarassov V.B. The Next Stage of Industry 4.0: From Cognitive to Collaborative and Understanding Agents // *Открытые семантические технологии проектирования интеллектуальных систем*. – 2019. – № 3. – С. 187-196.
79. Тарасов В.Б. Общение и взаимопонимание когнитивных агентов в коллаборативной робототехнике: на пути к «сознательным» роботам // Гибридные и синергетические интеллектуальные системы. Материалы V Всероссийской Поспеловской конференции с международным участием. – Калининград: БФУ им. И. Канта, 2020. – С. 74-101.
80. Тарасов В.Б. Гибридный интеллект и коллаборативная робототехника: расширенный партнерский интерфейс в системах "Человек-робот" // *Мягкие измерения и вычисления*. – 2020. – Т. 29, № 4. – С. 45-67.
81. Тарасов В.Б. От объяснимого искусственного интеллекта к «понимающим» когнитивным агентам // Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов X-й Международной научно-технической конференции (ИММВ-2021, Коломна, 17-20 мая 2021 г.). В 2-х томах. Т. 1. – Смоленск: Универсум, 2021. – С. 175-189.
82. Aliev R.A., Averkin A. About Valery Borisovich Tarasov (16.02.1955–22.07.2021) // In: Aliev R.A., Kasprzyk J., Pedrycz W., Jamshidi M., Babanli M., Sadikoglu F.M. (eds) 11th International Conference on Theory and Application of Soft Computing, Computing with Words and Perceptions and Artificial Intelligence. ICSCCW-2021. *Lecture Notes in Networks and Systems*. Springer, Cham, 2022. – Vol. 362. – P. 3-5.
83. Валерию Борисовичу Тарасову – 65 лет // *Мягкие измерения и вычисления*. – 2020. – Т. 29, № 4. – С. 143-52.

БИОИНСПИРИРОВАННЫЕ МОДЕЛИ ПОВЕДЕНИЯ. СИСТЕМНЫЙ КРИЗИС ИЛИ НОВЫЕ ГОРИЗОНТЫ?*

*Карнов В.Э., к.т.н., доцент
НИЦ «Курчатовский институт»
e-mail: karpov-ve@yandex.ru*

1. ВВЕДЕНИЕ

Считается, что одним из стратегических направлений искусственного интеллекта (ИИ) является исследование и создание моделей поведения, причем как искусственных агентов, так и человека. В этой работе мы сосредоточимся именно на моделях поведения искусственных агентов, причем относящихся к, так называемым, биоинспирированным.

Использование различных заимствованных из природы моделей поведения при создании технических систем началось еще на заре кибернетики. Это знаменитые «черепашки» Г. Уолтера, «Одноклеточное» Ф. Люкса, гомеостат Р. Эшби [1] и многие другие, и работы Института автоматки и телемеханики АН СССР, 1956–1957 гг. Хороший обзор подобного рода работ, которые были посвящены реализации на аппаратном уровне простейших форм поведения, приведен в книге М.Г. Гаазе-Рапопорта и Д.А. Поспелова [2]. Помимо условно-рефлекторных, создавались и более сложные модели, описывающие коллективное взаимодействие агентов. Прежде всего, это работы школы М.Л. Цетлина по поведению автоматов. Одним из основных тезисов этих исследований являлось утверждение, что любое достаточно сложное поведение складывается из совокупности простых поведенческих актов [3]. В результате исследований разнообразных форм поведения, была создана теоретическая основа для описания коллективного взаимодействия объектов самой разной природы [4]. В частности, на базе подобных моделей был разработан фундамент сотовой связи (задача о коллективе радиостанций, которым надо «договариваться» между собой о мощности своих сигналов) [5, 6].

* Работа выполнена при финансовой поддержке РФФИ, проект №20-07-00770.

Но уже здесь виден отход от явной биоинспирированности, характерной для ранних исследований. Модели становятся все более абстрактными, описывающими внешнюю феноменологию поведения и относящимися уже к категории «черного ящика». В них уже не рассматриваются вопросы составляющих их базисных элементов и, как следствие, опускаются столь важные вопросы генезиса механизмов поведения. В уже упомянутой работе [2] приведен ряд примеров реализации механизмов поведения такого рода, и дана некая схема классификации типов поведения, Рис. 1.

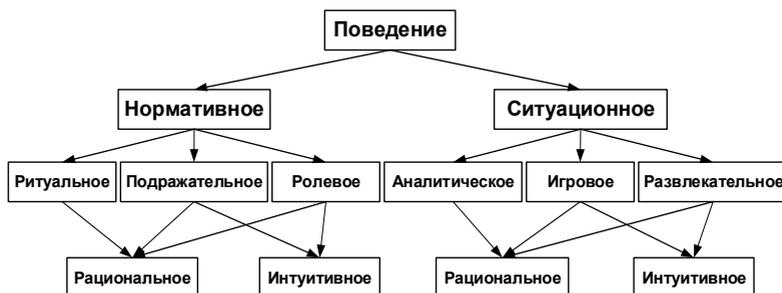


Рис. 1. Типы поведения по М.Г. Гаазе-Раппопорту и Д.А. Поспелову

Важно, что эта схема рассматривает поведение с принципиально иной точки зрения, нежели это принято в биологии, точнее, в этологии – разделе биологии, в котором изучается поведение животных.

В некотором смысле можно сказать, что фокус интересов ИИ к проблемам поведения сместился в сторону прагматического, феноменологического направления. Отчасти это было связано с тем, что ожидания в области создания моделей, адекватных природным объектам, оказались завышенными.

2. ВОЗВРАТ К БИОИНСПИРИРОВАННОСТИ

Вместе с тем, основная идея – заимствование у природы методов и способов решения тех или иных (прежде всего, технических) задач – по-прежнему оставалась. Правда, претерпевая некоторые изменения в расстановке акцентов. Так, наряду с уходом ИИ в область прагматики (или в противовес этому), появилось направление, которое условно можно называть «Искусственная жизнь».

«Искусственная жизнь». Чтобы подчеркнуть биологическую инспирированность исследуемых принципов и механизмов, был введен термин «анимат» – искусственная модель животного. С. Вильсон в

[7] определяет четыре базовые характеристики такого простого животного, однако мы не будем останавливаться на них. Отметим лишь, что речь идет о том, что это животное живет в мире сенсорных сигналов; оно способно к действиям, изменяющим эти сигналы; ряд сигналов является значимым для животного; деятельность животного направлена на оптимизацию уровня некоторых сигналов. Интересно, что в работе [8] Вильсон говорит о том, что «мы также применяем неформально термин *анимат* к животным и самим автономным роботам». Здесь следует отметить, что одной из первых деклараций о создании некой виртуальной сущности, моделирующей организацию поведения, является работа М.М. Бонгарда с коллегами [9]. В ней описывалась "попытка построить модель, воспроизводящую в грубых чертах поведение человека или хотя бы «разумных» животных». В нашей стране развитие теории аниматов и адаптивных агентов относится, прежде всего, к школам В.Г. Редько (см., например, работы [10, 11]), К.В. Анохина и их коллег [12].

Групповое управление. Большой интерес к биоинспирированным моделям возродился со становлением такого направления, как системы групповой робототехники (ГР). Основным тезисом ГР является утверждение о целесообразности решения сложных задач совокупностью взаимодействующих между собой простых технических устройств – роботов. При этом образцом для подражания были объявлены социальные животные, прежде всего – насекомые. Стали появляться различные коллективные, стайные, роевые модели, под каждую из которых подводилась биоинспирированная (с точки зрения поведения) база. Однако ожидаемых синергетических эффектов в явном виде получено так и не было, разнообразие форм моделей организации ограничилось в основном базовым роевым уровнем, а типовые задачи ГР свелись к совместному движению, см., например, обзор [13]. Да и вообще выяснилось, что лучше всего работают системы с централизованным управлением.

Причины такого положения дел заключаются в том, что если начальный, «романтический» период становления биоинспирированного поведения характеризовался тесными связями робототехников, специалистов в области интеллектуального управления (далее будем называть их обобщенным термином – «кибернетики»), инженеров и биологов (в целом), то в дальнейшем связи эти перестали носить систематический характер. Возникли проблемы взаимодействия.

3. ПРОБЛЕМЫ ВЗАИМОДЕЙСТВИЯ

Разработка моделей поведения, да еще биоинспирированных, по определению является междисциплинарной областью, однако чаще всего вместо кооперации мы видим вполне четкую границу между «инженерами», «кибернетиками» и «биологами». О причинах этого будет сказано ниже, а пока приведем несколько примеров.

Кибернетика без биологии

Непонимание или предвзятость специалистов в области ИИ («кибернетиков») к биологическим механизмам можно разделить на следующие категории.

Первой категорией такой предвзятости является отношение «технических» специалистов к некоторым биологическим механизмам и феноменам животного мира. Например, к эмоциям и альтруизму/эгоизму. С завидным упорством эти понятия относятся к гуманитарной сфере, хотя на самом деле, эмоции – это базовый, физиологический уровень регуляции поведения животных, причем не только высокоорганизованных. А проявления альтруизма/эгоизма – это, во-первых, прямое проявление т.н. родственного отбора в природе. А во-вторых, эти механизмы могут быть описаны явными моделями, основу которых составляет процедура сопоставления конспектива с образом «Я» самого агента (механизм распознавания свой-чужой). Это же касается и такого феномена, как социальное обучение, который хорошо изучен нейробиологами и этологами. Этотстораживающий термин – социальное обучение – всего-навсего представляет собой обучение на примере других, причем это обучение заключается в разовом образовании устойчивых рефлекторных связей.

Ко второй категории можно отнести неадекватность масштабов предлагаемых моделей. С одной стороны, в рассуждениях ряда авторов о создании групповых систем, аналогичных «роям» насекомых, встречаются слова про сотни и даже тысячи искусственных агентов (при этом часто упоминаются популяции муравьев в десятки и сотни тысяч особей), см., например, [13]. Либо, напротив, промоделировав или даже создав пару-тройку устройств, говорится о естественном и очевидном масштабировании этой группы. На самом же деле, если говорить об адекватном моделировании поведения тех же муравьев, то достаточно иметь 6-10 агентов, не более (сама семья вовсе не должна насчитывать огромное количество особей, для функционирования хватает нескольких сотен, и реально активных индивидов в таком клане – порядка 10). Меньшее же количество агентов приведет к тому, что просто не удастся создать условия для проявления эффектов их взаимодействия. Интересные оценки параметров моделирования приведены в работе А.А. Малышева и Е.В. Бургова [14].

Биология без кибернетики

Рассуждать о сложностях совершенно далекой предметной области – дело неблагодарное, однако здесь можно встать на позицию «заказчика», пытающегося получить свой продукт. Для робототехника это означает обращение к коллегам-биологам (этологам) с просьбой вида «расскажите, как это устроено и как это работает у вас, в природе». На деле же попытки понять модели поведения до такой степени конкретизации, которая позволила бы реализовать их в виде программно-аппаратного комплекса, обычно заканчиваются неудачей. Проблема в том, что многократно и хорошо изученное поведение описывается с внешней, феноменологической точки зрения. Утрируя, можно сказать, что речь идет об описании множества очень интересных и познавательных «примеров из жизни». Безо всякого обобщения, формального конструктивного описания, пригодного для дальнейшего использования. При этом «понимание» алгоритма поведения не становится легче, когда рассматривается поведение «простых» (с точки зрения робототехника) животных – насекомых. Если у высокоорганизованных животных еще можно отследить стимул-реактивные связи, исследовать мозг и т.п., то с насекомыми эти технологии уже не работают. Разумеется, можно предположить, что это – очевидная специфика естественных наук (просто по определению), но проблема остается, причем в основе ее – совершенно разный понятийный аппарат.

Приведем частный, но показательный пример. В мирмекологии, разделе энтомологии, в котором изучают муравьев, считается хорошо изученным процесс фуражировки. В рамках фуражировки имеется частная задача – возврат в гнездо муравья, нашедшего корм, с последующей мобилизацией рабочих муравьев на доставку найденного ресурса. При этом говорится о том, что муравей-разведчик может повести остальных за собой, либо, и это принципиально, сообщить информацию о местоположении этой пиццы, см., например, работу Ж.И. Резниковой [15]. Вместе с тем, если опираться на описание принципов ориентации тех же муравьев (см, например [16]), то окажется, что основными понятиями при запоминании маршрута являются такие сущности, как свето-компасная ориентация, определение опорного маршрутного ориентира и время. Эти вопросы были исследованы в работе И.П. Карповой [17]. Но каким бы ни был механизм запоминания маршрута и «формат» описания, его передача другой особи подразумевает оперирование знаками-символами, определяющими эти сущности. А это автоматически влечет вывод о наличии у насекомых языка, причем не в переносном, метафорическом, а в прямом смысле (язык как носитель семиотической системы). Вместе с тем, ряд этоло-

гов приводят соображения в пользу того, что языка животных не существует, а имеется лишь сигнальная коммуникация. Это убедительно (для «технаря») показано, например, в работах Е.Н. Панова [18].

Инженерия без «кибернетики» и «биологии»

Существует огромный пласт проектов сугубо «инженерного», «безыдейного» характера, в названии которых присутствуют «кибермуравьи», «роботы-насекомые», «роботы-рыбы» и т.д. Останавливаться на их обзоре нет никакого смысла. Здесь лишь отметим, что подобного рода дилетантская (в смысле отсутствия биологических оснований) деятельность в значительной мере была обусловлена работами Родни Брукса. В своей «архитектуре приоритетных взаимодействий» Брукс, апеллируя к миру насекомых, предложил свести к минимуму процесс мышления, предшествующий действию [19]. Другим вдохновителем такого подхода является так называемая ВЕАМ-робототехника, созданная своего рода «экстремистом» от поведенческой робототехники М. Гилденом. Поведение ВЕАМ-роботов, предельно примитивных изделий, вполне адекватно, по мнению авторов, поведению насекомых, см., например, [20].

Спекуляции и жонглирование терминами

Использование звучных, привлекательных терминов – распространенное явление в техническом мире.

Например, специалистам в области многоагентных систем хорошо знаком термин «Искусственное сообщество». В книге Дж. Эпштейна и Р. Акстелла [21] “Growing Artificial Societies: Social Science From the Bottom Up” описывается некоторая агентная модель, развитие которой приводит, по мнению авторов, к появлению эффектов и явлений, наблюдаемых в человеческих социальных сообществах. При этом сами агенты крайне примитивны, а правила, определяющие их поведение, крайне просты. Проста и их окружающая среда, “Sugarscape”, – двумерная сетка, на которой произрастает поедаемый агентами «сахар». Более того, сами основатели этого направления говорят об «удивительной достаточности простых правил» для создания сложных систем. Однако, как будет сказано ниже, агент, способный к социальному взаимодействию, никак не может быть простым в силу сложности решаемых им задач. В противном случае мы имеем дело с игрой с терминами (если не спекуляциями). Сейчас «Искусственные сообщества» – это одно из направлений теории многоагентных систем, решающих различные задачи из области экономики, логистики, социологии и т.п., при этом ни мало не заботящееся о методических (этологических) обоснованиях. О направленности исследований в этой области можно судить, например, по содержанию различного рода специальных журналов и прочих ресурсов (см., например, [22] или [23]).

Другим ярким примером являются работы В. Брайтенберга с применением языка психологии для описания поведения мобильных роботов. Так, в [24] речь идет о некоторой гипотетической мобильной платформе (тележке) с очень простой внутренней структурой. В зависимости от того, как были соединены датчики и двигатели, тележка демонстрировала разнообразное поведение. При этом Брайтенберг описывал поведение тележки в терминах страха и агрессии, трусости и любви, любопытства и жажды исследования и т.д. Дело даже не в том, что здесь зачастую путаются понятия чувства и эмоции. Такого рода спекуляции рассматривают лишь внешнее проявление поведения, не выделяя его внутреннюю структуру, взаимосвязи и причинно-следственные отношения, и, как следствие, не позволяя обобщать поведенческие механизмы.

4. МОДЕЛИ СОЦИАЛЬНОГО ПОВЕДЕНИЯ В ГРУППОВОЙ РОБОТОТЕХНИКЕ

Редким примером направления исследований, в котором принципиально важным является тесное взаимодействие «кибернетики» и биологии, является разработка социума искусственных агентов, осуществляемая в рамках парадигмы моделей социального поведения (МСП) в групповой робототехнике. МСП рассматривает образование социума с точки зрения адаптивного механизма и сводит разнообразие решаемых группой агентов задач к обобщенной задаче поддержания территориального гомеостаза. При этом МСП выделяет очень ограниченный ряд механизмов, из которых складывается все поведенческое разнообразие. Не останавливаясь на сути разрабатываемых моделей и примерах практической реализации, отметим несколько характерных моментов.

Первым является принятое соглашение о том, что агент, способный к социальному взаимодействию, не может быть «простым» (как это декларировалось в групповой робототехнике). Основной модельный объект – муравей – вовсе не прост. Его «система управления» содержит порядка 1 млн. нейронов. Как следствие – сложные формы индивидуального и группового поведения.

Во-вторых, определение базовых механизмов, позволяющих создавать социумы агентов, оказалось весьма сложной задачей. Более того, даже вопрос о видах поведения, в том числе – социального – является до сих пор открытым. Так, с одной стороны психологи и биологи говорят о четырех основных формах поведения: пищевое, оборонительное, половое и исследовательское, см., например, [25]. С другой стороны, те же этологи или специалисты в области социальной биологии выделяют дополнительные, специфические формы и модели поведения, необходимые для орга-

низации социума, такие, как контагиозное (заразное), подражательное, агрессивное и пр. Многие из этих механизмов достаточно сильно пересекаются друг с другом, зачастую представляя собой лишь некоторое условное обозначение проявления тех или иных базовых механизмов. Некая обобщенная типология социального поведения (см., например, [26–29]) показана на Рис. 2.

В любом случае такая ситуация имеет вполне положительное следствие: из множества типологий, подходов и представлений «кибернетику» можно выбрать наиболее подходящее, удобное с «технической» точки зрения. Более того, сами этологи относятся с осторожностью к предлагаемым моделям и методам. Вопрос адекватности моделей биологических процессов и явлений – это большая проблема. Так, в [30] говорится о том, что вместо описания создаваемых моделей зачастую имеется предписание им желаемых свойств, не имеющих ничего общего с реальными биологическими прототипами.

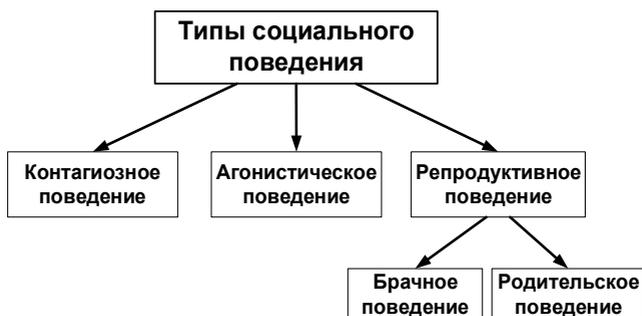


Рис. 2. Основные типы социального поведения

Одним из результатов исследований в области МСП является вывод, что базовыми механизмами, обеспечивающими существование социума, являются *воздействие, узнавание, сопоставление, «Я», модель мира, эмоции*. Они являются тем базисом, на котором основаны прочие феномены социального взаимодействия. В функциональном виде это представлено в табл. 1.

Это – перечень механизмов, необходимых для создания социума, компоненты для решения обобщенной задачи – поддержания территориального гомеостаза, к которой сводится ряд практических задач ГР. Звездочкой отмечены базовые механизмы, реализуемые «как есть», т.е. не рассматриваемые как отдельные модели.

Таблица 1. Механизмы для социального взаимодействия

Механизм		Описание (составные части)
Базовые механизмы		
1.	Воздействие*	Сенсоры, эффекторы: $\Psi_{inf} = \Psi_{inf}(S, A)$
2.	Узнавание	Сенсоры, модель мира $\Psi_{rec} = \Psi_{rec}(S, M) \rightarrow M_{img}$ $M_{img} \subseteq M$
3.	Сопоставление	Узнавание $\Psi_{comp} = \Psi_{comp}(M_{img}) = M_{img} \times M_{img} \rightarrow M_{name}$ $M_{img}, M_{name} \subseteq M$
4.	Субъективное Я	Модель мира $M_I = \Psi_I(M) = SignI \subseteq M$
5.	Модель мира	$M_{wmod} \subseteq M$
6.	Эмоции	Сенсоры, потребности, эффекторы $\Psi_{emo} = \Psi_{emo}(N, S, A, E) \rightarrow A \times E$
7.	Выбор лидера*	$\Psi_{leader} = \Psi_{leader}(W) \rightarrow A$
Производные механизмы		
8.	Движение*	Воздействие $\Psi_{mov} = \Psi_{mov}(\Psi_{inf})$
9.	Сигнальная коммуникация	Эмоции $\Psi_{sc} = \Psi_{sc}(E, \Psi_{inf})$
10.	Идентификация	Узнавание, сопоставление $\Psi_{ident} = \Psi_{ident}(\Psi_{rec}, \Psi_{comp})$
11.	Картина мира	Модель мира, Я $\Psi_{wp} = \Psi_{wpic}(\Psi_I, \Psi_{wmod})$
12.	Контагиозное поведение	Эмоции, сигнальная коммуникация $\Psi_{cont} = \Psi_{cont}(\Psi_{sc}, \Psi_{emo})$
13.	Агрессия	Идентификация, оказание воздействия, эмоции $\Psi_{aggr} = \Psi_{aggr}(\Psi_{inf}, \Psi_{ident}, \Psi_{emo}) \rightarrow A$
14.	Социальное обучение	Идентификация, сопоставление $\Psi_{ss} = \Psi_{ss}(\Psi_{ident}, M_I) \rightarrow A \times S$
15.	Подражательное поведение	$\Psi_{ib} = \Psi_{ib}(\Psi_{rec}, \Psi_{ident}, M_I) \rightarrow A \times S$
16.	Доминирование	Агрессия, идентификация, лидерство $\Psi_{dom} = \Psi_{ib}(\Psi_{aggr}, \Psi_{ident}, \Psi_{leader}) \rightarrow A$

Правда, следование этологическим (природным) принципам приводит иногда к появлению сугубо «технических» проблем, одной из которых является определение критериев качества функционирования социума искусственных агентов, его целевой функции. В конечном итоге все сводится к достижению единственной цели – выживанию особи. Согласно [31], основные задачи социума муравьев – это

самосохранение в годичном цикле и расширенное воспроизводство населения, обеспечивающее развитие социума в многолетней перспективе. И тогда критерием эффективности группы (социума) становится личное благополучие особи.

Есть и иные проблемы. Само понятие поведения (в том числе – социального) является некоторой условностью, и тогда возникает вопрос о том, имеет ли смысл вообще говорить о реализации тех или иных моделей поведения? Нет ли здесь опасности создания химеры из некоторого набора моделей и методов, реализующих отдельные поведенческие феномены и проявления социальности. Основная задача МСП заключается как раз в том, чтобы не строить эту химеру, а создать единый, целостный механизм поведения индивида и его взаимодействия с остальными членами группы. Используя при этом некий базовый, желателен – минимальный набор механизмов. При этом крайне привлекательно с «технической» точки зрения суждение, что сложность поведения индивида отражает сложность среды, а не обязательно сложность агента. Об этом говорит, например, Г. Саймон в [32] Таким образом, делается важный вывод: в основе природной эмерджентности лежат весьма примитивные механизмы. Природа лаконична, а наблюдаемая сложность поведения или функционирования – это, прежде всего, результат взаимодействия весьма ограниченного числа базовых механизмов, о которых говорилось выше.

Итак, вопрос не в том, насколько корректны предложенные методы, реализующие те или иные модели социального поведения и организации. Основной проблемный момент заключается в правомерности представления такого сложного объекта, как системы управления социального организма (агента, робота), в виде совокупности функциональных блоков. А феномен социального поведения – в виде множества пусть и взаимодействующих, но независимых механизмов.

5. ЗАКЛЮЧЕНИЕ

Разумеется, эта работа не ставит задачей объяснить, что и как надо делать. Ее задача – попытаться объяснить причины скромности результатов такого действительно многообещающего и потенциально мощного направления. Подытожив, мы имеем следующие основные проблемы этой междисциплинарной области:

1. Сложно договориться. Отсутствует единая система понятий, представлений, взглядов на суть изучаемых явлений.

2. Объективная техническая сложность решаемых задач и их большая стоимость, когда речь заходит о реализации технических устройств. Здесь зачастую требуется весьма специфическая элементная база, а сами эксперименты должны проводиться на больших группах технических устройств.

3. Реальная недостаточность знаний о принципах функционирования даже «простейших» представителей животного мира.

Разумеется, проблемы не новы. Более того, своего рода формой интеграции является т.н. биологическая кибернетика, но, подобно большому множеству прочих кибернетик, здесь дальше деклараций дело обычно не идет. Идеологическая разобщенность пока является препятствием непреодолимым.

Литература

1. Эшби У.Р. Введение в кибернетику. Изд.стереотип. – М.: URSS, 2017. – 430 с.
2. Гаазе-Рапопорт М.Г., Поспелов Д.А. От амебы до робота: модели поведения. – М.: Едиториал УРСС, 2004. – 296 с.
3. Цетлин М.Л. Исследования по теории автоматов и моделированию биологических систем. – М.: Наука, 1969. – 316 с.
4. Варшавский В.И. Коллективное поведение автоматов. – М.: Наука, 1973. – 408 с.
5. Стефанюк В.Л., Цетлин М.Л. О регулировке мощности в коллективе радиостанций // Проблемы передачи информации. – 1967. – Т. 3, № 4. – С. 59-67.
6. Стефанюк В.Л. Локальная организация интеллектуальных систем. – М.: Физматлит, 2004. – 328 с.
7. Wilson S.W. Knowledge Growth in an Artificial Animal // Adaptive and Learning Systems: Theory and Applications / Ed. K.S. Narendra. Boston, MA: Springer US, 1986. – P. 255-264.
8. Wilson S.W. Classifier Systems and the Animat Problem // Mach. Learn. – 1987. – Vol. 2, No. 3. – P. 199-228.
9. Бонгард М.М., Лосев И.С., Смирнов М.С. Проект модели организации поведения – «животное» // Моделирование обучения и поведения. – М.: Наука, 1975. – С. 152-171.
10. Редько В.Г. Модели автономных когнитивных агентов – бионический задел развития искусственного интеллекта // Труды XIII Национальной конференции по искусственному интеллекту с международным участием КИИ-2012 (16–20 окт. 2012, Белгород. Т. 4. – 2012. – С. 172-179.
11. Редько В.Г. Моделирование когнитивной эволюции. На пути к теории эволюционного происхождения мышления. – М.: УРСС, 2015. – 256 с.
12. Анохин К.В. и др. Проект «Мозг Анимата»: разработка модели адаптивного поведения на основе теории функциональных систем // Труды VIII Национальной конференции по искусственному интеллекту с международным участием. – М.: Физматлит, 2002. – С. 781-789.
13. Navarro I., Matia F. An Introduction to Swarm Robotics // ISRN Robot. Artic. ID 608164. – 2013. – Vol. 2013. – P. 10.
14. Малышев А.А., Бургов Е.В. К вопросу о параметрах биоинспирированных моделей поведения при моделировании групповой фуражировки // Труды СПИИРАН. – 2020. – Т. 19, № 1. – С. 79-103.

15. Резникова Ж.И. Интеллект и язык животных и человека. Основы когнитивной этологии. – М.: Академкнига, 2005. – 518 с.
16. Захаров А.А. Муравей. Семья. Колония. – М.: Фитон XXI, 2018. – 192 с.
17. Карпова И.П. Об одном бионспирированном подходе к ориентации роботов, или настоящий «муравьиный» алгоритм // Управление большими системами. – 2022. – Т. 96. – С. 69-117.
18. Панов Е.Н. Эволюция диалога. Коммуникации в развитии: от микро-аргонизмов до человека. – М.: Языки славянской культуры, 2014. – 400 с.
19. Brooks R.A. A robust layered control system for a mobile robot // IEEE J. Robot. Autom. – 1986. – Vol. 2. – P. 14-23.
20. Tilden M.W., Hasslacher B. The Design of “Living” Biomech Machines: How low can one go? // Physics Division, Los Alamos National Laboratory. – 1997. 505/667-2902.
21. Epstein J.M., Axtell R. Growing Artificial Societies: Social Science from the Bottom Up. – The MIT Press, 1996. – Iss. 1. – 228 p.
22. Journal of Artificial Societies and Social Simulation. – URL: <http://jasss.soc.surrey.ac.uk/JASSS.html> (дата обращения: 06.04.2019).
23. Mills A. Complexity Science: an introduction (and invitation) for actuaries. – 2010. – 220 с.
24. Braitenberg V. Vehicles: Experiments in synthetic psychology. – Cambridge, MA: MIT Press, 1984.
25. Лурия А.Р. Лекции по общей психологии. – СПб: Питер, 2007. – 320 с.
26. Гудолл Д. Шимпанзе в природе: поведение. – М.: Мир, 1992. – 670 с.
27. Зорина З.А., Полетаева И.И., Резникова Ж.И. Основы этологии и генетики поведения. – М.: Изд-во МГУ: Высшая школа, 2002. – 383 с.
28. Мак-Фарленд Д. Поведение животных: Психобиология, этология и эволюция. – М.: Мир, 1988. – 520 с.
29. Тинберген Н. Социальное поведение животных – Social Behavior in Animals, 1953 / пер. с англ. под ред. П.В. Симонова. – М.: Мир, 1993. – 81 с.
30. Богатырева О.А., Богатырев Н.Р. Муравьи и роботы в мире, полном неопределенности: природа адаптивности группы // Материалы XIV Всероссийского мирмекологического симпозиума «Муравьи и защита леса». Москва, 19–23 августа 2013 г. – М.: Товарищество научных изданий КМК, 2013. – С. 14-20.
31. Захаров А.А., Захаров Р.А. Влияние зоогенных и климатических факторов на годичный цикл жизни муравейника группы *Formica Rufa* // Материалы XIV Всероссийского мирмекологического симпозиума «Муравьи и защита леса», Москва, 19–23 августа 2013 г. – М.: Товарищество научных изданий КМК, 2013. – С. 210-215.
32. Simon H.A. The Sciences of the Artificial. – MIT Press, 1996. – 231 p.

ИНТЕГРИРОВАННЫЕ И ГИБРИДНЫЕ СИСТЕМЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА: МЕТОДОЛОГИЧЕСКИЕ ПРОБЛЕМЫ И ВОПРОСЫ ТЕРМИНОЛОГИИ

*Кобринский Б.А., д.м.н., профессор
ФИЦ «Информатика и управление» РАН,
Институт проблем искусственного интеллекта
e-mail: kba_05@mail.ru*

*Время разбрасывать камни, и время собирать камни –
всему свое время
Екклесиаст [1]*

*Основная задача состоит не в накоплении,
а в умении активизировать нужные знания
в процессе решения задач
Д.А. Поспелов [2]*

1. ВВЕДЕНИЕ

В интегрированной системе необходимые модули знаний должны активизироваться или подключаться по мере необходимости, как инструменты в хорошем ансамбле. Это аналогично процитированному в эпиграфе мнению Д.А. Поспелова [2] в отношении многоагентных систем о своевременном включении в работу необходимых для выполнения совместной работы знаний. То есть речь должна идти об объединении, о формировании общего искусственного интеллектуального поля.

Д.А. Поспелов [3] подчеркивал, что теоретические проблемы искусственного интеллекта (ИИ) возникают и исследуются на стыке философии, психологии, лингвистики, семиотики, логики, этики, а необходимым инструментом построения формальных моделей и прикладных интеллектуальных систем являются методы и средства прикладной математики (включая прикладную логику), теории систем, теории управления, информатика. Но последующее развитие ИИ привело к формированию многочисленных независимых направлений, к извлечению знаний из данных, к технологиям машинного обучения, в частности к обучению нейросетей глубокого обучения распознаванию ситуаций на основе примеров, предполагающих использование

Big Data. Однако плодотворность новых подходов имеет негативный подтекст разобщения, противоречащий созданию высокоинтеллектуальных продуктивных систем.

В.Б. Тарасов во введении к монографии «От многоагентных систем к интеллектуальным организациям» [4], комментируя теорию концептуальных разломов из фантастического рассказа Д.А. Пospelova «Соприкосновение» [5], отмечал, что она «напоминает историю долгого периода дифференциации наук, когда различные научные дисциплины развивались независимо, словно параллельные миры, лишь изредка соприкасаясь друг с другом, а отдельные ученые, получая все более узкую специализацию, мало что знали о достижениях даже своих «близких собратьев». Также В.Б. Тарасов [4] обращает внимание на то, что «вновь, как и на заре ИИ, актуальными становятся формирование единых методологических основ ИИ, разработках теоретических проблем создания интеллектуальных систем новых поколений, ... большие перспективы связаны с использованием идей и принципов синергетики». По мнению Г. Хакена [6], предложившего термин «синергетика», это отвечает понятию «сложных самоорганизующихся систем по двум причинам»: а) исследуются совместные действия многих элементов развивающейся системы; б) осуществляется поиск общих принципов самоорганизации, требуется объединение усилий представителей различных дисциплин.

Преодоление концептуального разлома между различными направлениями исследований в области искусственного интеллекта – это, своего рода «прыжок» через «концептуальную пропасть», совершить который можно по разным направлениям или путем постройки ряда мостовых переходов. Это обусловлено не только методологическими, но и терминологическими причинами, привычкой и нежеланием отказа от привычных наименований отдельных направлений ИИ, разветвление или дезинтеграция которых все более прослеживается в множатся топиках конференций (синергетические модели самоорганизации, интеллектуальные компьютерные системы, гибридные интеллектуальные компьютерные системы, коллаборативные интеллектуальные компьютерные системы, мультисенсорная конвергенция и многие другие). Широкое применение получил принцип «мягких вычислений» (*Soft Computing*) для создания гибридных систем, основанных на нечетких, нейросетевых, эволюционных технологиях, которые позволяли частично компенсировать ограничения, присущие каждой из моделей в отдельности. То есть, с одной стороны используются разные термины, но «во глубине» сохраняется представленная в разных формах идея интеграции. В первую очередь, это синергетика, интеграция, конвергенция, гибридизация. И пришло время формирования

единой или общей искусственно-интеллектуальной платформы. Гибридные интеллектуальные системы позволяют использовать преимущества составляющих моделей и одновременно смягчать ограничения каждого из используемых подходов [7]. Взаимодействие в решении задач может быть обеспечено при интеграции и взаимозависимости (*mutuality*) альтернативных техник [8]. Фактически речь идет о фундаментальной проблеме синергетического искусственного интеллекта, связанной с интеграцией различных форм знаний в интересах взаимокompенсации недостатков и объединения преимуществ разнородных моделей для преодоления комбинаторной сложности задач [9]. Применяемые способы гибридизации интеллектуальных моделей можно свести в три группы [10–13]: а) функциональное замещение, при котором отдельные компоненты доминирующей модели замещаются компонентами других моделей; б) взаимодействие, предполагающее относительно независимое функционирование моделей, выполняющих различные задачи и обменивающихся информацией; в) полиморфная гибридизация, обеспечивающая имитацию и подмену функционирования другой модели. Метод построения композиционных гибридных нечетких моделей ориентирован на выполнение комплексных задач анализа систем и процессов, для которых возможна декомпозиция в соответствии с этапами или стадиями выполняемых процессов [14].

В настоящее время разработано большое количество моделей представления и обработки знаний различного вида, архитектур для построения интеллектуальных систем, методов и технологий решения прикладных задач. Многообразие видов статичных и динамичных интеллектуальных систем и, соответственно, многообразие используемых ими комбинаций моделей представления знаний определяется спецификой предметных и проблемных областей, разнотипностью хранимых знаний, многоликостью моделей обработки знаний и разнообразием методов решения задач. Фактически в связи с этим дорожная карта научных исследований США по ИИ в качестве ключевых проблем выделяет такие направления, как связность систем ИИ (*Integrated Intelligence*) и их осмысленное взаимодействие (*Meaningful Interaction*) [15]. Хотя это не исключает, что в различных приложениях могут быть востребованы разные комбинации моделей, методов, технологий.

В то же время существует большой разрыв в интеграции систем, реализованных с использованием различных подходов, в особенности при обработке лингвистической и образной информации, хотя та и другая чаще всего является символической. Человек с древности использовал изображения (петроглифы) как символы для обмена информацией.

2. УНИФИКАЦИЯ, ТИПИЗАЦИЯ, ОНТОЛОГИИ

Унификация предполагает приведение к определенному единому образу, что обеспечивает переход к типовым решениям. Унификация процессов построения моделей интеллектуальных систем основана на применении однотипных схем конструирования описаний системы понятий. Применение унифицированного семейства базовых понятий и форматов их описания обеспечивает гибкость процессов разработки моделей, получаемых из начальной абстрактной модели [16]. Знания, представляющие описания понятий, относящихся к процессам конструирования моделей интеллектуальных систем, составляют онтологию их проектирования. В то же время способы унификации онтологических моделей позволяют учесть разнородность обрабатываемых знаний и многообразие механизмов их использования [17]. Следует вспомнить определение Р. Мизогучи [18], указывающее, что онтология снабжает нас некоторым множеством понятий (словарем), используемым как предикаты, которые нам нужны для описания мирового знания об окружающем мире, определяет смысл и ограничения, присущие каждому понятию. Основной задачей онтологии является выполнение роли основы для понимания и коммуникаций всех участников (и людей, и программ) моделируемых производственных или других процессов предметной области [19]. Таким образом, онтология может рассматриваться как базовая характеристика в проблеме построения интегрированных систем. Это позволяет высказать предположение о возможности унифицировать систему используемых понятий, специфицируемых соответствующими онтологиями, для обеспечения семантической совместимости и интероперабельности интеллектуальных систем, построенных с использованием различных подходов.

В качестве принципов организации совместимости моделей представления и обработки знаний в интеллектуальных системах следует рассматривать совокупность подходов – синтаксический, семантический (согласованность систем понятий) и функциональный (операционный) [20]. Продолжая развитие этих представлений, В.В. Голенков и соавторы [21] предлагают стандарт внутренней семантической репрезентации знаний в памяти интеллектуальной системы, который называют SC-кодом (*Semantic Code*). При этом интеграция различных типов знаний предполагается на основе гибридных моделей баз знаний, включающих в себя иерархический набор онтологий верхнего уровня, которые должны обеспечивать семантическую совместимость различных видов знаний и позволять интегрировать факты, различные характеристики объектов, логические утверждения, ситуации.

3. ИНТЕГРАЦИЯ МЕТОДОВ И ТЕХНОЛОГИЙ В ИСКУССТВЕННОМ ИНТЕЛЛЕКТЕ

При разработке и реализации различных интеллектуальных компьютерных систем соответствующие методы и средства должны гарантировать логико-семантическую совместимость разрабатываемых компонентов и, в частности, их способность использовать общие информационные ресурсы. Для этого, очевидно, необходима унификация указанных моделей.

Особые сложности представляет интеграция систем на знаниях, извлеченных у экспертов или из литературных источников, и систем, извлекающих знания из данных. Рассматриваются два возможных направления – коммуникация через входы и выходы искусственных нейронных сетей (ИНС) или с использованием онтологических структур и их интерпретации в базе знаний в рамках многоагентной модели на основе ИНС и знаний других видов при взаимодействии агентов через общую память [22]. Онтологическая модель как система понятий для описания различных уровней предметной области предложена в технологии, именуемой OSTIS [23, 24]. Однако остаются вопросы в отношении возможности и эффективности функционирования таких интегрированных систем при обработке изображений различной сложности. Хотя для этого было предложено понятие нейроэкспертных (нейрологических) гибридных интеллектуальных систем, основанных на синергетическом взаимодействии образных и символично-логических механизмов принятия решения [25, 26]. Однако практической реализации и этот подход до настоящего времени не получил.

Другой аспект проблемы интеграции пытается преодолеть K.R. Thórisson [27], предлагая реализацию общего открытого API, который, по его мнению, может помочь в преодолении барьеров операционных систем и языков программирования при интеграции систем ИИ. Одновременно он рассматривает методологию проектирования, основанную на модульности для поддержки строительства больших архитектур. «Двухполушарная» экспертная система (ЭС) [28] базируется на комбинации традиционной экспертной системы, осуществляющей логическую обработку, и искусственной нейронной сети, реализующей ассоциативную обработку. Взаимодействие между этими двумя компонентами осуществляется посредством простых фактов или гипотез, которые имеют смысл наличия некоторых событий, свойств или отношений между сущностями. Для этого служит механизм «доски объявлений» (black board). Функционируя относительно независимо, ЭС и ИНС ориентированы на параллельную асинхронную обработку фактов и гипотез при множественности линий рассуждений. При этом, по мнению авторов, появляется возможность предварительной обработки (классификации или кластери-

зации) сигнальной информации или данных, поступающих на систему, и преобразование их в факты или порождение гипотез на основе текущего набора фактов. Несомненно интересная реализация «двухполушарной» системы вызывает некоторые сомнения в отношении достаточного учета в сложных случаях контентных зависимостей и многообразных связей при формировании гипотезы на основе используемого обратного нечеткого логического вывода.

Движение в направлении конвергенции может опираться на принцип комбинированного приложения и слабой интеграции (composite application and poor integration) [29]. Это предполагает определенную степень избыточности при одновременном запасе гибкости системы. В то же время метабазы полипредметных областей могут обеспечить формирование общего поля или интегрированной базы знаний (БЗ). В конвергирующих системах обмен информацией между суббазами можно организовать через общую «память». Управляющая метабаза знаний позволит активизировать, по мере необходимости, соответствующие БЗ по отдельным предметным областям, характеризующиеся семантической интероперабельностью, что необходимо для решения задач на стыке наук. В этом случае возможен обмен отдельными модулями баз знаний [20, 30]. Следует отметить, что в данном контексте не идет речь о *Knowledge Management* как проблемы обработки информации в крупных корпорациях, чему посвящены многочисленные публикации, в частности в [31].

Говоря о синергетическом подходе в искусственном интеллекте, следует признать, что, несмотря на различия, это понятие объединяет методы интеграции, конвергенции и гибридизации. Не останавливаясь на деталях, можно говорить о принципах обмена информацией с тем или иным взаимопроникновением отдельных методов и/или замещении отдельных компонент.

Интегрированность или гибридность интеллектуальных систем должна предусматривать сочетание как различных методов и технологий, так и разных типов знаний – лингвистических и иконических (образов или изображений различной природы).

4. ИНТЕРОПЕРАБЕЛЬНОСТЬ И ОБЪЯСНИМОСТЬ

Функциональная совместимость или интероперабельность (способность к взаимодействию при условии семантической совместимости) систем ИИ должна послужить основой для формирования осмысленного объяснения для пользователя при подключении модулей, использующих технологии машинного обучения. Понимание, как отмечает А.С. Есенин-Вольпин [32], требует тщательного анализа слов и их употреблений. В связи с этим нам следует стремиться к рассуждаю-

щим объяснениям предлагаемых ИИ гипотез или решений. Именно поэтому существенной особенностью информационных систем, по мнению Ю.М. Арского и В.К. Финна [33], является реализация в рассуждателе когнитивных правдоподобных эмпирических рассуждений.

К примеру, интерпретация в ИНС не решает вопроса объяснимости, хотя и является крайне полезным условием для повышения обучаемости нейросети. В то время как гибридизация систем на знаниях с ИНС может позволить решить проблему содержательного понимания предлагаемых решений, вызывающую постоянные вопросы пользователей. Распознавание ситуации с помощью ИНС, при анализе изображений, может дополняться контентом и подтверждаться соответствующими фактами с выдачей необходимых объяснений на основе правил базы знаний. Такой подход обеспечит эмерджентную эволюцию интегрированных систем.

5. ИНТЕГРАЦИЯ ИНТЕЛЛЕКТУАЛЬНЫХ И ИНФОРМАЦИОННЫХ СИСТЕМ

Интеграционные процессы в рамках систем искусственного интеллекта открывают новые возможности для интерактивного интеллектуального управления информационными системами. Развивается направление по интеграции систем управления знаниями и управления бизнес-процессами (*integrating knowledge management and business process management*) [34]. Конвергенция в среде распределенных баз знаний, предполагающая их совместное использование, должна послужить основой для повышения эффективности управления информационными системами [35]. Этот процесс может быть реализован двумя путями.

Возможно погружение систем поддержки принятия решений в информационные системы [36], примером чего может служить российский федеральный генетический регистр, включающий интеллектуальные модули определения типа наследования болезни и выбора методов лабораторных исследований для подтверждения диагностической гипотезы [37].

Внешнее «управление» предполагает анализ содержания баз данных информационных систем, извлечение и представление необходимой пользователям информации. Управляющая метабаза знаний через специальные модули или суббазы знаний, связанные с базами данных соответствующих предметных областей позволит, на основе формируемых гипотез, обеспечить выявление необходимых фактов или подбор необходимых данных, включая изображения [38].

Сегодня такой подход уже востребован умными больницами и умными городами, хотя и не получил соответствующей полноценной реализации.

6. ЗАКЛЮЧЕНИЕ

Интеграция систем ИИ, как бы мы не называли этот процесс (синергия, конвергенция, гибридизация) является необходимостью и вызовом времени. В практическом плане это требует создания систем, отвечающих понятию гибридности как результата конвергенции, соединения многообразных подходов, методов и технологий, обрабатывающих различные виды знаний и данных. Семантическая же совместимость остается необходимым условием интероперабельности систем.

В качестве перспективы мы должны рассматривать возможные способы конвергенции, сближения, унификации подходов и построения на этих основах фундаментальной (но модифицирующей в соответствии с новыми технологиями) общей теории систем ИИ, обладающих способностью разумно интерпретировать предлагаемые гипотезы и решения в различных предметных областях.

Литература

1. Екклесиаст Книга Екклесиаста или проповедника. Глава 3:5. – М.: Присцельс, 1996. – С. 24.
2. Поспелов Д.А. Многоагентные системы – настоящее и будущее // Информационные технологии и вычислительные системы. – 1998. – № 1. – С. 14-21.
3. Поспелов Д.А. Системы искусственного интеллекта. Итоги и ближайшее будущее // Problems of Control and Information Theory. – 1980. – Vol. 9, No. 1. – P. 3-18.
4. Тарасов В.Б. От многоагентных систем к интеллектуальным организациям. – М: Эдиториал УРСС, 2002.
5. Поспелов Д. Соприкосновение // Знак Водолея. – М.: Фантом Пресс интер В.М., 1997. – С. 181-192.
6. Хакен Г. Синергетика. Иерархии неустойчивостей в самоорганизующихся системах и устройствах. – М.: Мир, 1985.
7. Seera M., Lim Ch.P. A hybrid intelligent system for medical data classification // Expert Systems with Applications: An International Journal. – 2014. – Vol. 41, Iss. 5. – P. 2239-2249.
8. Jacobsen H. A generic architecture for hybrid intelligent systems // 1998 IEEE International Conference on Fuzzy Systems Proceedings. IEEE World Congress on Computational Intelligence (Anchorage, AK, USA, 4-9 May 1998). Vol. 1. Piscataway, N.J.: IEEE, 1998. – P. 709-714.
9. Колесников А.В., Кириков И.А., Листопад С.В. Гибридные интеллектуальные системы с самоорганизацией: координация, согласованность, спор. – М.: ИПИ РАН, 2014.
10. Аверкин А.Н., Прокопчина С.В. Мягкие вычисления и измерения // Интеллектуальные системы. – 1997. – Т. 2. – Вып. 1-4. – С. 93-113.
11. Tarassov V.B. General Approaches to the Modelling of Soft Estimates and Beliefs in Strategic Decision Engineering // Proceedings of 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS 2002, Divnomorskoe, Russia, 5-10 September). – Los Alamitos, CA: IEEE, 2002. – P. 45-49.

12. Батыршин И.З., Недосекин А.О., Стецко А.А., Тарасов В.Б., Язенин А.В., Ярушкина Н.Г. Нечеткие гибридные системы. Теория и практика / под ред. Н.Г. Ярушкиной. – М.: Физматлит, 2007.
13. Тарасов В.Б. От гибридных систем к интеллектуальным средам // Гибридные и синергетические интеллектуальные системы: теория и практика: Материалы I Международного симпозиума (ГИСИС'2012, Калининград, Светлогорск, 29 июня – 2 июля 2012 г.). Ч. 1. – Калининград: БФУ им. И. Канта, 2012. – С. 42-54.
14. Борисов В.В. Систематизация нечетких и гибридных нечетких моделей // Мягкие измерения и вычисления. – 2020. – Т. 29, № 4. – С. 100-123.
15. Gill Y., Selman B. A 20-year Community Roadmap for Artificial Intelligence Research in the US / Computing Community Consortium (CCC) and Association for the Advancement of Artificial Intelligence (AAAI) // arHiv. – 2019. – URL: <https://arxiv.org/abs/1908.02624> (дата обращения: 12.03.2022).
16. Костенко К.И., Левицкий Б.Е. Онтология иерархического проектирования интеллектуальных систем // Онтология проектирования. – 2022. – Т. 12, № 1 (43). – С. 68-81.
17. Смирнов А.В., Пашкин М.П., Шилов Н.Г. и др. Онтологии в системах искусственного интеллекта: способы построения и организации. Ч. 1 // Новости искусственного интеллекта. – 2002. – № 1. – С. 3-13.
18. Мизогучи Р. Шаг в направлении инженерии онтологий // Новости искусственного интеллекта. – 2000. – № 1-2. – С. 11-36.
19. Гаврилова Т., Кудрявцев Д., Муромцев Д. Инженерия знаний. Модели и методы. – СПб: Лань, 2016.
20. Golenkov V., Shunkevich D., Davydenko I., Grakova N. Principles of organization and automation of the semantic computer systems development // Open Semantic Technologies for Intelligent Systems. Iss.3. – Minsk: BSUIR, 2019. – P. 53-90.
21. Golenkov V., Guliakina N., Golovko V., Krasnoproshein V. Artificial intelligence standardization is a key challenge for the technologies of the future // Communications in Computer and Information Science. Vol. 1282: Open Semantic Technologies for Intelligent System (OSTIS'2020). – Cham: Springer, 2020. – P. 34-47.
22. Головкин В.А., Голенков В.В., Ивашенко В.П. и др. Интеграция искусственных нейронных сетей с базами знаний // Онтология проектирования. – 2018. – Т. 8. № 3 (29). – С. 366-386.
23. Голенков В.В., Гулякина Н.А. Проект открытой семантической технологии компонентного проектирования интеллектуальных систем. Ч. 1: Принципы создания // Онтология проектирования. – 2014. – № 1 (11). – С. 42-64.
24. Голенков В.В., Гулякина Н.А. Проект открытой семантической технологии компонентного проектирования. Ч. 2: Унифицированные модели проектирования // Онтология проектирования. – 2014. – № 4 (14). – С. 34-53.
25. Голицын Г.А., Фоминых И.Б. Нейронные сети и экспертные системы: перспективы интеграции // Новости искусственного интеллекта. – 1996. – № 4. – С. 121-145.
26. Фоминых И.Б. Гибридные интеллектуальные системы // Мягкие измерения и вычисления. – 2018. – № 12 (13). – С. 34-41.

27. Thórisson K.R. Integrated A.I. systems // *Minds and Machines*. – 2007. – Vol. 17, No. 1. – P. 11-25.
28. Гаврилов А.В. Гибридные интеллектуальные системы. – Новосибирск: НГТУ, 2002.
29. Mocanu M., Muste M., Lungu V., Drobot R. Composite application for water resource management // *Advances in Intelligent Control Systems and Computer Science* / L. Dumitrache, ed. Berlin, Heidelberg: Springer-Verlag, 2013. – P. 295-306.
30. Kobrinskii B.A., Yankovskaya A.E. Convergence of Applied Intelligent Systems with Cognitive Component // *Communications in Computer and Information Science*. Vol. 1282: *Open Semantic Technologies for Intelligent System (OSTIS'2020)*. – Cham: Springer, 2020. – P. 34-47.
31. Barad M. Applying Quality Function Deployment to Build a Knowledge Management Conceptual Framework // *Proceedings of the 15th International Conference on Intellectual Capital, Knowledge Management & Organisational Learning ICICKM 2018*. (University of the Western Cape South Africa, 29–30 November 2018) / Sh. Pather, ed. Reading: Publ. Academic Conferences and Publ. International Ltd, 2018. – P.1-6.
32. Есенин-Вольпин А.С. Свободный философский трактат // *Философия. Логика. Поэзия. Защита прав человека: Избранное*. – М.: РГТУ 1999. – С. 31-52.
33. Арский Ю.М., Финн В.К. Принципы конструирования интеллектуальных систем // *Информационные технологии и вычислительные системы*. – 2008. – Вып. 4. – С. 4-37.
34. Jung J., Choi I., Song M. An integration architecture for knowledge management systems and business process management systems // *Computers in Industry*. – 2007. – Vol. 58, No. 1. – P.21-34.
35. Кобринский Б.А. Формирование медицинских информационных систем, управляемых знаниями // *Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2016): материалы VI Международной научно-технической конференции (Минск, 18–20 февраля 2016 г.)* / В.В. Голенков (отв. ред.). – Минск: БГУИР, 2016. – С. 89-92.
36. Velickovski F., Seccaroni L., Roca J., Burgos F., Galdiz J.B., Marina N., Lluch-Ariet M. Clinical Decision Support Systems (CDSS) for preventive management of COPD patients // *J Translational Medicine*. – 2014. – Vol. 12, Suppl 2, S9.
37. Kobrinsky B., Tester I., Demikova N., Sedov Yu., Marjanchik B., Taperova L., Glukhovskaya Yu., Podolnaja M. A Multifunctional System of the National Genetic Register // *Medinfo'98: Proc.9th International Congress on Medical Informatics*. Pt 1 (Seoul, South Korea, August 18-22, 1998). – Seoul. – P. 121-125.
38. Kobrinskii B., Donitova V. Metaphorical images and knowledge in art and medicine // *Knowledge in Digital Age: 15th International Forum on Knowledge Asset Dynamics (IFKAD 2020): Proceedings, Italy, Matera, 9–11 September 2020*. Matera: Institute of Knowledge Asset Management (IKAM). – 2020. – P. 1 56-167.

ОТ НЕЧЕТКОЙ К ВОЗМОЖНОСТНО-ВЕРОЯТНОСТНОЙ ОПТИМИЗАЦИИ*

*Язенин А.В., д.ф.-м.н., профессор
Тверской государственной университет*

e-mail: yazenin.av@tversu.ru

*Егорова Ю.Е., старший преподаватель
Тверской государственной университет*

e-mail: j.e.egorova@gmail.com

*Солдатенко И.С., к.ф.-м.н., доцент
Тверской государственной университет*

e-mail: soldis@tversu.ru

Говорят, что «большое видится издалека». Однако это высказывание не применимо к творческому и научному наследию профессора Лотфи Заде. Уже сейчас с уверенностью можно сказать, что оно является не просто большим, а громадным. Его величие и масштабность, влияние на развитие современной науки, особенно теории искусственного интеллекта и методов принятия решений в условиях неопределенности и связанных с ними информационных технологий, присутствует и ощущается с середины прошлого столетия и до настоящего времени. Профессор Лотфи Заде был факелоносцем, освещающим дорогу идущим вслед за ним. Его идеи, связанные с методами моделирования знаний с элементами неопределенности, привели к формированию новых научных направлений и созданию информационных технологий, основанных на мягких вычислениях. В данной работе представлена авторская концепция развития одного из них, очень важного для практики научного направления, каким является возможно-вероятностная оптимизация. У истоков его создания стояли такие титаны, как Лотфи Заде и Ричард Беллман [1].

В работе [1] авторами была предложена симметричная схема для принятия решений в условиях неопределенности нечеткого типа. Математический инструментарий для моделирования данного типа неопределенности ранее был заложен в работе профессора Лотфи Заде

* Работа выполнена при финансовой поддержке РФФИ, проект №20-01-00669.

[2]. Все это позволило в дальнейшем разработать и сформировать новое научное направление – нечеткая оптимизации. Классическими в этом контексте являются работы [3, 4] и других авторов.

Следующим шагом на пути к созданию возможно-вероятностной оптимизации являются работы [5, 6], в которых заложены основы современной теории возможностей. В [6] была осуществлена возможность интерпретация нечеткого подмножества. Однако математически обоснованный переход к интерпретации, основанный на возможностной мере, фактически был сделан С. Намиасом [5], хотя он и называл соответствующую функцию множества оценочной функцией. Им также были введены понятия нечеткой переменной и ее функции распределения, аналога случайной величины в вероятностной модели неопределенности, и тем самым заложены основы аксиоматизированного исчисления возможностей по схеме, аналогичной колмогоровской [7].

В работах [8, 9] и более ранних работах этого автора впервые в задачах линейного программирования с нечеткими параметрами было в полной мере использовано исчисление возможностей, это, собственно говоря, и есть возможность оптимизация, в то время как в релевантных работах, к примеру, в [10], и в других, по-прежнему, использовалось исчисление нечеткости, основанное на принципе обобщения Лотфи Заде. Однако эти задачи также классифицировались как задачи возможностного программирования. Подход Беллмана–Заде [1] может быть погружен в возможностный контекст и представлен как частный случай одной из рассматриваемых в докладе моделей возможно-вероятностной оптимизации.

Идеи Лотфи Заде, связанные с понятием вероятности нечеткого события, методов приближенных рассуждений и нечеткой логики [11–13] способствовали построению математической модели случайного эксперимента с нечетким исходом и теории нечеткой случайной переменной [5, 14–18]. Это позволило перейти к разработке моделей и методов оптимизации в условиях гибридной неопределенности возможно-вероятностного типа [19–23]. Современный математический аппарат для представления неопределенных знаний указанного выше типа, ориентированный на их использование в задачах оптимизации, систематизирован и развит в [24]. Архитектура моделей возможно-вероятностной и возможно-вероятностной оптимизации, классификация методов их решения [19, 23, 25–32] обусловлены способами агрегирования возможно-вероятностной информации [33–38], в том числе на основе как сильнейшей, так и слабейшей t -норм [24, 39].

Модели и методы возможно-вероятностной оптимизации находят применение в анализе портфельных инвестиций [40–42].

Литература

1. Bellman R.E., Zadeh L.A. Decision making in fuzzy environment // *Management Sci.* – 1970. – Vol. 17. – P. 141-162.
2. Zadeh L.A. Fuzzy sets // *Information and Control.* – 1965. – Vol. 8 (3). – P 338-353.
3. Zimmermann H.J. Description and optimization of fuzzy systems // *Int. J. of Gen. Systems.* – 1976. – Vol. 16. – P. 209-215.
4. Zimmermann H.J. Fuzzy programming and linear programming with several objective function // *Fuzzy Sets and Syst.* – 1978. – Vol. 1. – P. 45-55.
5. Nahmias S. Fuzzy variables // *Fuzzy Sets Syst.* – 1978. – Vol. 1. – P. 97-110.
6. Zadeh L.A. Fuzzy sets as a basis for a theory of possibility // *Fuzzy Sets and Syst.* – 1978. – Vol. 1. – P. 3-28.
7. Kolmogorov A.N. *Foundations of the Theory of Probability.* – Chelsea Publishing Company, New York, 1956.
8. Yazenin A.V. Fuzzy and stochastic programming // *Fuzzy sets and syst.* – 1987. – Vol. 22. – P. 171-180.
9. Yazenin A.V. On the problem of possibilistic optimization // *Fuzzy sets and syst.* – 1996. – Vol. 81. – P. 133-140.
10. Luhandjula M.K. On possibilistic linear programming // *Fuzzy Sets Syst.* – 1986. – Vol. 18. – P. 15-30.
11. Zadeh L.A. The concept of a linguistic variable and its application to approximate reasoning – i // *Information Sciences.* – 1975. – Vol. 8 (3). – P. 199-249.
12. Zadeh L.A. The concept of a linguistic variable and its application to approximate reasoning – ii // *Information Sciences.* – 1975. – Vol. 8 (4). – P. 301-357.
13. Zadeh L.A. The concept of a linguistic variable and its application to approximate reasoning – iii // *Information Sciences.* – 1975. – Vol. 9 (1). – P. 43-80.
14. Feng Y., Hu L., Shu H. The variance and covariance of fuzzy random variables and their applications // *Fuzzy Sets Syst.* – 2001. – Vol. 120. – P. 478-497.
15. Kwakernaak H. Fuzzy random variables I // *Inf. Sciences.* – 1978. – Vol. 15. – P. 1-29.
16. Kwakernaak H. Fuzzy random variables ii // *Inf. Sciences.* – 1979. – Vol. 17. – P. 253-278.
17. Nahmias S. Fuzzy variables in a random environment. In: M.M. Gupta, R.K. Ragade, R.R. Yager (eds.) // *Advances in fuzzy sets theory and applications*, NHCP, Amsterdam, 1979. – P. 165-180.
18. Puri M.L., Ralescu D.A. Fuzzy random variables // *J. Math. Anal. Appl.* – 1986. – Vol. 114. – P. 409-422.
19. Egorova Y.E., Yazenin A.V. The problem of possibilistic-probabilistic optimization // *J. Comput. Syst. Sci. Int.* – 2017. – Vol. 56 (4). – P. 642-667.
20. Luhandjula M.K. Optimisation under hybrid uncertainty // *Fuzzy Sets Syst.* – 2004. – Vol. 146. – P. 187-203.
21. Yazenin A.V. Linear programming with random fuzzy data // *Soviet J. Comput. Syst. Sci.* – 1992. – Vol. 30. – P. 86-93.
22. Yazenin A.V. On a method of solving a problem of linear programming with random fuzzy data // *J. Comput. Syst. Sci. Int.* – 1997. – Vol. 36 (5). – P. 737-741.

23. Yazenin A.V., Soldatenko I.S. On the problem of possibilistic-probabilistic optimization with constraints on possibility/probability // In: R.F. et al. (ed.) WILF 2018, LNAI 11291. – Springer, Switzerland, 2019. – P. 43-54.
24. Yazenin A.V. Basic concepts of possibility theory: a mathematical apparatus for decisionmaking under hybrid uncertainty condition. – Moscow: Fizmatlit Publ., 2016 (in Russian).
25. Antropov A. On solution methods for one task of possibility-necessity optimization // *Nechetkie Sistems i Myagkie Vychisleniya*. – 2010. – Vol. 5 (2). – P. 5-24.
26. Ermolev Y.M.: *Methods of Stochastic Programming*. – Moscow: Nauka, 1976 (in Russian)
27. Goldstein E.G., Judin D.V. *New trends in linear programming*. – Moscow: Sovetskoe radio, 1996.
28. Minoux M. *Mathematical programming: theory and algorithms*. – Wiley, Hoboken, 1986.
29. Soldatenko I., Yazenin A. Possibilistic optimization problems with mutually t-related parameters: Comparative study // *Journal of Computer and Systems Sciences International*. – 2008. – Vol. 47 (5). – P. 752-763.
30. Yazenin A.V. On the problem of maximization of the possibility to attain a fuzzy goal // *J. Comput. Syst. Sci. Int.* – 1999. – Vol. 38 (4). – P. 621-624.
31. Yazenin A.V., Egorova Y.E. On solution methods for tasks of possibilistic-probabilistic optimization. *Vestnik TvGU. Seriya: Prikladnaya matematika*. – 2013. – Vol. 4. – P. 85-103 (in Russian).
32. Yazenin A.V., Wagenknecht M. *Possibilistic Optimization*. Vol. 6/96. Brandenburgische Technische Universitat, Cottbus: Aktuelle Reihe, 1996.
33. Dubois D., Prade H. *Possibility Theory*. – Plenum Press, New York, 1988.
34. Hong D.H. Parameter estimations of mutually t-related fuzzy variables // *Fuzzy Sets Syst.* – 2001. – Vol. 123. – P. 63-71.
35. Khokhlov M.Y., Yazenin A.V. The calculation of numerical characteristics of fuzzy random data // *Vestnik TvGU. Seriya: Prikladnaya Matematika*. – 2003. – Vol. 2. – P. 39-43. (in Russian)
36. Mesiar R. A note to the t-sum of l-r fuzzy numbers. *Fuzzy Sets and Syst.* – 1996. – Vol. 2. – P. 259-261.
37. Mesiar R. Triangular norm-based additions of fuzzy intervals // *Fuzzy Sets and Syst.* – 1997. – Vol. 91. – P. 231-237.
38. Nguen T., Walker E.A. *A First Course in Fuzzy Logic*. – CRC Press, Boca Raton, 1997.
39. Fuller R., Keresztfalvi T. On generalization of nguyen's theorem // *Fuzzy Sets Syst.* – 1991. – Vol. 41 (3). – P. 371-374.
40. Egorova Y.E., Yazenin A.V. A method for minimum risk portfolio optimization under hybrid uncertainty // *J. Phys.: Conf. Ser. p.* – 2018. – Vol. 973. 012033.
41. Markowitz H.M.: *Portfolio selection* // *The Journal of Finance*. – 1952. – Vol. 7. – P. 77-91.
42. Yazenin A.V., Soldatenko I.S. A portfolio of minimum risk in a hybrid uncertainty of a possibilistic-probabilistic type: comparative study. In: J. Kacprzyk, E. Szmidt, S. Zadrozny, K.T. Atanassov, M. Krawczak (eds.) // *IWIFSGN/EUSFLAT-2017. AISC*. – Springer, Cham, 2018. – Vol. 643. – P. 551-563.

ИНТЕГРАЦИЯ МЕТОДОВ КОНСТРУИРОВАНИЯ ПРИЗНАКОВ И МЕТОДОВ ОБРАБОТКИ ЗНАНИЙ В ЗАДАЧАХ МАШИННОГО ОБУЧЕНИЯ

*Ярушкина Н.Г., д.т.н., профессор, ректор
Ульяновский государственный технический университет
e-mail: jng@ulstu.ru*

Целью представляемого исследования является поиск эффективной интеграции моделей, методов конструирования признаков машинного обучения, методов оценки наборов данных, методов формирования предметных рекомендательных систем, методов интеллектуального анализа динамики процессов в задачах предиктивной аналитики с учетом требований доверительного интеллекта при проектировании отраслевых решений. Для целей предиктивной аналитики необходимы новые методы интеллектуального анализа данных динамики показателей, методы конструирования признаков машинного обучения, методы оценки наборов данных, методы формирования предметных рекомендательных систем.

В настоящее время анализ данных динамики данных (показателей деятельности предприятия в том числе) позволяет решать широкий круг задач, связанных с определением состояния, с возможностью получения динамики изменения состояния объекта и/или процесса, а также определить аномалии в поведении. В последние 10 лет в промышленном производстве активно используются различные методы анализа данных для повышения управляемости производственными процессами, а также для повышения производительности труда и объемов производства. Эффективный подход к анализу данных позволяет предприятию повысить оперативность принятия решений, а в некоторых случаях использовать предиктивные методы аналитики. Активное внедрение современных средств производства, оснащенных разнообразными датчиками, способствовало значительному повышению объемов данных, которые необходимо проанализировать в процессе принятия решений. Роль анализа данных в производстве быстро увеличивается. Этому способствуют различные средства анализа данных, основанные на принципах автоматического машинного обучения (*AutoML*) и автоматического конструирования признаков (*Deep Feature Synthesis, Featuretools*), что снижает требования к квалификации спе-

циалиста по анализу данных. В настоящее время предприятия стремятся максимально использовать потенциал методов анализа данных и машинного обучения для повышения качества производственных процессов. Но в то же время отсутствуют методы и средства формирования контекста анализа данных динамики показателей, извлеченных из различных источников, с учетом высокой степени неопределенности в описании объектов и процессов предметной области, а также при постановке задачи анализа. Для эффективной предиктивной аналитики данных, извлеченных из различных источников, необходимо решение фундаментальных проблем учета нечеткости и фрагментарности данных, описывающих объекты и процессы некоторой предметной области, отсутствия переносимых в рамках различных предметных областей механизмов доверительного и интерпретируемого искусственного интеллекта, методов автоматического конструирования признаков для данных с учетом высокой степени неопределенности и ограничений предметной области.

Формирование множества обучающих данных имеет принципиально важное значение для успешного решения задач машинного обучения. Часто задачи машинного обучения сводятся именно к правильному формированию обучающего множества. Ошибки в формировании обучающего множества обычно оказываются критичными и способны свести на нет эффективность самих алгоритмов обучения. Среди специалистов по машинному обучению общепризнанным считается, что наличие хороших обучающих данных гораздо важнее качества алгоритма обучения. В связи с активным развитием глубоких нейронных сетей в последнее десятилетие вопросы формирования множества обучающих данных принимают особенно важное значение, поскольку во многих задачах глубокие нейронные сети демонстрируют качество, существенно превосходящее остальные алгоритмы машинного обучения, однако, чтобы получить подобный выигрыш в качестве, необходимо использовать обучающее множество очень большого размера, а также специальные методы расширения и имитации расширения обучающего множества. В то же время, в современной литературе по машинному обучению вопросам формирования обучающего множества уделяется недостаточное внимание, зачастую данные вопросы полностью игнорируются, недостаточно развита теоретическая база, объясняющая явления, возникающие в процессе формирования множества обучающих данных.

СЕКЦИЯ 1

**НЕЧЁТКИЕ МОДЕЛИ,
МЯГКИЕ ВЫЧИСЛЕНИЯ,
ИЗМЕРЕНИЯ И ОЦЕНКИ**

УДК 004.8

**ИНТЕРПРЕТИРУЕМАЯ МОДЕЛЬ
КЛАССИФИКАТОРА НА ОСНОВЕ НЕЧЕТКОЙ
ЛОГИСТИЧЕСКОЙ РЕГРЕССИИ**

*Долгий А.И., к.т.н., генеральный директор
АО «НИИАС»*

*Колоденкова А.Е., д.т.н., заведующий кафедрой
ФГБОУ ВО «СамГТУ»
e-mail: anna82_42@mail.ru*

*Ковалев С.М., д.т.н., профессор
Ростовский филиал АО «НИИАС»
e-mail: ksm@rfniias.ru*

1. ВВЕДЕНИЕ

Теория комбинирования свидетельств Демпстера–Шеффера (DS) [1, 2] широко используется в качестве формализма для обработки неопределенностей [3]. В последние годы данная методология все чаще находит применение в задачах распознавания образов и классификации. Одним из направлений современных исследований является объединение классификаторов, когда результаты классификаторов выражаются в виде вероятностных функций с последующим комбинированием результатов на основе различных правил слияния данных [4, 5]. Другой подход – доказательная классификация, которая преобразует решения отдельных статистических классификаторов в функ-

ции доверия или убеждения, а затем объединяет их по правилу Демпстера–Шеффера (*DS*-правилу) [6]. Полученная объединенная функция вероятностной массы, называемая ортогональной суммой, используется для принятия решений об отнесения объекта к тому или иному классу [7]. Благодаря общности и выразительности формализма функций доверия модели доказательных классификаторов обеспечивают более информативные результаты, чем традиционные классификаторы. Эта выразительность может быть использована, в частности, для количественной оценки неопределенности, обнаружение аномалий и сюрпризов в данных при слиянии информации.

За прошедшие годы было разработано несколько принципов проектирования доказательных классификаторов. В [8] было проведено различие между, так называемым модельным подходом, основанным на обобщенной теореме Байеса [9], и дистанционным подходе, при котором массовые функции строятся на основе расстояний до обучающих экземпляров или прототипов. Доказательные классификаторы в последних модификациях используются в широком спектре приложений [10, 11]. Эти модификации включают модели на основе правил k -ближайшего соседа [12], модификации правил k -ближайшего соседа [13]), а также доказательные нейросетевые классификаторы [14], в которых массовые функции строятся на основе расстояний до прототипов.

В настоящей статье авторы решают задачу разработки модели обобщенного классификатора на основе нечеткой логистической регрессии (лог-модель), обладающей свойствами интерпретируемости, то есть возможностью обосновывать процесс вывода результатов работы классификатора понятным для человека образом, а также способностью интегрировать знания специалистов в процесс проектирования, обучения и адаптации классификаторов.

2. МОДЕЛЬ НА ОСНОВЕ ЛОГИСТИЧЕСКОЙ РЕГРЕССИИ (ЛОГ-МОДЕЛЬ)

Рассмотрим задачу классификации d -мерных векторов-признаков $\mathbf{x} = (x_1, \dots, x_d)$ по k классам $Y \in \Theta = \{q_1, \dots, q_k\}$ на основе лог-модели. Обозначим через $p_i(\mathbf{x})$ вероятность того, что $Y = q_i$ при данном $\mathbf{X} = \mathbf{x}$. В модели множественной логистической регрессии предполагается, что лог-апостериорные вероятности $p_i(\mathbf{x})$ являются аффинными функциями от \mathbf{x} :

$$\ln P_k(\mathbf{x}) = \beta_k^T \mathbf{x} + \beta_{k_0}, \quad k=1, \dots, K, \quad (1)$$

где $p_k^{\text{in}}(\mathbf{x}) = P(Y = q_k | \mathbf{x})$ – логистическая вероятность класса q_k , $\beta_k \in \mathbb{R}^d$ и $\beta_{k_0} \in \mathbb{R}$ являются параметрами, зависящими от класса k . Апостериорная вероятность класса q_k может быть выражена из (1):

$$P(q_k | \mathbf{x}) = \frac{\exp(\beta_k^T \mathbf{x} + \beta_{k_0})}{\sum_{l=1}^K \exp(\beta_l^T \mathbf{x} + \beta_{l_0})}.$$

Классификатор на основе логистической регрессии является адаптивной моделью. Для обучающего множества примеров $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ параметры β и β_0 обычно оцениваются путем максимизации условного логарифмического правдоподобия:

$$\sum_{i=1}^n \sum_{j=1}^k [\delta_{Q(\mathbf{x}_i)}^j \ln p_j(\mathbf{x}_i) + (1 - \delta_{Q(\mathbf{x}_i)}^j) \ln(1 - p_j(\mathbf{x}_i))].$$

Преобразование линейных комбинаций признаков $\beta_k^T \mathbf{x} + \beta_{k_0} \in \mathbb{R}$ в вероятности из $[0, 1]$ называют *softmax*-преобразованием [15].

Классификаторы логистической регрессии определяют области решения, разделенные гиперплоскостями: они являются линейными классификаторами. Нелинейные классификаторы можно построить, применив логистическую регрессию к преобразованным признакам $\phi_j(x)$, ($j = 1, \dots, J$), где ϕ_j – нелинейные отображения: $\mathbb{R}^d \rightarrow \mathbb{R}$. Такие модели получили название классификаторов обобщенной логистической регрессии (*GLR*) (рис. 1).

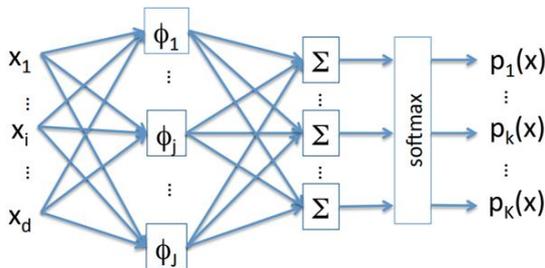


Рис. 1. Пример нечеткой онтологии

Признаки $\phi_j(x)$ и коэффициенты (β_k, β_{k_0}) обычно обучаются одновременно путем минимизации некоторой функции стоимости. Популярные модели, основанные на этом принципе, включают обобщенные аддитивные модели [16], многослойные нейронные сети с прямой связью [17]. В частности, нейронные сети с прямой связью (*FNN*)

представляют собой модели, состоящие из элементарных вычислительных блоков (или «нейронов»), расположенных в слоях. Каждый слой вычисляет вектор новых значений в качестве функций выходных данных предыдущего уровня. Для классификации выходной слой обычно является *softmax*-слоем с K выходами. Таким образом, эта модель эквивалентна логистической регрессии, выполненной для новых функций, вычисленных в скрытых слоях. Все веса в сети обучаются путем минимизации функции стоимости, которая часто принимается как обратная условная вероятность (перекрестная энтропия).

3. МОДЕЛЬ СЛИЯНИЯ СВИДЕТЕЛЬСТВ НА ОСНОВЕ ВЕСОВЫХ ФУНКЦИЙ

Особо важную роль в *DS*-анализе классификационных моделей и проектировании доказательных классификаторов играют, так называемые, простейшие функции вероятностных масс (симплекс функции). Простейшая симплекс функция m определяется следующим образом:

$$\exists A \subset \Omega \quad m(A) = p, \quad m(\Omega) = 1 - p. \quad (2)$$

Величина $p \in [0, 1]$ в (2) называется степенью поддержки класса (гипотезы) A .

Введем в рассмотрение двойственные весовые функции:

$$w := -\ln(1-p), \quad \bar{w} := -\ln(p).$$

Весовые функции обладают рядом важных свойств, позволяющих упростить вычисления, связанные с комбинированием свидетельств на основе *DS*-правил. Для двух m -функций m_1 и m_2 , заданных на одном и том же фокальном множестве A , имеют место следующие формулы вычисления ортогональных сумм:

$$\begin{aligned} (m_1 \oplus m_2)(\Omega) &= m_1(\Omega)m_2(\Omega), \\ \bar{w}_{1 \oplus 2}(\Omega) &= -\ln(m_1(\Omega)m_2(\Omega)) = -\ln m_1(\Omega) - \ln m_2(\Omega) = \bar{w}_1(\Omega) + \bar{w}_2(\Omega), \\ \bar{w}_{1 \oplus 2}(A) &= \bar{w}_1(A) + \bar{w}_2(A), \end{aligned} \quad (3)$$

$$\begin{aligned} (m_1 \oplus m_2)(A) &= 1 - (1 - m_1(A))(1 - m_2(A)), \\ w_{1 \oplus 2}(A) &= -\ln [1 - (m_1 \oplus m_2)(A)] = \\ &= -\ln(1 - m_1(A)) - \ln(1 - m_2(A)), \\ w_{1 \oplus 2}(A) &= w_1(A) + w_2(A). \end{aligned} \quad (4)$$

На основе индуктивного обобщения формул (3)-(4) получаем формулы вычисления ортогональных сумм для N m -функций m_i , заданных на одном фокальном множестве $A \subset \Omega$:

$$\begin{aligned}\bar{w}_{\oplus_{i=1}^N}(\Omega) &= \sum_{i=1}^N \bar{w}_i(\Omega), \\ w_{\oplus_{i=1}^N}(A) &= \sum_{i=1}^N w_i(A) = \bar{w}_{\oplus_{i=1}^N}(\Omega).\end{aligned}\quad (5)$$

Ортогональная сумма двух простых m -функций m_1 и m_2 , заданных на разных фокальных множествах A и B , уже не будет симплекс-функцией. Правила комбинирования свидетельств имеют вид:

$$\begin{aligned}\forall A, B \subset \Omega \quad A \cap B &= \emptyset, \\ (m_1(A) \oplus m_2(B))(A) &= m_1(A)m_2(\Omega): (1 - m_1(A)m_2(B)), \\ (m_1(A) \oplus m_2(B))(B) &= m_1(\Omega)m_2(B): (1 - k), \\ (m_1 \oplus m_2)(\Omega) &= m_1(\Omega)m_2(\Omega): (1 - k), \\ \bar{w}_{1 \oplus 2}(A) &= \bar{w}_1(A) + \bar{w}_2(\Omega) - \ln(1 - k), \\ \bar{w}_{1 \oplus 2}(B) &= \bar{w}_2(B) + \bar{w}_1(\Omega) - \ln(1 - k).\end{aligned}\quad (6)$$

Величина k в (6) является нормировочным коэффициентом.

Из (6) вытекает следующая формула вычисления ортогональной суммы N простых m -функций m_i , заданных на разных множествах $A_i \subset \Omega$ ($i = 1, \dots, N$):

$$\begin{aligned}\forall A_i \subset \Omega \quad \bigcap_{i=1}^K A_i &= \emptyset, \\ (m_1(A_1) \oplus \dots \oplus m_K(A_K))(A_j) &= m_j(A_j) \cdot \prod_{i=1, i \neq j}^K m_i(\Omega): (1 - k), \\ \bar{w}_{\oplus_{i=1}^K}(A_j) &= \bar{w}_j(A_j) + \sum_{i=1, i \neq j}^K \bar{w}_i(\Omega) - \ln(1 - k), \\ \bar{w}_{\oplus_{i=1}^K}(A_j) &= \bar{w}_j(A_j) + \sum_{i=1, i \neq j}^K w(A_i) - \ln(1 - k).\end{aligned}\quad (7)$$

Для двойственных весовых функций w и \bar{w} имеют место следующие переходные формулы преобразования весов:

$$\begin{aligned}w &= -\ln(1 - e^{-\bar{w}}), \quad \bar{w} = -\ln(1 - e^{-w}), \\ m(A) &= 1 - m(\Omega), \\ -\ln(m(A)) &= -\ln(1 - m(\Omega)), \quad \bar{w}(A) = w(\Omega),\end{aligned}\quad (8)$$

$$\begin{aligned}\bar{w}(\Omega) &= -\ln(m(\Omega)) = -\ln(1 - m(A)) = w(A), \\ w(A) &= -\ln(1 - m(A)) = -\ln(m(\Omega)) = \bar{w}(\Omega).\end{aligned}$$

Формулы (5)-(8) вместе с нечетким гранулированием логистической регрессии (1) положены в основу проектирования нечетких интерпретируемых *DS*-классификаторов.

4. НЕЧЕТКОЕ ОТОБРАЖЕНИЕ

Для дальнейшего повышения интерпретируемости и выразительных возможностей доказательных классификаторов ниже предлагается нечеткое обобщение лог-модели.

Также, как и ранее, предполагаем, что каждый признак x_i во входном векторе \mathbf{x} выступает для каждого класса q_k в качестве некоего свидетельства в пользу данного класса. Степень этого свидетельства p_{ki} оценивается нечеткой зависимостью w_{ik} , представленной в виде нечеткой модели Такаги-Сугено (*TS*-модели) нулевого порядка. Нечеткие правила *TS*-модели имеют вид:

$$R_{ij}^k: \text{ IF } x_i = A_{ij} \text{ THEN } w_i^k = z_{ij}^k,$$

где A_{ij} – нечеткий терм, характеризующий j -е лингвистическое значение признака x_i ; $\mathbb{R} \ni z_{ij}^k$ – значение весовой функции w_i^k для признака-свидетельства x_i , поддерживающее класс q_k ($z_{ij}^k \geq 0$) в нечетком правиле R_{ij}^k .

TS-модель представляет собой кусочно-линейные аппроксимации весовой функции масс и является обобщением линейной логистической модели. Нечеткий вывод в *TS*-модели осуществляется путем вычисления для каждого входного x_i значения весовой функции $w_i(x)$ на основании формулы:

$$y = f_1(x_1) + \dots + f_n(x_n) = \sum_{i=1}^n f_i(x_i) = \sum_{i=1}^n y_i(x_i). \quad (9)$$

Таким образом, вес $w_i(x)$ вычисляется на основе L нечетких правил *TS*-модели. Область каждой входной переменной x_i гранулируется с использованием комплементарных функций принадлежности (ФП) треугольного вида. Такой выбор ФП, во-первых, обеспечивает наилучшую интерпретируемость нечетких разбиений признаков шкал, порождаемых ФП, а, во-вторых, позволяет обойтись без нормализации выходных значений в формуле (9).

5. ЗАКЛЮЧЕНИЕ

В статье рассмотрена модель логистической регрессии лог-модель, а также ее расширения, включая нечеткую лог-модель. Предложенные расширения лог-модели открывают путь к новым исследованиям, цель которых состоит в том, чтобы лучше использовать существую-

щие модели классификаторов и разрабатывать новые алгоритмы слияния данных с учетом повышенной интерпретируемости и информативности классификатора, а также возможности интеграции в них априорной экспертной информации на стадиях проектирования и обучения.

Литература

1. Dempster A.P. Upper and lower probabilities induced by a multivalued mapping. *Annals of Mathematical Statistics*. – 1967. – Vol. 38. – P. 325-339.
2. Shafer G. A mathematical theory of evidence. – Princeton University Press, Princeton, N.J., 1976.
3. Yager R.R., Liu L. editors. *Classic Works of the Dempster-Shafer Theory of Belief Functions*. – Springer, Heidelberg, 2008.
4. Rogova G. Combining the results of several neural network classifiers // *Neural Networks*. – 1994. – Vol. 7 (5). – P. 777-781.
5. Xu L., Krzyzak A., Y. C. Suen. Methods of combining multiple classifiers and their applications to handwriting recognition // *IEEE Transactions on Systems, Man and Cybernetics*. – 1992. – Vol. 22 (3). – P. 418-435.
6. Liu Z.-G., Zhang Z., Liu Y., Dezert J., Pan Q. A new pattern classification improvement method with local quality matrix based on K-NN // *Knowledge-Based Systems*. – 2019. – Vol. 164. – P. 336-347.
7. Dencœur T. Analysis of evidence-theoretic decision rules for pattern classification // *Pattern Recognition*. – 1997. – Vol. 30 (7). – P. 1095-1107.
8. Dencœur T., Smets P. Classification using belief functions: the relationship between the case-based and model-based approaches // *IEEE Transactions on Systems, Man and Cybernetics*. – 2006. – Vol. 36 (6). – P. 1395-1406.
9. Smets P. Belief functions: the disjunctive rule of combination and the generalized Bayesian theorem // *International Journal of Approximate Reasoning*. – 1993. – Vol. 9. – P. 1-35.
10. Chen X.-L., Wang P.-H., Hao Y.-S., Zhao M. Evidential KNN-based condition monitoring and early warning method with applications in power plant. – *Neurocomputing*, 2018.
11. Guettari N., Capelle-Laiz'e A. S., Carr'e P. Blind image steganalysis based on evidential k-nearest neighbors // In 2016 IEEE International Conference on Image Processing (ICIP), Sept 2016. – P 2742-2746.
12. Dencœur T. A k-nearest neighbor classification rule based on Dempster-Shafer theory // *IEEE Trans. on Systems, Man and Cybernetics*. – 1995. – Vol. 25 (05). – P. 804-813.
13. Jiao L., Pan Q., Feng X., Yang F. An evidential k-nearest neighbor classification method with weighted attributes // In Proceedings of the 16th International Conference on Information Fusion, July 2013. – P. 145-150.
14. Dencœur T. A neural network classifier based on Dempster-Shafer theory // *IEEE Trans. on Systems, Man and Cybernetics*. – 2000. – Vol. 30 (2). – P. 131-150.
15. Hastie T., Tibshirani R., Friedman J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. – Springer, 2nd edition, 2009.
16. Hastie T.J., Tibshirani R.J. *Generalized Additive Models*. – Chapman and Hall/CRC, London, 1990.
17. Goodfellow I., Bengio Y., Courville A. *Deep Learning*. – MIT Press, 2016. <http://www.deeplearningbook.org>.

ВЕРОЯТНОСТНЫЕ МОДЕЛИ В ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЯХ*

Иляхинский А.В. к.т.н., с.н.с.

ИПМ РАН, Нижний Новгород

e-mail: ilyahinsky-aleks@bk.ru

Богатиков В.Н. д.т.н., профессор

ТвГТУ

e-mail: vnbgtk@mail.ru

1. ВВЕДЕНИЕ

Современные информационные системы представляют собой комплекс различных подсистем, выполняющих функции передачи и обработки данных связанные между собой процессами интенсивного динамического взаимодействия обмена энергией и информацией. Вместе с тем необходимо отметить, что обработка данных предусматривает наличие некоторых механизмов анализа полученных результатов с целью выбора оптимального управленческого решения. К тому же при аналитическом исследовании поведения информационных систем важным является обоснование математических моделей процессов, протекающих в этих системах. Эти модели, с одной стороны, должны наиболее адекватно отражать свойства реальных сигналов, с другой – допускать исследования этих процессов известными теоретическими методами. Модели случайных процессов являются наиболее типичными в практике исследований динамических систем с учетом реальных режимов, возмущений и помех. Статистическая модель случайного процесса – это алгоритм, с помощью которого имитируют работу сложной системы, подверженной случайным возмущениям; имитируют взаимодействие элементов системы, носящих вероятностный характер.

* Работа выполнена при финансовой поддержке РФФИ, проект № 20-07-00914, а также в рамках государственного задания ИПФ РАН на проведение фундаментальных научных исследований на 2021-2023 гг. по теме № 0030-2021-0025.

2. РАСПРЕДЕЛЕНИЯ ВЕРОЯТНОСТЕЙ КАК СТАТИСТИЧЕСКИЕ МОДЕЛИ

Статистическое исследование, как правило, начинается с анализа закона распределения рассматриваемой случайной величины. Семейства распределений Пирсона [1] и Джонсона [2] многие десятилетия используются в задачах аппроксимации эмпирических распределений, расчета и оценки вероятностных характеристик [3]. Спектр представления экспериментальных данных эмпирическими распределениями достаточно широк: в задачах анализа изображений [4], в моделировании радиолокационных сигналов [5], в оценке рисков [6], в страховом деле [7], в регрессионном анализе [8], в оценке надежности [9], а также во многих других прикладных задачах теории вероятностей и математической статистики [10, 11]. Для большинства практических расчетов используются такие распределения Пирсона как бета-распределение (распределение Пирсона I типа), гамма-распределение (распределение Пирсона III типа), распределение Стьюдента (распределение Пирсона VII типа), показательное распределение (распределение Пирсона X типа), нормальное распределение (распределение Пирсона XI типа). В инженерной практике для описания многих реальных процессов особое место занимают гауссовские случайные процессы, связанные с аппроксимацией, нормальным или гауссовым распределением. Такое положение сложилось, как метко заметил один из физиков потому, что «экспериментаторы верят в нормальные распределения, полагаясь на доказательства математиков (центральная предельная теорема), а математики – полагаясь на экспериментальное обоснование» [12]. Параметры выбранного распределения (среднее, коэффициент вариации, центральные моменты и т.д.) позволяют дать единое количественное описание наиболее общих закономерностей и свойств, характерных для объектов и явлений самой различной природы. В этом и сила, и слабость статистических методов. Уходя от узких понятий, связанных с объектом определенной природы, при исследованиях совершенно незнакомых объектов, можно обнаружить в их свойствах много общего со свойствами хорошо изученных объектов. Однако сложившиеся к настоящему времени статистические методы, использующие эмпирические распределения, не всегда позволяют связать результаты анализа с состоянием исследуемого объекта или природного явления. Это связано с тем, что в основном применяются распределения параметров сигнала – *модели данных* и не используются все или используются в ограниченном объеме распределения, основанные на гипотезах о структуре и процессах моделируемой системы – *модели систем*.

Или в известной формулировке В.В. Налимова о выборе распределения вероятностей в качестве статистической модели «выбор распределения должен базироваться, прежде всего, на понимании механизма изучаемого явления» [13]. Распределение как модель изучаемого явления должно быть информационно эквивалентно объекту исследования путем соблюдения следующих условий [14, 15]:

- распределение, выбранное в качестве статистической модели, должно быть определено на ограниченном интервале;
- энтропия распределения должна состоять из произведения и потока;
- распределения, определенные на одномерных симплексах, должны допускать переход к распределению, определенному на многомерном симплексе.

Следует отметить, что определенное на неограниченном интервале нормальное распределение, как статистическая модель исследуемого процесса, предполагает наличие у исследуемого объекта физических свойств, параметры которых стремятся к неограниченно большой или малой величине. Очевидно, что таких свойств нет. При этом гамма-распределение используется для описания случайных величин ограниченных с одной стороны и только бета-распределение – для описания случайных величин, ограниченных сверху и снизу.

3. РАСПРЕДЕЛЕНИЕ ДИРИХЛЕ – СТАТИСТИЧЕСКАЯ МОДЕЛЬ ПРОЦЕССОВ В СЛОЖНЫХ МНОГОМЕРНЫХ И МНОГОСВЯЗАННЫХ СИСТЕМАХ

Ограниченная конечным интервалом, статистическая модель распределение Дирихле оказалась полезной при описании процессов, связанных с явлением акустической эмиссии [16]. Модель отражает результат совместной реализации $n-1$ независимых процессов x_j , протекающих со скоростями (интенсивностями) v_j , и противоположного им по смыслу процесса, протекающего со скоростью v_n . В этом смысле распределение информационно эквивалентно объекту любой природы и степени сложности. Функция плотности вероятности распределения Дирихле, определенная на k -мерном симплексе, равна

$$D(x_1, \dots, x_k) = \frac{\Gamma(a_n)}{\prod_{i=1}^n \Gamma(v_i)} \prod_{i=1}^k x_i^{v_i-1} \times (1 - \sum_{i=1}^k x_i)^{v_n-1}, \quad (1)$$

$$0 \leq \sum x_i \leq 1; v_i \geq 0, \dots, v_n \geq 0; \sum_{i=1}^n v_i = a_n; n = k + 1.$$

Если рассматривать состояние исследуемого объекта как результат совместной реализации двух независимых и противоположных по смыслу процессов: x , протекающего со скоростью (интенсивностью) v_1 , и противоположного ему процесса $1-x$, протекающего со скоростью v_2 , то в качестве статистической моделью такой системы может быть выбрано, определенное на одномерном симплексе распределение Дирихле – бета-распределение, с функцией плотности вероятности

$$Be(x) = \frac{\Gamma(\alpha)}{\Gamma(v_1)\Gamma(v_2)} x^{v_1-1}(1-x)^{v_2-1}, \quad (2)$$

$$0 \leq x \leq 1, v_1 > 0, v_2 > 0, \alpha = v_1 + v_2.$$

В (1) и (2) $\Gamma(x)$ – гамма-функция. Энтропия распределений, согласно определению,

$$H = - \int_x \phi(x) \log_\alpha \phi(x) dx \quad (3)$$

составляет с точностью до основания логарифма для распределения Дирихле

$$H(v_1, \dots, v_n) = \prod_{i=1}^n \Gamma(v_i) - \ln \Gamma(\alpha_n) + (\alpha_n - n)\psi(\alpha_n) - \sum_{i=1}^n (v_i - 1)\psi(v_i) \quad (4)$$

и

$$H(v_1, v_2) = \ln \frac{\Gamma(v_1)\Gamma(v_2)}{\Gamma(\alpha)} + (\alpha - 2)\psi(\alpha) - (v_1 - 1)\psi(v_1) - (v_2 - 1)\psi(v_2) \quad (5)$$

для бета-распределения. Здесь в (4) и (5) $\psi(x) = \frac{d}{dx} \ln \Gamma(x)$ – логарифмическая производная гамма-функции (пси-функция Эйлера). При $n = 2$ энтропия (3) соответствует (4). Энтропия распределения Дирихле может быть представлена в виде суммы

$$H(D) = H_i(v_1, \dots, v_n) + H_e(a_n), \quad (6)$$

в которой всегда положительное слагаемое

$$H_i(v_1, \dots, v_n) = \ln \prod_{i=1}^n \Gamma(v_i) - \sum_{i=1}^n (v_i - 1)\psi(v_i) \quad (7)$$

представляет собой, отвечающее второму закону термодинамики, производство энтропии, а слагаемое

$$H_e(a_n) = -\ln \Gamma(a_n) + (a_n - n)\psi(a_n) \quad (8)$$

представляет собой, характеризующий процессы взаимодействия с внешней средой, поток энтропии. При $n \geq 3$ поток энтропии (7) может принимать как положительные, так и отрицательные значения, что в

терминах модели распределения Дирихле позволяет рассматривать $H_e(a_n) < 0$ как одно из условий самоорганизации и свидетельствовать о процессах возникновения упорядоченных пространственно-временные образования [17] (диссипативных структур по терминологии И. Пригожина).

В случае бета-распределения $H_e(a_n)$ всегда имеет положительное значение и как статистическая модель не может отражать ситуаций, когда в объекте возможны процессы самоорганизации. При неограниченном росте числа процессов, протекающих с одинаковыми скоростями ($v_1 = \dots = v_n = v_\gamma$), распределение Дирихле может быть приближенно заменено гамма-распределением с плотностью

$$\gamma(x) = \frac{x^{v_\gamma-1}}{\Gamma(v_\gamma)} \exp(-x), x \geq 0, v_\gamma \geq 0 \quad (9)$$

и энтропией

$$H_\gamma = v_\gamma + \ln \Gamma(v_\gamma) - (v_\gamma - 1)\psi(v_\gamma). \quad (10)$$

Выражение для энтропии гамма-распределения не позволяет выделить какую-либо его часть, характеризующую поток энтропии. Поэтому модель, представленная гамма-распределением, соответствует изолированной системе. Следовательно, выбор этой модели при описании процессов в задачах зондирования атмосферы [18] или в задачах обнаружения [19] некорректно, поскольку изолированная система не учитывает взаимодействия объекта с внешней средой. Как статистическая модель гамма-распределение представляет собой лишь в той степени удовлетворительную модель наблюдения состояния объекта, в какой возможна реализация неограниченно возрастающего числа процессов, протекающих с одинаковыми скоростями. При $v_\gamma = 1$ (8) переходит в экспоненциальное распределение – модель наблюдения (но не механизма) радиоактивного распада. Если скорости процессов принимают дискретные значения, а число процессов велико, но конечно, то модель будет соответствовать распределению Пуассона, которое имеет вид:

$$f_p = \frac{v^x}{x!} \exp(-x), v > 0, x = 1, 2, \dots \quad (11)$$

Как и гамма-распределение, распределение Пуассона представляет собой флуктуационную модель, так как математическое ожидание (пространственно-временная характеристика) для (10) по определению равно второму центральному моменту (импульсно-энергетическая характеристика). Выражение для энтропии распределения (10)

$$H_p = v(1 - \ln v) + \ln \Gamma(1+v) + \frac{v}{2} \psi'(1+v) \quad (12)$$

не позволяет, как и в случае с гамма-распределением, выделить поток энтропии. Следовательно, модель (10) соответствует изолированной системе со всеми вытекающими из этого последствиями.

4. ЗАКЛЮЧЕНИЕ

Таким образом, из всех рассмотренных распределений, только распределение Дирихле обладает достаточной гибкостью и реализмом, что выгодно отличает его от моделей, построенных на основе нормального распределения, бета-распределения, гамма-распределения и распределения Пуассона.

Литература

1. Pearson, K. Contributions to the Mathematical Theory of Evolution. II. Skew Variations in Homogeneous Material // Philosophical Transactions of the Royal Society of London. Ser. A. – 1895. – Vol. 186. – P. 343-414.
2. Johnson N.L. Systems of Frequency Curves Generated by Methods of Translation // Biometrika. – 1949. – Vol. 36, No 1/2. – P. 149-176.
3. Джонсон Н.Л. Коц С., Балакришнан Н. Одномерные непрерывные распределения: в 2 ч. Ч. 1. – М.: БИНОМ. Лаборатория знаний, 2010. – 703 с.
4. Воронов И.В. Применение универсального семейства распределений Пирсона для аппроксимации распределения значений вектора псевдоградиента при совмещении изображений // Радиоэлектронная техника. – 2015. – № 2 (8). – С. 123-127.
5. Карпов И.Г., Галкин Е.А. Модернизация распределений Пирсона для аппроксимации экспериментальных распределений радиолокационных сигналов // Известия вузов. Радиоэлектроника. – 2004. – Т. 47, № 11. – С. 52-61.
6. Naguez N. Dynamic Portfolio Insurance Strategies: Risk Management under Johnson Distributions // Annals of Operations Research. 06 February 2016. – P. 1-25.
7. Смагина С.П., Кошкин Г.М. Расчет тарифных ставок с использованием кривых Джонсона при страховании автотранспорта // Известия высших учебных заведений. Физика. – 2015. – Т. 58, № 11-2. – С. 297-302.
8. Тимофеев В.С. Оценивание параметров регрессионных зависимостей с использованием кривых Пирсона // Науч. вестн. Новосибир. гос. техн. ун-та. – 2009. – № 4 (37). – С. 57-66.
9. Игнатенко В.И. Применение распределений Джонсона и Пирсона в задачах технической эксплуатации автомобилей // Сборник научных трудов конференции «Проблемы современной науки». – Липецк: ЛГТУ, 2016. – С. 63-66.
10. Mateus A., Tome M. Fitting Johnson's SB Distribution to Forest Tree Diameter // Advances in Regression, Survival Analysis, Extreme Values, Markov Processes and Other Statistical Applications. Studies in Theoretical and Applied Statistics. – Berlin: Springer-Verlag, 2013. – P. 289-296.

11. Губарев В.В. Идентификация эмпирических распределений // Вестник Самарского государственного аэрокосмического университета. – 2011. – № 6 (30). – С. 205-215.
12. Сквайрс Дж. Практическая физика. – М. Мир, 1964. – 300 с.
13. Налимов В.В. В кн.: Хан Г., Шапиро С. Статистические модели в инженерных задачах. – М.: Мир, 1969. – С. 5-6.
14. Серeda Ю.С. Проблемы информационно-статистической теории. – Н. Новгород: Типография «Поволжье», 2007. – 356 с.
15. Шеннон Р. Имитационное моделирование – искусство и наука. – М.: Мир, 1978. – 424 с.
16. Иляхинский А.В., Родюшкин В.М., Рябов Д.А., Хлыбов А.А., Ерофеев В.И. Исследование сигналов акустической эмиссии при испытании на растяжение стали 20 // Проблемы прочности и пластичности. – 2021. – Т. 83, № 2. – С. 133-139.
17. Пригожин И. Время, структура и флуктуации // Успехи физических наук. – 1980. – Т. 131. – Вып. 2. – С. 185-207.
18. Ватутин В.А., Телевинова Т.М., Чистяков В.П. Вероятностные методы в физических исследованиях. – М.: Наука, 1985. – 208 с.
19. Иган Дж. Теория обнаружения сигналов и анализ рабочих характеристик. – М.: Наука, 1983. – 216 с.

МЕТОД И ПАРАЛЛЕЛЬНАЯ РЕАЛИЗАЦИЯ НЕЧЕТКОЙ КЛАССИФИКАЦИИ ОБЪЕКТОВ С НЕСИНГЛТОННОЙ ФАЗЗИФИКАЦИЕЙ*

*Каратач С.А., аспирант
БГТУ им. В.Г. Шухова
e-mail: karatach1998@yandex.ru
Синюк В.Г., к.т.н., профессор
БГТУ им. В.Г. Шухова
e-mail: vgsinuk@mail.ru*

1. ВВЕДЕНИЕ

Наряду классификацией, использующей методы теории вероятностей, нейронные сети, разделения линейных пространств и другие, известны методы классификации, основанные на нечетком выводе [1]. Особенностью такого подхода является сохранение лингвистической интерпретации признакового описания классифицируемых объектов.

Обычно эти методы значения используют фаззификации входных признаков типа синглтон (singleton). Это обусловлено стремлением снизить вычислительную сложность алгоритмов нечеткого вывода. Однако метод нечеткого вывода на основе теоремы о декомпозиции [2] позволяет реализовать эффективный вывод с использованием несинглтонной (non-singleton) фаззификации значений входных признаков [3, 4]. Например, в случае зашумленных значений признаков, несинглтонная фаззификация позволяет отразить в функции принадлежности степень соответствия данного значения признака вероятностному распределению этого признака. Данный подход хорошо адаптирован к применению параллельных вычислений. В совокупности эти факторы позволяют повысить эффективность решения задачи классификации.

2. НЕЧЕТКИЙ ВЫВОД ДЛЯ МОДЕЛИ С МИМО-СТРУКТУРОЙ

Используемая нечеткая модель имеет МИМО-структуру [5, 6] и представляет собой базу нечетких правил $R_k, k = \overline{1, N}$ вида:

$$R_k: \text{Если } x_1 \text{ есть } A_{1k} \text{ и } x_2 \text{ есть } A_{2k} \text{ и, } \dots, \text{ и } x_n \text{ есть } A_{nk}, \quad (1)$$

* Работа выполнена при финансовой поддержке РФФИ, проект № 20-07-00030.

то y_1 есть B_{1k} и y_2 есть B_{2k} и ... и y_m есть B_{mk} ,

где N – количество нечетких правил, $A_{ik} \subseteq X_i, i = \overline{1, n}$ и $B_{jk} \subseteq Y_j, j = \overline{1, m}$ – нечеткие множества, которые с функциями принадлежности $\mu_{A_{ik}}(x_i)$ и $\mu_{B_{jk}}(y_j)$, а x_1, \dots, x_n и y_1, \dots, y_m – входные и выходные лингвистические переменные, которые образуют два вектора:

$$[x_1, x_2, \dots, x_n]^T = \mathbf{x} \in X_1 \times X_2 \times \dots \times X_n = \mathbf{X},$$

$$[y_1, y_2, \dots, y_m]^T = \mathbf{y} \in Y_1 \times Y_2 \times \dots \times Y_m.$$

Обозначив $A_{1k} \times \dots \times A_{nk} = \mathbf{A}_k$, а также рассматривая выходы системы как независимые, можно записать правило (1) в виде нечеткой импликации:

$$R_{jk}: \mathbf{A}_k \rightarrow B_{jk}, j = \overline{1, m}.$$

Импликацию, в свою очередь, можно выразить как нечеткое отношение, определённое на базовом множестве $\mathbf{X} \times Y_j$, а $R_{jk} \subseteq \mathbf{X} \times Y_j$ – нечеткое множество с функцией принадлежности:

$$\mu_{R_{jk}}(\mathbf{x}, y_j) = \mu_{\mathbf{A}_k \rightarrow B_{jk}}(\mathbf{x}, y_j) = I(\mu_{\mathbf{A}_k}(\mathbf{x}), \mu_{B_{jk}}(y_j)).$$

Тогда, в соответствии с нечетким правилом *fuzzy modus ponens*, нечеткий вывод для нечеткого отношения R_k и комбинации поданных на вход нечеткой модели нечетких множеств $A'_1 \times \dots \times A'_n = \mathbf{A}'$ определяется их композицией:

$$B'_{jk} = \mathbf{A}' \circ (\mathbf{A}_k \rightarrow B_{jk}). \quad (2)$$

Вычислительная сложность выражения (2) определяется мощностью базовых множеств $O(|X_1| \times \dots \times |X_n| \times |Y_j|)$. Однако в результате применения теоремы о декомпозиции [2, 7] функция принадлежности нечеткого множества B'_{jk} выражается как:

$$\begin{aligned} \mu_{B'_{jk}}(y_j) &= \sup_{\mathbf{x} \in \mathbf{X}} \left\{ \mu_{\mathbf{A}_k}(\mathbf{x})^T * I(\mu_{\mathbf{A}_k}(\mathbf{x}), \mu_{B_{jk}}(y_j)) \right\} = \\ &= \max_{i=1, n} \left\{ \sup_{x_i \in X_i} \left\{ \mu_{A'_i}(x_i)^T * I(\mu_{A_{ik}}(x_i), \mu_{B_{jk}}(y_j)) \right\} \right\}. \end{aligned} \quad (3)$$

Тогда, дискретизированный вариант последнего выражения характеризуется полиномиальной вычислительной сложностью $O(n \times |X_i| \times |Y_j|)$.

3. ВЫВОД ДЛЯ БАЗЫ ПРАВИЛ

Для получения результата нечеткого вывода по j -му выходу для базы из N правил, предлагается использовать метод дефаззификации центра тяжести:

$$y'_j = \frac{\sum_{k=1}^N \bar{y}_{jk} \times \mu_{B'_j}(\bar{y}_{jk})}{\sum_{k=1}^N \mu_{B'_j}(\bar{y}_{jk})}. \quad (4)$$

Нечеткое множество B'_j формируется посредством пересечения:

$$B'_j = \bigcap_{k=1}^N B'_{jk}.$$

Тогда, функция принадлежности нечеткого множества B' определяется как результат применения – нормы, т.е.:

$$\mu_{B'}(z_j) = \bigwedge_{l=1, \overline{N}} \mu_{B'_l}(z_j). \quad (5)$$

Если выполняется условие о невозрастании $I(\mu_{A_{ik}}(x_i), \mu_{B_{jk}}(y_j))$, $i = \overline{1, n}$, $j = \overline{1, m}$ относительно первого аргумента и в правилах используется связка «И», то получим выражение степени принадлежности к классу w_j .

$$y'_j = \frac{\sum_{k=1, N} \bar{y}_{jk} \cdot T_{j=1, N} \left\{ \max_{i=1, n} \left\{ \sup_{t_i \in [0, 1]} \left\{ \mu_{A'_{ii}}(x_i)^T * I(\mu_{A'_{ii}}(x_i), \mu_{B_{jl}}(\bar{y}_{jk})) \right\} \right\} \right\}}{\sum_{k=1, N} T_{j=1, N} \left\{ \max_{i=1, n} \left\{ \sup_{t_i \in [0, 1]} \left\{ \mu_{A'_{ii}}(x_i)^T * I(\mu_{A'_{ii}}(x_i), \mu_{B_{jl}}(\bar{y}_{jk})) \right\} \right\} \right\}}. \quad (6)$$

4. НЕЧЕТКИЙ КЛАССИФИКАТОР

Описанную выше нечеткую модель можно использовать для определения принадлежности некоторого объекта к каждому из заданного множества классов, то есть решать задачу мультиклассификации.

В классических методах классификация проводится для данного объекта q с набором значений атрибутов, каждое из которых формализуется числовым значением признака. В случае использовании нечеткой модели, значения признаков формализуются посредством термов лингвистических переменных, совокупность значений которых формирует вектор $[x_1, \dots, x_n]$. При этом рассмотренный выше нечеткий вывод позволяет задавать значения признаков объекта с использованием несингтонной фаззификации, учитывающей зашумленность значений данного признака для набора классифицируемых объектов. На-

пример, значение лингвистической переменной может быть задано нечетким множеством, характеризующегося гауссовой функцией принадлежности, для которой матожидание равно значению признака объекта, а среднеквадратичное отклонение совпадает с среднеквадратичным отклонением всего набора объектов.

Пусть классификация проводится для m классов $\Omega = \{\omega_1, \dots, \omega_m\}$. Тогда нечеткий классификатор включает в себя набор из N правил:

$$R_k: \text{Если } x_1 \text{ есть } A_{1k} \text{ и } x_2 \text{ есть } A_{2k} \text{ и, } \dots, \text{ и } x_n \text{ есть } A_{nk}, \\ \text{то } q \in \omega_1(\bar{z}_{1k}) \text{ и } q \in \omega_2(\bar{z}_{2k}) \text{ и } \dots \text{ и } q \in \omega_m(\bar{z}_{mk}). \quad (7)$$

В этих правилах степень принадлежности объекта q к классу ω_j задается значением \bar{z}_{jk} , которому можно поставить в соответствие значение лингвистической переменной z_j . Это значение выражается нечетким множеством с базовым множеством $[0,1]$, и синглтоном в качестве функции принадлежности:

$$\mu_{B_{jk}}(z_j) = \begin{cases} 1, & \text{если } z_j = \bar{z}_{jk}, \\ 0, & \text{если } z_j \neq \bar{z}_{jk}. \end{cases}$$

Тогда, правило (7) можно переписать в виде (1), заменив лингвистические переменные y_j на z_j , где $j = \overline{1, m}$.

$$R_k: \text{Если } x_1 \text{ есть } A_{1k} \text{ и } x_2 \text{ есть } A_{2k} \text{ и, } \dots, \text{ и } x_n \text{ есть } A_{nk}, \\ \text{то } z_1 \text{ есть } B_{1k} \text{ и } z_2 \text{ есть } B_{2k} \text{ и } \dots \text{ и } z_m \text{ есть } B_{mk}.$$

И в итоге выражение (6) примет вид:

$$z'_j = \frac{\sum_{k=1, N} \bar{z}_{jk} \prod_{l=1, N} \left\{ \max_{i=1, n} \left\{ \sup_{t_i \in [0, 1]} \left\{ \mu_{A'_{il}}(x_i)^T I(\mu_{A'_{il}}(x_i), \mu_{B_{jl}}(\bar{z}_{jk})) \right\} \right\} \right\}}{\sum_{k=1, N} \prod_{l=1, N} \left\{ \max_{i=1, n} \left\{ \sup_{t_i \in [0, 1]} \left\{ \mu_{A'_{il}}(x_i)^T I(\mu_{A'_{il}}(x_i), \mu_{B_{jl}}(\bar{z}_{jk})) \right\} \right\} \right\}}$$

5. ПРЕДСТАВЛЕНИЕ НЕЧЕТКОЙ МОДЕЛИ ДЛЯ ОРГАНИЗАЦИИ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ

В качестве инструмента реализации предложенной модели использовалась библиотека для языка программирования Python Numba, предоставляющая набор декораторов, позволяющих выполнить JIT-компиляцию функций, написанных на языке Python в низкоуровневое представление.

Для параллельной реализации вывода использовалась программная технология CUDA, подразумевающая одновременное исполнение программного кода на центральном процессоре и на графическом процессоре.

Данные организовывались в виде сгруппированных n -мерных тензоров 1-го, 2-го и 3-го порядка, располагаемые таким образом, что элементы 1-го порядка были выровнены для более эффективного доступа к ним из отдельных блоков. Первая группа данных формирует набор правил, где каждый элемент представляет собой параметры гауссовой функции принадлежности. Далее следуют входные данные, подлежащие классификации.

Эффективность вычислений достигалась за счет их оптимизации, исходя из принципов высокопроизводительных CUDA-приложений [8–10]:

- максимизация занятости (осципансу) потоковых мультипроцессоров посредством сбалансированной конфигурации CUDA-ядра, при которой задействовано максимально возможное количество потоков при минимальном простое модулей потокового мультипроцессора;
- кэширование данных в быстрой разделяемой памяти;
- использование эффективной схемы свертки, исключающей конфликты банков разделяемой памяти, и расходящиеся потоки исполнения.

6. РЕЗУЛЬТАТЫ ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА

Параллельная реализация нечеткого классификатора сравнивалась с ее с однопоточной реализацией. Нечеткая классификация проводилась для датасета KDD99 [11], содержащего 494021 записей, каждая из которых фиксирует различные атрибуты TCP пакетов. Набор атрибутов включает 41 количественный и категориальный входные признаки, а также относит пакет либо к нормальным, либо к одному из 4-х классов атак: DOS, R2L (Remote to user), U2R (User to root), Probing (атака зондированием). Из всех входных признаков были отобраны 7 значащих количественных признаков, в перечень которых включались признаки, обладающие приемлемой степенью зашумления, позволяющей провести разделение между некоторым подмножеством классов. Распределение значений параметров в различных классах и их зашумленный характер показан на рис. 1.

Для оценки точности нечеткого классификатора использовались метрики: precision, recall, f1-мера. Также была выполнена ресурсоемкость алгоритма нечеткой классификации. Точность классификации, достигнутая на текущем этапе исследования, приведена в табл. 1. Время вывода по всему датасету для параллельной реализации составило в среднем 4 сек, а для многопоточной – 32 сек.

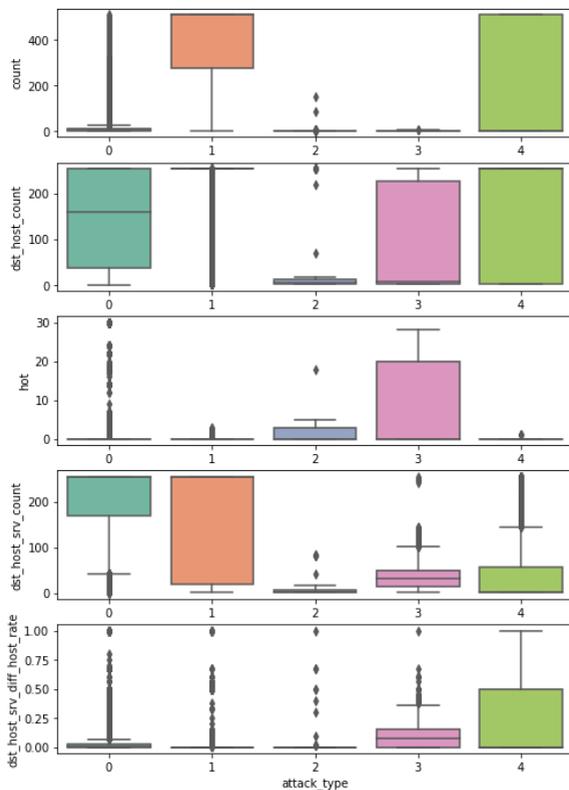


Рис. 1. Зашумленный характер данных на диаграммах распределения значений значащих признаков

Таблица 1

Значения метрик качества модели

Класс	Кол-во объектов	Precision	Recall	F1-мера	F1-мера (наивный байес)	F1-мера (SVM)
0 – Нет атаки	97278	0.98	0.24	0.38	0.37	0.99
1 – DOS	391458	0.88	0.94	0.91	0.40	0.83
2 – R2L	72	0.00	0.40	0.00	0.18	0.06
3 – U2R	1106	0.04	0.56	0.07	0.01	0.00
4 - Probing	4107	0.00	0.00	0.00	0.75	0.74
Взвешенное среднее	494021	0.89	0.80	0.80	0.39	0.90

Полученные результаты вычислительного эксперимента показали целесообразность использования параллельного подхода при реализации алгоритма нечеткой классификации.

7. ЗАКЛЮЧЕНИЕ

Результаты экспериментов показали, что точность классификации с использованием предлагаемого нечеткого классификатора для объектов с зашумленными признаками сопоставима с другими методами. Реализация же параллельного подхода обеспечивает увеличение производительности классификации в сравнении с однопоточным подходом до одного порядка.

Дальнейшие исследования будут направлены на повышение точности нечеткого классификатора за счёт предподготовки зашумленных данных и разделения многомерного признакового пространства.

Литература

1. Spałka K., Rutkowski L. Neuro-fuzzy structures for pattern classification // In Proceedings of the 9th International Conference on Computers ICCOMP'05. – Wisconsin, USA, 2005. – P. 1-4.
2. Синюк В.Г., Пенченко М.В. Методы нечеткого вывода для одного класса систем MISO-структуры при нечетких входах // Искусственный интеллект и принятие решений. – 2017. – № 4. – С. 33-39.
3. Борисов А.Н., Алексеев А.В., Крумберг О.А., Модели принятия решений на основе лингвистической переменной. – Рига: Зинатне, 1982.
4. Дюбуа Д., Прад А. Теория возможностей. Приложения к представлению знаний в информатике. – М.: Радио и связь, 1990.
5. Рутковская Д., Пилиньковский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. – М.: Горячая линия – Телеком, 2004.
6. Заде Л. Понятие лингвистической переменной и его применение к принятию приближенных решений. – М.: Мир, 1976.
7. Sinuk V., Karatach S.A. Inference method and parallel implementation for MISO-structure systems for inputs with linguistic values // Journal of informational technologies and computing systems. – 2020. – P. 85-93.
8. Sanders J., Kandrot E. CUDA by example: An introduction to general-purpose GPU programming, Addison-Wesley Professional, 2010.
9. CUDA Best Practices Guide [Online]. Available: Nvidia Developer Zone. [Accessed 11 April 2022].
10. CUDA Programming Guide [Online]. Available: Nvidia Developer Zone. <https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html>. [Accessed 11 April 2022].
11. UCI Knowledge Discovery in Databases Archive. KDD Cup 1999 Data. [Online]. Available: <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>. [Accessed 11 April 2022].

СРАВНЕНИЕ КАЧЕСТВА ДАННЫХ ГЕОМАГНИТНЫХ ОБСЕРВАТОРИЙ, ВХОДЯЩИХ В МИРОВУЮ СЕТЬ INTERMAGNET С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ BIG DATA

*Коробейников А.Г., д.т.н., профессор
СПбФ ИЗМИРАН
e-mail: Korobeynikov_A_G@mail.ru
Мунтян Е.Р., к.т.н., доцент
Южный федеральный университет
e-mail: ermuntyan@sfnu.ru*

1. ВВЕДЕНИЕ

Интеллектуальные методы работы с «большими данными» (Big Data) значительно повышают эффективность обработки информации о геофизических измерениях [0]. Реализация этих методов произведена в различных программных инструментариях. При исследовании, результаты которого представлены в данной работе, в качестве такого инструментария, под управлением операционной системы Windows, использовалась система MATLAB версии R2021b, позволяющая решать задачи в различных предметных областях [2–9]. Например, для решения задач предварительной обработки первичных данных, получаемых с датчиков входящих в состав различных геофизических измерительных комплексов [0, 0].

В работе представлены результаты решения задачи сравнения данных о геомагнитном поле, получаемых с пяти геомагнитных обсерваторий, включенных в международную сеть ИНТЕРМАГНЕТ (INTERMAGNET – International Real-Time Magnetic Observatory Network) [0]. Это входящая в состав СПбФ ИЗМИРАН геомагнитная обсерватория «Санкт-Петербург» (международный IAGA-код SPG), геомагнитная обсерватория Nurmijarvi (Финляндия, Финский метеорологический институт, международный IAGA-код NUR), геомагнитная обсерватория Sodankyla (Финляндия, авроральная обсерватория, международный IAGA-код SOD), геомагнитная обсерватория Lycksele (Швеция, Геологическая служба Швеции, международный IAGA-код

ЛУС), геомагнитная обсерватория Abisko (Швеция, Геологическая служба Швеции, международный IAGA-код АВК). Необходимую информацию об этих источниках геофизических данных можно получить на сайте <https://www.intermagnet.org>. Выбор этих пунктов сбора обусловлен обстоятельством их расположения на достаточно небольших расстояниях друг от друга.

Анализ получаемых с этих геомагнитных обсерваторий данных о геомагнитном поле, помогает оперативно моделировать элементы магнитного поля Земли, что является очень важным фактором для решения большого класса фундаментальных и практических задач.

2. ПОЛУЧЕНИЕ И ПРЕДСТАВЛЕНИЕ ИСХОДНЫХ ДАННЫХ

Данные, предоставляемые обсерваториями, можно получить на сайте <https://www.intermagnet.org/index-eng.php>. Затем их необходимо поместить любым удобным для пользователя способом в директорию, с заданным статусом Current Folder на время решения данной задачи в MATLAB. Файлы с данными имеют большие размеры, и, следовательно, могут во время сеанса не поместиться в память компьютера. Поэтому были применены методы работы с Big Data, а именно механизм Datastore [0].

В данной работе были выбраны данные за период с 1 января 2018 г. по 28 февраля 2022 г. Их графики для каждой геомагнитной обсерватории представлены на рис. 1–5. На графиках значения представлены в нанотеслах x , y , z – компонент и полного вектора F магнитного поля Земли.

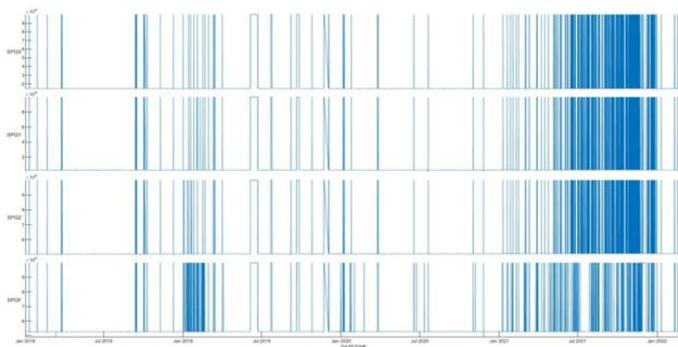


Рис. 1. Исходные данные SPG с 01.01.2018 г. по 28.02.2022 г.

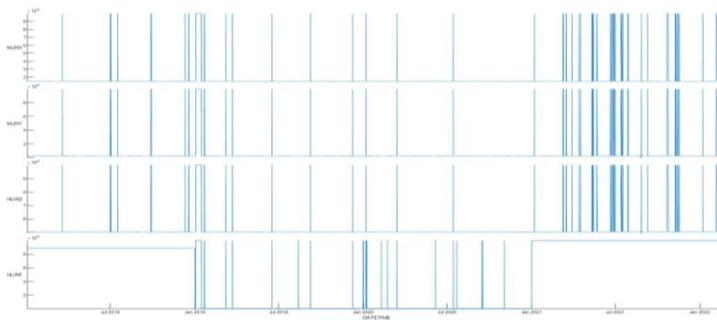


Рис. 2. Исходные данные NUR с 01.01.2018 г. по 28.02.2022 г.

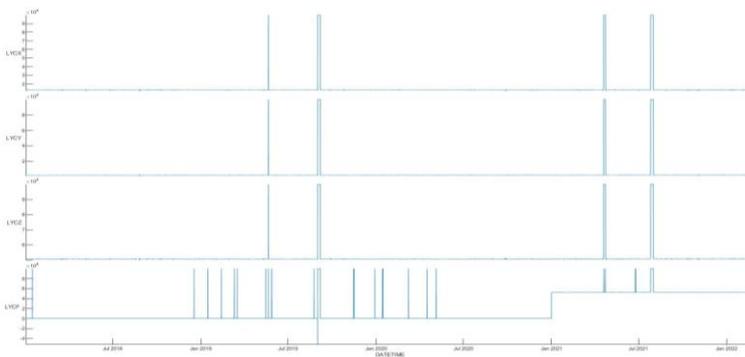


Рис. 3. Исходные данные LYC с 01.01.2018 г. по 28.02.2022 г.

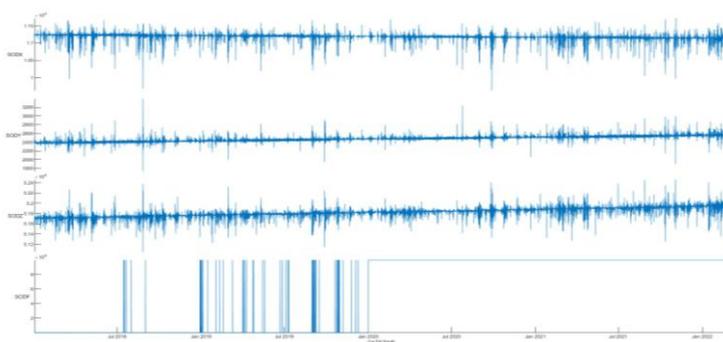


Рис. 4. Исходные данные SOD с 01.01.2018 г. по 28.02.2022 г.

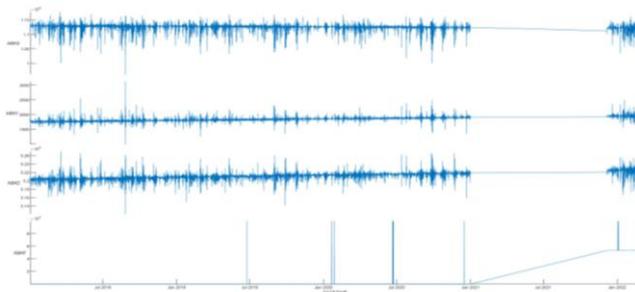


Рис. 5. Исходные данные АВК с 01.01.2018 г. по 28.02.2022 г.

Исходя из анализа был выбран общий интервал наблюдения, характеризующийся данными без выбросов: 1–14 марта 2020 г. На рис. 6–10 представлены исходные данные с SPG, NUR, LYC, SOD и АВК соответственно за выбранный период.

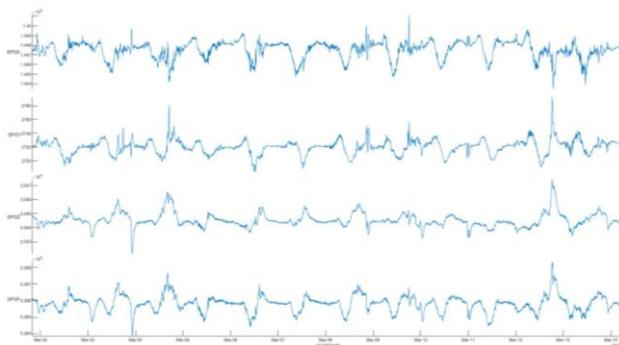


Рис. 6. Исходные данные SPG с 01.03.2020 г. по 14.03.2020 г.

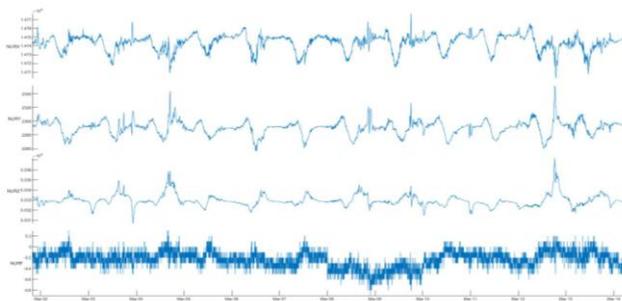


Рис. 7. Исходные данные NUR с 01.03.2020 г. по 14.03.2020 г.

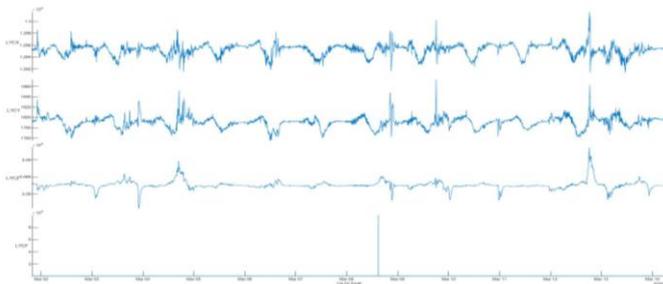


Рис. 8. Исходные данные LYC с 01.03.2020 г. по 14.03.2020 г.

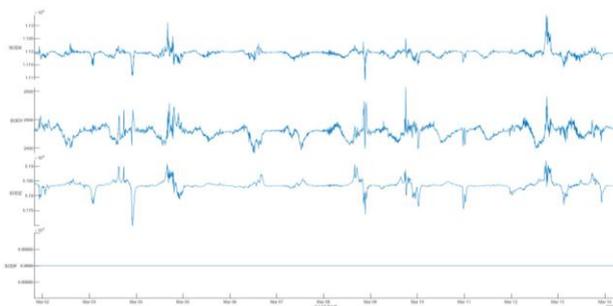


Рис. 9. Исходные данные SOD с 01.03.2020 г. по 14.03.2020 г.

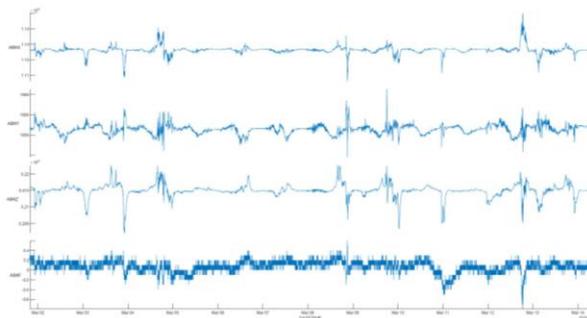


Рис. 10. Исходные данные AVK с 01.03.2020 г. по 14.03.2020 г.

3. ЗАКЛЮЧЕНИЕ

Исходя из результатов можно сделать следующие выводы.

Геомагнитная обсерватория «Санкт-Петербург» (международный IAGA-код SPG) в промежутке наблюдений за геомагнитным полем с 01.01.2018 – 28.02.2022 г. предоставляет наименее качествен-

ные данные. Особенно это заметно в конце промежутка наблюдений. Но необходимо отметить, что данные АВК отсутствуют практически с 2020 г.

Геомагнитная обсерватория Sodankyla (Финляндия, авроральная обсерватория, международный IAGA-код SOD) предоставляет наиболее качественные данные. Но необходимо отметить, что с начала наблюдений до января 2019 г и с января 2021 г данные о измерении полного поля отсутствуют.

Выбранный промежуток наблюдений, где отсутствуют “выбросы” показал, что данные со всех обсерваторий практически одинаковы для x , y , z – компонент. А вот данные измерения полного поля на станции, полученные в геомагнитных обсерваториях NUR и АВК нуждаются в предварительной обработке (данные АВК отсутствуют).

Литература

1. Макшанов А.В., Журавлев А.Е., Тындыкарь Л.Н. Большие данные. Big Data. – СПб.: Лань, 2021. – 188 с.
2. Гилат А. MATLAB. Теория и практика. – 5-е изд. – М.: ДМК Пресс, 2016. – 416 с.
3. Гришенцев А.Ю., Коробейников А.Г. Разработка модели решения обратной задачи вертикального зондирования ионосферы // Научно-технический вестник информационных технологий, механики и оптики. – 2011. – № 2 (72). – С. 109-113.
4. Гришенцев А.Ю., Коробейников А.Г. Средства интероперабельности в распределенных геоинформационных системах // Журнал Радиоэлектроники. – 2015. – № 3. – С. 1-16.
5. Гришенцев А.Ю., Коробейников А.Г. Декомпозиция n -мерных цифровых сигналов по базису прямоугольных всплесков // Научно-технический вестник информационных технологий, механики и оптики. – 2012. – № 4 (80). – С. 75-79.
6. Коробейников А.Г. Проектирование математической модели системы автоматического управления магнитной левитации // Международный журнал гуманитарных и естественных наук. – 2021. – № 12-2 (63). – С. 15-26.
7. Korobeinikov A.G., Grishentsev A.Y., Velichko E.N., Aleksanin S.A., Fedosovskii M.E., Bondarenko I.B., Korikov C.C. Calculation of Regularization Parameter in the Problem of Blur Removal in Digital Image // Optical Memory & Neural Networks (Information Optics). – 2016. – Vol. 25, No. 3. – P. 184-191.
8. Коробейников А.Г., Кутузов И.М. Алгоритм обфускации // Кибернетика и программирование. – 2013. – № 3. – С. 1-8.
9. Богатырев В.А. Надежность и эффективность резервированных компьютерных сетей // Информационные технологии. – 2006. – № 9. – С. 25-30.
10. Коробейников А.Г., Гришенцев А.Ю., Святкина М.Н. Применение интеллектуальных агентов магнитных измерений для мониторинга объектов железнодорожной инфраструктуры // Кибернетика и программирование. – 2013. – № 3. – С. 9-20.
11. Коробейников А.Г. Обработка и анализ данных с российского сегмента мировой сети магнитных обсерваторий ИНТЕРМАГНЕТ // Международный журнал гуманитарных и естественных наук. – 2018. – № 8. – С. 91-98.

АВТОМАТИЗИРОВАННЫЙ КОМПЛЕКС НЕРАЗРУШАЮЩЕГО КОНТРОЛЯ НА БАЗЕ МЕТОДОВ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА *

*Коробейников А.Г., д.т.н., профессор, зам. директора по науке
СПбФ ИЗМИРАН
профессор*

*Университет ИТМО
e-mail: Korobeynikov_A_G@mail.ru*

*Ткалич В.Л., д.т.н., профессор
Университет ИТМО*

e-mail: vera_leonidovna_tkalich@mail.ru

*Пирожникова О.И., к.т.н., доцент
Университет ИТМО*

e-mail: O.Pirozhnikova10@gmail.com

*Калинкина М.Е., магистрант
Университет ИТМО*

e-mail: mariia_kalinkina@mail.ru

1. ВВЕДЕНИЕ

Решение задач теории трещин, относящихся к так называемым «экстремальным задачам механики», в настоящее время приобретает все большую актуальность [0–0]. Оно способствует развитию методов неразрушающего контроля, в частности при дефектоскопии металлоизделий на предмет наличия в них скрытых дефектов [0–0].

Для решения вышеназванных задач был разработан специализированный комплекс с функциями искусственного интеллекта (ИИ), позволяющий проводить техническую экспертизу для надежного прогнозирования ресурса работы металлоконструкций, применяемых в сфере транспортной инфраструктуры. В статье представлены результаты, связанные с реализацией процесса адаптивного диагностирования субструктуры материала объекта мониторинга при проведении его дефектоскопии на базе решения задачи машинного обучения.

* Работа выполнена при финансовой поддержке гранта Президента Российской Федерации № МК-5323.2022.

2. КОМПЛЕКС НЕРАЗРУШАЮЩЕЙ ДИАГНОСТИКИ МЕТАЛЛОКОНСТРУКЦИЙ

Наиболее распространенными методами неразрушающего контроля (НК) являются ультразвуковой и акустико-эмиссионный. Трудности применения акустических методов связаны с тем, что акустический сигнал, распространяясь в объекте мониторинга, искажается. При этом распространяющиеся шумы и помехи оказывают воздействие на измерительные датчики. Сигналы от помех и шумов по форме схожи с сигналами от дефектообразований, что приводит к ложным срабатываниям. Поэтому важны требования, предъявляемые к помехозащищенности.

При цифровой обработке сигналов важно грамотно осуществить эффективное и достоверное выделение именно полезных составляющих сигнала, исключив помехи. В дальнейшем при обработке сигналов необходимо производить их преобразования для последующей возможной интерпретации.

Комплекс осуществляет анализ материала объекта мониторинга в зонах незащищенной металлической поверхности после сгенерированного искусственного виброакустического воздействия, с последующей передачей виброакустического отклика с объекта мониторинга на акселерометр (пьезоэлектрический преобразователь) и формированием на выходе датчика виброакустического сигнала, поступающего в блок обработки сигнала. В блоке обработки сигнала происходит оцифровка электрического колебания посредством преобразования в встроенном аналоговом преобразователе так называемой «звуковой карты» с помощью быстрого вейвлет-преобразования полученного аналогового сигнала во временной ряд с $2\div 3$ частотными поддиапазонами.

Функциональная блок-схема комплекса представлена на рис. 1. Она содержит источник виброакустического воздействия 1 и объект мониторинга (изделие из металла) 2, а также пьезоэлектрический акселерометр 3, блок обработки сигнала 4, блок адаптивного принятия решения (на основе методов ИИ) 5, и блок оценки параметров модели состояния объекта мониторинга 6, который может передавать адаптивное зондирующее воздействие на источник виброакустического воздействия 1 посредством обратной связи.

Программное обеспечение (ПО) комплекса функционирует под управлением операционной среды WINDOWS 10 и реализовано в среде MATLAB R2021b, которая позволяет решать разнообразные задачи в различных предметных областях [0-0]. ПО позволяет осуществлять анализ изображения электрических колебаний объекта мониторинга в виде спектров структурных колебаний.

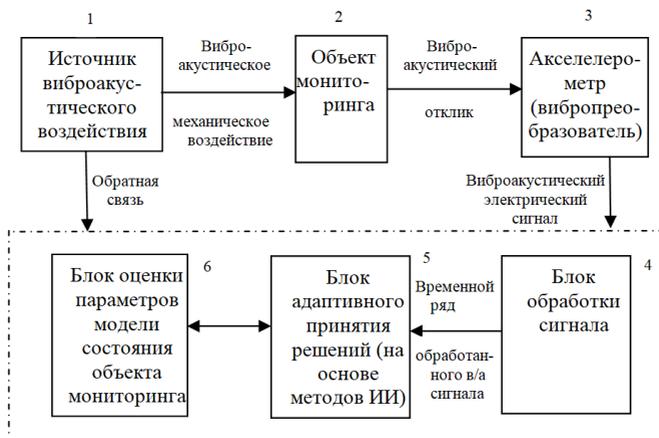


Рис. 1. Функциональная блок-схема комплекса неразрушающей диагностики металлоконструкций

Полученные спектры отражают по частоте имеющиеся изменения в форме структуры материала (металла) объекта мониторинга, а по амплитуде сигналов можно судить о фазовом составе этого материала.

Комплекс позволяет повысить точность, а также помехоустойчивость формы структуры объекта мониторинга. Он имеет расширенные функциональные возможности, что связано с наличием функций ИИ. Созданный комплекс позволяет осуществлять критериальный анализ в ходе выполнения НК металлоконструкции при: при прогнозировании долговечности и остаточного ресурса; при выявлении потенциально опасных дефектных зон; при обнаружении областей хрупкого разрушения; при выявлении изменений в зонах фазового состава в ходе проведения входного контроля.

В разработанном авторами комплексе источник виброакустического воздействия генерирует возникновение механических колебаний (волн акустического диапазона) в структуре объекта мониторинга (металлоконструкции). Разработанный и запатентованный пьезоэлектрический акселерометр с магнитным держателем [0, 0] связан с объектом мониторинга. Функция пьезоэлектрического акселерометра на магнитном держателе заключается в преобразовании, порождаемых источником, акустических сигналов в подлежащие анализу электрические колебания. Блок обработки сигнала может быть связан с пьезоэлектрическим вибропреобразованием (акселерометром) посредством микрофонного входа с использованием соединительного кабеля.

Электрические колебания отображаются на экране в реальном масштабе времени, что помимо функции наглядной визуализации позволяет определять источник возмущения в структуре материала объекта мониторинга. Далее осуществляется определение местоположения максимальных дефектных зон на всей длине объекта мониторинга.

Возможна за счёт обратной связи и корректировка сигнала от источника воздействия (адаптивное зондирующее воздействие) по уровню сформированных электрических колебаний на выходе пьезоэлектрического акселерометра. Комплекс позволяет проводить оцифровку электрических колебаний. Спектральный анализ оцифрованных акустических сигналов осуществляется с использованием функции Хэмминга и посредством быстрого вейвлет-преобразования Маллата, что позволяет достичь максимальной локализации функции по частоте, а также максимального возможного снижения корреляционной зависимости отрезков спектров частотных диапазонах.

В ходе исследований авторами результаты компьютерной интерпретации обработки данных от сигналов с объекта мониторинга (рис. 2).

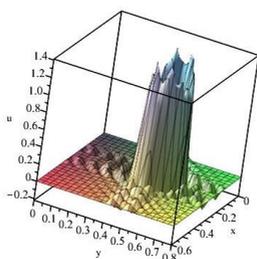


Рис. 2. Результаты компьютерной интерпретации обработки данных от сигналов с объекта мониторинга

3. ЗАКЛЮЧЕНИЕ

Создание новых или усовершенствованных средств для осуществления неразрушающего контроля, а также методики, базирующейся на методах ИИ, является важной целью современных разработок в области модификации методов НК.

Вейвлет-преобразование является перспективным для использования при анализе нестационарного сигнала.

Литература

1. BSI. Guide to Methods for Assessing the Acceptability of Flaws in Metallic Structures, British Standards Institute, Tech. Rep. BS 7910:2013+A1:2015, 2015. [Online]. Available: <https://doi.org/10.3403/30241230>.

2. Лурье А.И. Теория упругости. – М.: Наука, 1970. – 940 с.
3. Newman J.C., Raju I.S. An empirical stress-intensity factor equation for the surface crack // *Engineering Fracture Mechanics*. – Vol. 15, No. 1. – P. 185-192.
4. Bowness D., Lee M.M.K. Prediction of weld toe magnification factors for semi-elliptical cracks in Tbutt joints // *International Journal of Fatigue*. – 2000. – Vol. 22, No. 5. – P. 369-387.
5. Морозов Н.Ф. Математические вопросы теории трещин. – М.: Наука, 1984.
6. Korobeynikov A.G., Fedosovsky M.E., Maltseva N.K., Baranova O.V., Zharinov I.O., Gurjanov A.V., Zharinov O.O. Use of information technologies in design and production activities of instrument-making plants // *Indian Journal of Science and Technology*. – 2016. – Vol. 9, No. 44. – P. 104708.
7. Korobeynikov A.G., Grishentsev A.Y., Velichko E.N., Aleksanin S.A., Fedosovskii M.E., Bondarenko I.B., Korikov C.C. Calculation Of Regularization Parameter In The Problem Of Blur Removal In Digital Image // *Optical Memory & Neural Networks (Information Optics)*. – 2016. – Vol. 25, No. 3. – P. 184-191.
8. Грищенко А.Ю., Коробейников А.Г. Средства интероперабельности в распределенных геоинформационных системах // *Журнал Радиоэлектроники*. – 2015. – № 3. – С. 1-19.
9. Богатырев В.А, Богатырев С.В. Своевременность обслуживания в многоуровневых кластерных системах с поэтапным уничтожением просроченных запросов // *Вестник компьютерных и информационных технологий*. – 2018. – № 2 (164). – С. 28-35.
10. Коробейников А.Г. Применение MATLAB для работы с большими данными на примере обработки информации из мировой сети магнитных обсерваторий INTERMAGNET // Сборник трудов X Междунар. науч.-техн. конф. «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2021). – Смоленск, 2021. – С. 388-394.
11. Исмагилов В.С., Коробейников А.Г. Применение методов искусственного интеллекта при обработке геофизических данных, связанных с обнаружением берегового эффекта в естественных вариациях электрического поля // Сборник трудов X Междунар. науч.-техн. конф. «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2021). – Смоленск, 2021. – С. 395-402.
12. Ткалич В.Л., Бибиков С.В., Калинкина М.Е., Пирожникова О.И., Марусина М.Я. Пьезоэлектрический акселерометр. Патент на полезную модель № 202246 от 09.02.2021.
13. Ткалич В.Л., Бибиков С.В., Калинкина М.Е., Пирожникова О.И., Марусина М.Я. Магнитный держатель пьезоэлектрического виброобразователя. Патент на полезную модель № 201522 от 21.12.2020.

ГИБРИДНЫЙ АЛГОРИТМ ПОИСКА АНОМАЛИЙ ВРЕМЕННЫХ РЯДОВ С УЧЕТОМ ОСОБЕННОСТЕЙ ПРЕДМЕТНОЙ ОБЛАСТИ*

Курило Д.С., ассистент

Ульяновский государственный технический университет

e-mail: dimokkurilo@gmail.com

Мошкин В.С., к.т.н., доцент

Ульяновский государственный технический университет

e-mail: v.moshkin@ulstu.ru

Мошкина И.А., к.т.н., доцент

e-mail: i.timina@ulstu.ru

Ульяновский государственный технический университет

1. ВВЕДЕНИЕ

Выявление аномалий – область интеллектуального анализа данных, позволяющая находить выделяющиеся из общей массы значения. Эти значения необходимы во многих предметных областях и могут говорить о различных проблемах, сбоях или данных, на которые стоит обратить внимание.

Для поиска используются различные технологии, такие как кластерный анализ, скрытые марковские модели, нейронные сети и так далее. Они работают с различной эффективностью, зависящей от затрагиваемой предметной области [1, 2].

2. ЗАДАЧА ПОИСКА АНОМАЛИЙ

Для поиска аномалий используется три основных подхода.

Подход к распознаванию аномалий без учителя используется в тех случаях, если по условиям задачи неизвестен характер временных рядов и неопределенно наличие аномалий. В этом случае алгоритм базируется на предположении, что аномальные значения встречаются реже нормальных.

Подход к распознаванию аномалий частично с учителем предполагает наличие обучающего множества, в котором определены только нормальные объекты (не аномальные). В данном случае алгоритм предполагает, что объекты, которые не относятся к классу нормальных, являются аномальными.

* Работа выполнена при финансовой поддержке РФФИ (проект № 20-07-00672).

Подход к распознаванию аномалий с учителем предполагает наличие полностью размеченного обучающего множества на класс нормальных объектов и аномальных объектов. В большинстве случаев предполагается, что данные не меняют свои статистические характеристики, так как возникает необходимость изменять классификатор. Главной сложностью применения этих подходов является необходимость формирования данных для обучения, особенно актуально это при обучении классов аномальных данных [3–5].

Для использования онтологических методов в поиске аномалий предполагается создание набора продукционных правил, определяющих нормальное поведение экземпляра временного ряда профиля. Объект, не соответствующий правилам, является аномальным.

3. АЛГОРИТМ ПОИСКА АНОМАЛИЙ

Преимуществом предлагаемого решения по поиску аномалий во временных рядах является гибридизация алгоритмов машинного обучения с семантическими алгоритмами, позволяющими провести дополнительную фильтрацию результатов анализа посредством учета особенностей предметной области используемых временных рядов. Схема разработанного алгоритма представлена на рис. 1.



Рис. 1. Блок-схема алгоритма

Условными переходами в алгоритме является проверка значения на аномальность на этапе анализа данных нейронной сетью и на этапе работы с онтологией при прохождении предыдущего условия. Вне зависимости от исхода проверок данные используются для логического вывода, завершающего работу алгоритма.

Нейронная сеть включает в себя следующие слои:

- Conv1D – свёрточный слой для глубокого обучения;
- Conv1DTranspose – транспонированный свёрточный слой;
- Dropout – слой сброса для предотвращения переобучения.

Обобщенная структура нейронной сети показана на рис. 2.

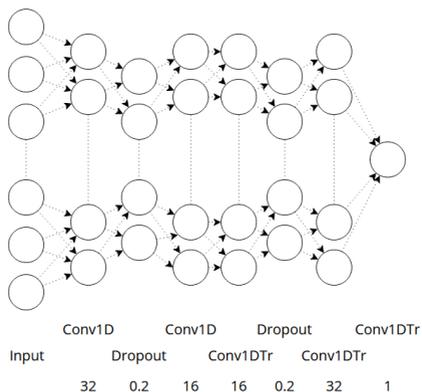


Рис. 2. Структура нейронной сети

4. ПРИМЕНЕНИЕ АЛГОРИТМА

В качестве экспериментальных данных использовались данные производительности нефтяных вышек. Данные разбиваются на несколько независимых друг от друга выборок по имени вышки, на которой происходили замеры. Это позволяет иметь несколько массивов данных, подходящих для последующих экспериментов. Полученные данные при необходимости возможно отобразить на графике. Готовые данные передаются в модуль нейронной сети.

Для данной предметной области была разработана OWL-онтология. Структура онтологии включает в себя несколько классов, показанных на рис. 3.

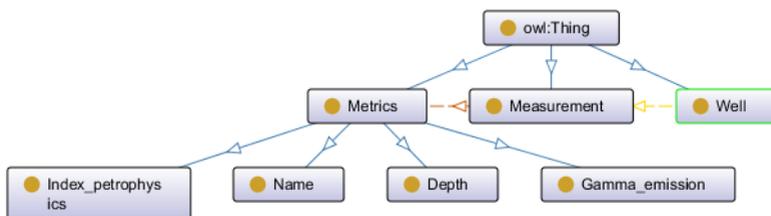


Рис. 3. Структура онтологии

Класс Well – характеризует реальный объект, «привязывающий» измерения к предметной области. Measurement – класс, показывающий, что значение является замером и частью временного ряда измерений. Класс Metrics показывает связь между замером и каким-либо показателем предметной области.

Так же онтология хранит SWRL правила, по которым происходит проверка аномалий на корректность. Пример такого правила представлен на рис. 4. Оно использует значение временного ряда и делает вывод по параметру, не используемому в нейронной сети.

```

hasGR(?x, ?gr) ^ Well(?x) ^ hasDeltaPHI(?x, ?deltaphi) ^
swrlb:lessThan(?deltaphi, 2.0) -> hasAnomaly(?x, true)
  
```

Рис. 4. Пример SWRL-правила

Правила содержат атомы класса и несколько необходимых свойств: аномального значения и дополнительного параметра, по которому подтверждается корректность суждения об аномальности [8].

Цель применения правила состоит в проверке достижения параметром константы. Если значение не достигнуто, то замер может считаться аномальным, иначе значение корректно. Первым шагом работы семантической части алгоритма является очистка всех уже существующих индивидов в онтологии. Каждый запуск загружает все найденные нейронной сетью аномалии в онтологию, а после работы правил выгружает все имеющиеся в онтологии значения.

Для загрузки индивидов в онтологию используется класс и свойства, уже описанные в структуре онтологии. После загрузки всех аномалий, происходит пересохранение онтологии и запуск правил. После обработки из файла можно получить все индивиды. В цикле проверяется флаг аномальности и отбираются все значения, принятые онтологией за действительно аномальные (рис. 5).

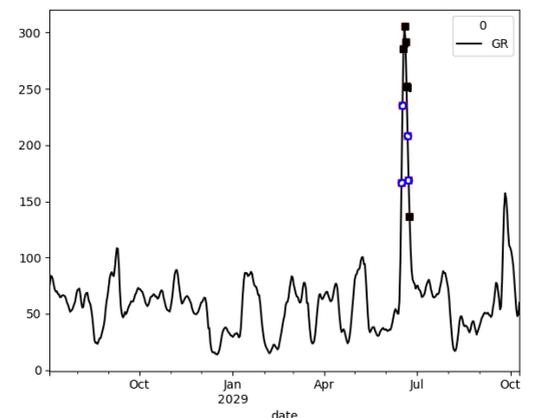


Рис. 5. Результаты работы системы

Темные маркеры – это аномалии, найденные с помощью нейронной сети, но не получившие подтверждения в онтологии. Светлые маркеры – это значения, подтвержденные правилами онтологии и действительно являющиеся аномалиями.

5. ЗАКЛЮЧЕНИЕ

Разработанный алгоритм обрабатывает исходные данные временного ряда и выдаёт набор аномальных значений. Предлагаемый подход использует технологии нейронных сетей и онтологический анализ. Нейронная сеть находит аномалии с помощью построения предсказания, на основании обучающей выборки и сравнения с тестовыми данными. Онтология использует заранее разработанный класс и SWRL правила.

В дальнейшем планируется усовершенствовать архитектуру нейронной сети для получения наилучших результатов. Также предполагается дополнить онтологию классами метрик, что позволит оперировать переменными, не привязанными к конкретным параметрам. Вследствие этого, развитие подхода будет заключаться только в добавлении новых классов метрик и написании новых правил.

Литература

1. Линдигрин А.Н. Анализ специфики и проблематики процессов поиска аномалий в сетевых данных // Известия ТулГУ. Технические науки. – 2021. – № 5. – С. 304-309.
2. Гасанов В.И. Выявление аномалий в сетевом трафике на основе нейросетевого моделирования динамики изменения объемов IP-пакетов // ММС. – 2018. – № 2. – С. 40-45.

3. Зуев В.Н. Обнаружение аномалий сетевого трафика методом глубокого обучения // Программные продукты и системы. – 2021. – № 1. – С. 91-97.
4. Wen T., Keyes R. Time series anomaly detection using convolutional neural networks and transfer learning //arXiv preprint arXiv: 1905.13628. 2019.
5. Zhong C., Gursoy M.C., Velipasalar S. Anomaly Detection and Sampling Cost Control via Hierarchical GANs // GLOBECOM 2020-2020 IEEE Global Communications Conference. – IEEE, 2020. – P. 1-6.
6. Мошкин В.С., Ярушкина Н.Г. Система онтологического анализа временных рядов // Автоматизация процессов управления. – 2014. – № 2 (36). – С. 78-85.
7. Malhotra P., Vig L., Shrof, G.M., Agarwal P. Long Short Term Memory Networks for Anomaly Detection in Time Series // Proc. of the 23rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN. ESANN. 2015. – P 89-94.
8. Мошкин, В.С., Ярушкина, Н.Г. Особенности интеграции механизмов логического вывода в онтологическую модель представления знаний с помощью SWRL-правил // Труды XIV Национальной конференции по искусственному интеллекту с международным участием КИИ-2014: Т. 1. – Казань: РИЦ «Школа», 2014. – С. 173-181.

СОСТОЯТЕЛЬНАЯ И ПОЛНАЯ СИСТЕМА ДЕДУКЦИИ ДЛЯ ПРОПОЗИЦИОНАЛЬНОЙ НЕЧЕТКОЙ ЛОГИКИ С ОЦЕНКАМИ*

*Плесневич Г.С., к.ф.-м.н., профессор
НИУ «МЭИ»
e-mail: salve777@mail.ru*

1. ВВЕДЕНИЕ. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

Пусть \mathbf{L} обозначает нечеткую пропозициональную логику Заде с множеством \mathbf{P} пропозициональных переменных. Ее синтаксис совпадает с синтаксисом классической пропозициональной логики. формулы.

Формулы логики \mathbf{L} .

φ – формула логики $\mathbf{L} \Leftrightarrow \varphi \in \mathbf{P}$ или φ имеет один из видов $\sim \varphi_1$, $(\varphi_1 \wedge \varphi_2)$, $(\varphi_1 \vee \varphi_2)$, $(\varphi_1 \rightarrow \varphi_2)$, где φ_1 и φ_2 – формулы логики \mathbf{L} .

Семантика логики \mathbf{L} определяются нечеткими интерпретациями формул, Такие интерпретации являются продолжениями нечетких интерпретаций пропозициональных переменных. Интерпретация пропозициональных переменных есть просто функция вида “ $_$ ”: $\mathbf{P} \mapsto [0,1]$ (где $[0,1]$ – единичный интервал чисел $\{x / 0 \leq x \leq 1\}$). Нечеткие интерпретации произвольных формул логики \mathbf{L} определяются индуктивно исходя из значений “ p ” ($p \in \mathbf{P}$).

Реально применяя нечеткую логику, мы берем истинностные значения $a \in [0,1]$ с определенной точностью. Например, можно взять целое число h и обозначить $[0,1]_h$ множество всех чисел из интервала $[0,1]$, записанных в десятичной системе счисления как дроби с h разрядами:

$$[0,1]_h = \{0.x_1x_2\dots x_h \mid x_i \in \{0,1,2,\dots,9\}, 1 \leq i \leq h\}.$$

Зафиксировав параметр h , мы получаем обозначаемую \mathbf{L}_h нечеткую пропозициональную логику, которая имеет элементы множества $[0,1]_h$ в качестве констант, обозначающих истинностные значения.

* Работа выполнена при финансовой поддержке РФФИ, проекты № 20-07-00615 А и № 20-57-00015 Бел_а.

Другими словами, \mathbf{L}_h – многозначная логика с множеством $[0,1]_h$ из 2^h истинностных значений, начиная с $0.00\dots 0$ (представляет «ложь» 0) и кончая $0.99\dots 9$ (представляет «истина» 1). Для данного числа $z \in [0,1]_h$ ($z \neq 0$) обозначим z^- предыдущее значение (например, $0.013^- = 0.012$ при $h = 3$).

Таким образом, многозначная логика \mathbf{L}_h имеет синтаксис, определяемый такими же правилами, как и логика Заде \mathbf{L} . Семантика логики \mathbf{L} (\mathbf{L}_h) определяется индуктивно правилами, указанными в следующей рамке.

Нечеткие интерпретации формул логик.

$$\begin{aligned} \sim \varphi &= \sim \text{“}\varphi\text{”} = 1 - \text{“}\varphi\text{”}, \\ \text{“}(\varphi \wedge \psi)\text{”} &= \text{“}\varphi\text{”} \wedge \text{“}\psi\text{”} = \min\{\text{“}\varphi\text{”}, \text{“}\psi\text{”}\}, \\ \text{“}(\varphi \vee \psi)\text{”} &= \text{“}\varphi\text{”} \vee \text{“}\psi\text{”} = \max\{\text{“}\varphi\text{”}, \text{“}\psi\text{”}\}, \\ \text{“}(\varphi \rightarrow \psi)\text{”} &= \text{“}\varphi\text{”} \rightarrow \text{“}\psi\text{”} = \max\{1 - \text{“}\varphi\text{”}, \text{“}\psi\text{”}\}, \end{aligned}$$

где φ, ψ – формулы логики \mathbf{L} (\mathbf{L}_h).

Мы будем рассматривать оценки для формул нечетких логик \mathbf{L} (\mathbf{L}_h) как выражения со следующим синтаксисом.

Простые оценки

$$(\varphi \geq a), (\varphi > a), (\varphi \leq a), (\varphi < a), \text{ где } \varphi - \text{формула логики } \mathbf{L} (\mathbf{L}_h) \text{ и } a \in [0,1] ([0,1]_h)$$

В частности, атомарные оценки ($p \geq a$), ($p > a$), ($p \leq a$), ($p < a$) являются простыми. Произвольные оценки строятся индуктивно исходя из простых оценок.

Произвольные оценки для формул логики Заде \mathbf{L} и \mathbf{L}_h

$$\begin{aligned} \lambda &- \text{произвольная оценка} \Leftrightarrow \\ \lambda &- \text{простая оценка или } \lambda \text{ имеет один из видов} \\ &\sim \lambda_1, (\lambda_1 \wedge \lambda_2), (\lambda_1 \vee \lambda_2), (\lambda_1 \rightarrow \lambda_2), (\varphi_1 \leq \varphi_2), (\varphi_1 < \varphi_2), \\ &\text{где } \lambda_1 \text{ и } \lambda_2 - \text{произвольные оценки, а } \varphi_1 \text{ и } \varphi_2 - \\ &\text{произвольные формулы } \mathbf{L} (\mathbf{L}_h). \end{aligned}$$

Вот примеры формул и оценок:

$$(p_2 \vee \sim (p_1 \rightarrow \sim (p_2 \sim (p_3 \wedge p_1)))) - \text{формула логики } \mathbf{L} \mathbf{L} (\mathbf{L}_h),$$

$$((p_2 \vee \sim (p_1 \rightarrow \sim (p_2 \sim (p_3 \wedge p_1)))) \leq 0.3) - \text{простая оценка,}$$

$$(((\sim (p_3 \vee \sim p_1)) \geq 0.5) \rightarrow ((p_2 \leq \wedge \sim (p_1 \leq 0.7))) - \text{оценка,}$$

$$(\sim (p_1 \leq p_3) \vee \sim (p_3 \leq p_3)) \rightarrow ((p_2 \wedge \sim p_1) \leq 0.6) - \text{оценка.}$$

Семантика оценок определяется интерпретациями, которые являются продолжениями интерпретаций “ \sim ” формул логик \mathbf{L} (\mathbf{L}_h). Заметим, что продолженные интерпретации являются четкими значениями “ α ” оценок α служат 0 (ложь) и 1 (истина).

Интерпретация простых оценок

$$\begin{aligned} &“(\varphi \geq a)”=1 \Leftrightarrow “\varphi” \geq a, “(\varphi > a)”=1 \Leftrightarrow “\varphi” > a, \\ &“(\varphi \leq a)”=1 \Leftrightarrow “\varphi” \leq a, “(\varphi < a)”=1 \Leftrightarrow “\varphi” < a, \\ &\text{где } \varphi \text{ – формула } \mathbf{L}(\mathbf{L}_h) \text{ и } a \in [0,1] ([0,1]_h). \end{aligned}$$

Интерпретация произвольных оценок

$$\begin{aligned} &\sim\lambda” = \sim “\lambda” = 1 - “\lambda”, \\ &“(\lambda \wedge \mu)” = “\lambda” \wedge “\mu” = \min\{“\lambda”, “\mu”\}, \\ &“(\lambda \vee \mu)” = “\lambda” \vee “\mu” = \max\{“\lambda”, “\mu”\}, \\ &“(\lambda \rightarrow \mu)” = “\lambda” \rightarrow “\mu” = \max\{1 - “\lambda”, “\mu”\}, \\ &“(\varphi_1 \leq \varphi_2)” = 1 \Leftrightarrow “\varphi_1” \leq “\varphi_2”, \\ &“(\varphi_1 < \varphi_2)” = 1 \Leftrightarrow “\varphi_1” < “\varphi_2”, \\ &\text{где } \lambda \text{ и } \mu \text{ – оценки, а } \varphi_1, \varphi_2 \text{ – формулы} \\ &\text{логик } \mathbf{L}(\mathbf{L}_h). \end{aligned}$$

Будем рассматривать оценки для логик $\mathbf{L}(\mathbf{L}_h)$ в качестве предложений пропозициональной логики, которую назовем *логикой оценок* и обозначим $\mathbf{LE}(\mathbf{LE}_h)$.

Логика $\mathbf{LE}(\mathbf{LE}_h)$ индуцирует отношение логического следования между оценками, обозначаемое \models . Для данных множества оценок E и оценки λ по определению имеет место логическое следование $E \models \lambda$, если не существует интерпретации, при которой истинны все оценки из E , но ложна оценка λ .

С понятием логического следования непосредственно связано понятие логической противоречивости, или невыполнимости. Множество E предложений логики называется выполнимым, если существует интерпретация, при которой истинны все предложения из E . Ясно, что из E логически следует оценка λ тогда и только тогда, когда невыполнимо множество E с добавленным к нему отрицанием оценки λ , т.е.

$$E \models \lambda \Leftrightarrow (E \cup \{\sim\lambda\} \text{ невыполнимо}).$$

Таким образом, задача логических следствий может быть сведена к задаче распознавания невыполнимости множеств оценок.

Для логики оценок $\mathbf{LE}(\mathbf{LE}_h)$ имеется также отношение *оптимального логического следования*, обозначаемое \models^* . Это отношение действует между любым множеством оценок E и любой простой оценкой $\varphi \geq a$. По определению имеем $E \models^* (\varphi \geq a)$ тогда и только тогда, когда $E \models (\varphi \geq a)$ и не верно, что $E \models (\varphi \geq b)$, если $b < a$. Отношение логического следствия используется в методе вычисления ответов на запросы, адресуемые к представленным оценками базам знаний.

Важный мотив для введения логик оценок заключается в следующем. Применяя нечеткие логики при построении реальных предложений, эксперты имеют трудности в точном назначении истинност-

ных значений нечетким утверждениям, тогда как при применении логики оценок мы можем обойтись только границами для истинностных значений, и затем манипулировать этими границами.

Ранее S. Kundu рассматривал простые оценки утверждений логики Заде, для которых он построил состоятельную и полную систему логического вывода, основанную на принципе резолюции [2, 4]. В настоящей работе мы представим состоятельную и полную систему логического вывода Σ , основанную на аналитических таблицах для логики оценок (Правила системы Σ даны в табл. 1, табл. 2 (см. в [1, 3] изложение методов логического вывода, использующего аналитические таблицы.)

Правила системы Σ даны в табл. 1, табл. 2, табл. 3. Обозначим \vdash_{Σ} отношение логического вывода, определяемое следующим образом. Пусть E – множество оценок и λ – оценка. Тогда полагаем $E \vdash_{\Sigma} \lambda$, если существует вывод, полученный из множества E путем применения правил системы Σ . Отношение \vdash_{Σ} называют отношением *доказуемости* в системе Σ .

Мы докажем, что система Σ состоятельна и полна. Ее состоятельность означает, что отношение \vdash_{Σ} влечет отношение \models (т.е. $E \vdash_{\Sigma} \lambda \Rightarrow E \models \lambda$). Ее полнота означает, что отношение $E \models \lambda$ влечет отношение $E \vdash_{\Sigma} \lambda$ (т.е. $E \models \lambda \Rightarrow E \vdash_{\Sigma} \lambda$). Другими словами, экстенционально отношения доказуемости и логического следования совпадают.

2. ЭКВИВАЛЕНТНОСТИ В ЛОГИКЕ ОЦЕНОК

Оценки λ и μ *эквивалентны*, если при любой интерпретации они имеют одинаковые истинностные значения:

$$\lambda \equiv \mu \Leftrightarrow_{df} \text{“}\lambda\text{”} = \text{“}\mu\text{” для любой интерпретации “}__”.$$

В следующей лемме дан список эквивалентностей, используемых при доказательстве состоятельности и полноты системы Σ . В этой лемме φ – формула логики \mathbf{L} (\mathbf{L}_h), а $a \in [0,1]$ ($[0,1]_h$).

Лемма 1. В логике оценок \mathbf{LE} справедливы следующие эквивалентности:

$$\begin{aligned} \sim (X \geq a) &\equiv (X < a), \\ \sim (X > a) &\equiv (X \leq a), \\ \sim (X \leq a) &\equiv (X > a), \\ \sim (X < a) &\equiv (\varphi \geq a), \\ (\sim \varphi \geq a) &\equiv (\varphi \leq 1 - a), \\ (\sim \varphi > a) &\equiv (\varphi < 1 - a), \\ (\sim \varphi \leq a) &\equiv (\varphi \geq 1 - a), \\ (\sim \varphi < a) &\equiv (\varphi > 1 - a), \\ ((\varphi \wedge \psi) \geq a) &\equiv ((\varphi \geq a) \wedge (\psi \geq a)), \end{aligned} \tag{1}$$

$$\tag{2}$$

$$\begin{aligned} ((\varphi \wedge \psi) > a) &\equiv ((\varphi > a) \wedge (\psi > a)), \\ ((\varphi \wedge \psi) \leq a) &\equiv ((\varphi \leq a) \vee (\psi \leq a)), \end{aligned} \quad (3)$$

$$\begin{aligned} ((\varphi \wedge \psi) < a) &\equiv ((\varphi < a) \vee (\psi < a)), \\ ((\varphi \vee \psi) \geq a) &\equiv ((\varphi \geq a) \vee (\psi \geq a)), \end{aligned} \quad (4)$$

$$\begin{aligned} ((\varphi \vee \psi) > a) &\equiv ((\varphi > a) \vee (\psi > a)), \\ ((\varphi \vee \psi) \leq a) &\equiv ((\varphi \geq a) \wedge (\psi \geq a)), \end{aligned} \quad (5)$$

$$\begin{aligned} ((\varphi \vee \psi) < a) &\equiv ((\varphi > a) \wedge (\psi > a)), \\ ((\varphi \rightarrow \psi) \geq a) &\equiv ((\varphi \leq 1-a) \vee (\psi \geq a)), \end{aligned} \quad (6)$$

$$\begin{aligned} ((\varphi \rightarrow \psi) > a) &\equiv ((\varphi < 1-a) \vee (\psi > a)), \\ ((\varphi \rightarrow \psi) \leq a) &\equiv ((\varphi \geq a) \wedge (\psi \geq a)), \end{aligned} \quad (7)$$

$$((\varphi \rightarrow \psi) < a) \equiv ((\varphi > a) \wedge (\psi > a)).$$

Первые 4 эквивалентности очевидны. Мы докажем только эквивалентности, содержащие знаки \leq и \geq . Доказательства эквивалентностей со знаками $<$ и $>$ получается просто заменой знаков \leq и \geq на знаки $<$ и $>$ (соответственно). Эквивалентность $\varphi \equiv \psi$ имеет место тогда и только тогда, когда справедлива эквивалентность “ φ ”=1 \Leftrightarrow “ ψ ”=1.

Доказательство эквивалентности (1):

$$\begin{aligned} “(\sim \varphi \geq a)” = 1 &\Leftrightarrow “\sim \varphi” \geq a \\ &\Leftrightarrow 1 - “\varphi” \geq a \\ &\Leftrightarrow “\varphi” \leq 1 - a \\ &\Leftrightarrow “(\varphi \leq 1 - a)” = 1. \end{aligned}$$

Отсюда при произвольной интерпретации “ $_$ ” имеем $(\sim \varphi \geq a) = 1 \Leftrightarrow “(\varphi \leq 1 - a)” = 1$. Следовательно, эквивалентность (1) справедлива.

Доказательство эквивалентности (2):

$$\begin{aligned} “((\varphi \wedge \psi) \geq a)” = 1 &\Leftrightarrow “(\varphi \wedge \psi)” \geq a \\ &\Leftrightarrow (“\varphi” \wedge “\psi”) \geq a \\ &\Leftrightarrow \min\{“\varphi”, “\psi”\} \geq a \\ &\Leftrightarrow “\varphi” \geq a \text{ и } “\psi” \geq a \\ &\Leftrightarrow “(\varphi \geq a)” = 1 \text{ и } “(\psi \geq a)” = 1 \\ &\Leftrightarrow (“\varphi \geq a” \wedge “\psi \geq a”) = 1 \\ &\Leftrightarrow “((\varphi \geq a) \wedge (\psi \geq a))” = 1 \end{aligned}$$

Отсюда “ $((\varphi \wedge \psi) \geq a)$ ”=1 \Leftrightarrow $(\varphi \geq a) \wedge (\psi \geq a)$ ”=1 при произвольной интерпретации “ $_$ ”. Следовательно, эквивалентность (2) справедлива.

Доказательство эквивалентности (3):

$$\begin{aligned} “((\varphi \wedge \psi) \leq a)” = 1 &\Leftrightarrow “(\varphi \wedge \psi)” \leq a \\ &\Leftrightarrow (“\varphi” \wedge “\psi”) \leq a \\ &\Leftrightarrow \min\{“\varphi”, “\psi”\} \leq a \\ &\Leftrightarrow “\varphi” \leq a \text{ или } “\psi” \leq a \\ &\Leftrightarrow “\varphi \leq a” = 1 \text{ или } “\psi \leq a” = 1 \\ &\Leftrightarrow (“\varphi \leq a” \vee “\psi \leq a”) = 1 \\ &\Leftrightarrow “((\varphi \leq a) \vee (\psi \leq a))” = 1 \end{aligned}$$

Отсюда $((\varphi \wedge \psi) \leq a) = 1 \Leftrightarrow (\varphi \leq a) \vee (\psi \leq a) = 1$ при произвольной интерпретации “ $_$ ”. Следовательно, эквивалентность (3) справедлива.

Доказательство эквивалентности (4):

$$\begin{aligned} ((\varphi \vee \psi) \geq a) = 1 &\Leftrightarrow ((\varphi \vee \psi) \geq a) \\ &\Leftrightarrow (\varphi \vee \psi) \geq a \\ &\Leftrightarrow \max\{\varphi, \psi\} \geq a \\ &\Leftrightarrow \varphi \geq a \text{ или } \psi \geq a \\ &\Leftrightarrow \varphi \geq a = 1 \text{ или } \psi \geq a = 1 \\ &\Leftrightarrow (\varphi \geq a) \wedge (\psi \geq a) = 1 \\ &\Leftrightarrow ((\varphi \geq a) \wedge (\psi \geq a)) = 1 \end{aligned}$$

Отсюда $((\varphi \wedge \psi) \leq a) = 1 \Leftrightarrow (\varphi \geq a) \wedge (\psi \geq a) = 1$ произвольной интерпретации “ $_$ ”. Следовательно, эквивалентность (4) справедлива.

Доказательство эквивалентности (5):

$$\begin{aligned} (\varphi \vee \psi) \leq a &\equiv ((\varphi \leq a) \wedge (\psi \leq a)): \\ ((\varphi \vee \psi) \leq a) = 1 &\Leftrightarrow ((\varphi \vee \psi) \leq a) \\ &\Leftrightarrow (\varphi \vee \psi) \leq a \\ &\Leftrightarrow \max\{\varphi, \psi\} \leq a \\ &\Leftrightarrow \varphi \leq a \text{ и } \psi \leq a \\ &\Leftrightarrow (\varphi \leq a) = 1 \text{ и } (\psi \leq a) = 1 \\ &\Leftrightarrow ((\varphi \leq a) \wedge (\psi \leq a)) = 1 \\ &\Leftrightarrow ((\varphi \leq a) \wedge (\psi \leq a)) = 1 \end{aligned}$$

Отсюда имеем $((\varphi \wedge \psi) \leq a) = 1 \Leftrightarrow (\varphi \geq a) \wedge (\psi \geq a) = 1$ при произвольной интерпретации “ $_$ ”. Следовательно, эквивалентность (5) справедлива.

Доказательство эквивалентности (6):

$$\begin{aligned} ((\varphi \rightarrow \psi) \geq a) = 1 &\Leftrightarrow ((\varphi \rightarrow \psi) \geq a) \\ &\Leftrightarrow (\varphi \rightarrow \psi) \geq a \\ &\Leftrightarrow \max\{1 - \varphi, \psi\} \leq a \\ &\Leftrightarrow 1 - \varphi \leq a \text{ и } \psi \leq a \\ &\Leftrightarrow \varphi \leq a = 1 \text{ и } \psi \leq a = 1 \\ &\Leftrightarrow ((\varphi \leq a) \wedge (\psi \leq a)) = 1 \\ &\Leftrightarrow ((\varphi \leq a) \wedge (\psi \leq a)) = 1 \end{aligned}$$

Отсюда $((\varphi \vee \psi) \leq a) = 1 \Leftrightarrow ((\varphi \leq a) \wedge (\psi \leq a)) = 1$ при произвольной интерпретации “ $_$ ”. Следовательно, эквивалентность (6) справедлива.

Доказательство эквивалентности (7):

$$\begin{aligned}
 ((\varphi \rightarrow \psi) \leq a) = 1 &\Leftrightarrow ((\varphi \rightarrow \psi) \leq a) \\
 &\Leftrightarrow ((\varphi \rightarrow \psi) \leq a) \\
 &\Leftrightarrow (\min\{\varphi, 1 - \psi\} \leq a) \\
 &\Leftrightarrow 1 - \varphi \leq a \text{ или } \psi \leq a \\
 &\Leftrightarrow \varphi \leq a = 1 \text{ или } \psi \leq a = 1 \\
 &\Leftrightarrow (\varphi \leq a \wedge \psi \leq a) = 1 \\
 &\Leftrightarrow ((\varphi \leq a) \wedge (\psi \leq a)) = 1
 \end{aligned}$$

Отсюда $((\varphi \vee \psi) \leq a) = 1 \Leftrightarrow ((\varphi \leq a) \wedge (\psi \leq a)) = 1$ при произвольной интерпретации “_”. Следовательно, эквивалентность (7) справедлива.

3. ПРАВИЛА ВЫВОДА ДЛЯ ЛОГИКИ ОЦЕНОК. СОСТОЯТЕЛЬНОСТЬ СИСТЕМЫ ПРАВИЛ Σ

Правила системы Σ записаны в таблицах табл. 1, табл. 2 и табл. 3. В табл. 1 даны правила, в результате применения которых получается структура дерева вывода. В табл. 2 содержатся бинарные правила вывода, которые используются для модификации ветвей дерева. В табл. 3 представлены правила для получения противоречий.

Следующая терминология используется в теории аналитических таблиц. *Ветвь* – это путь в дереве, начинающийся в корневой вершине дерева вывода (таблицы). *Полная ветвь* заканчивается в конечной вершине дерева. *Замкнутая ветвь* заканчивается в вершине со знаком X. *Открытая ветвь* не содержит вершин со знаком X. *Атомарная ветвь* получается из ветви выписыванием из нее атомарных оценок. *Выполнимая таблица* – дерево, имеющее открытую ветвь. *Завершенная таблица* – дерево, к которому уже не применимо никакое правило вывода. *Атомарные оценки* имеют вид $(p \geq a)$, $(p > a)$, $(p \leq a)$, $(p < a)$. *Замкнутая таблица* – это дерево вывода, у которого все ветви замкнуты. *Атомарная ветвь* получается из ветви выписыванием из нее атомарных оценок.

Правила из табл.1 делятся на *альтернативные* и *безальтернативные*. Первые – это правила со связкой «или», правые – со связкой «и». Таким образом, 1, 2, 5,7, 8–17, 22, 23 – номера всех безальтернативных и 3, 4, 6, 18–21 – номера всех альтернативных правил из табл. 1.

Рассмотрим в качестве примера построение дерева вывода для множества *Kb* («базы знаний»), состоящего из следующих оценок:

$$(\sim((p_3 \vee \sim p_1) \leq 0.6) \rightarrow ((p_2 \wedge \sim p_1) \geq 0.3))) \text{ и } ((p_1 \leq 0.2) \rightarrow (p_2 \geq 1)).$$

Дерево Т – результат этого построения – изображено на рис. 1.

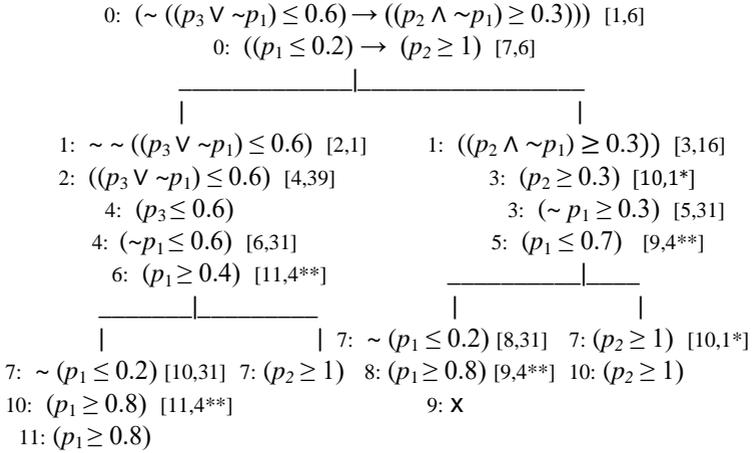


Рис. 1. Пример дерева вывода

Вершины Т имеют правые и левые метки. Например, первая вершина дерева имеет левую метку « 0: » и правую метку [1,4], Метка « 0: » сообщает, что первая вершина с оценкой

$$(\sim((p_3 \vee \sim p_1) \leq 0.6) \rightarrow ((p_2 \wedge \sim p_1) \geq 0.3))$$

принадлежит исходной базе знаний *Kb*, а метка [1,4] информирует, что к этой оценке на шаге 1 было применено правило номер 4 из табл. 1. Результатом применения этого правила стала пара оценок

$$\sim \sim((p_3 \vee \sim p_1) \leq 0.6) \text{ и } ((p_2 \wedge \sim p_1) \geq 0.3)),$$

которая в виде «вилки» была присоединена к начальной ветви (состоящей из первой и второй оценок дерева Т). При этом оценкам – компонентам этой пары – был приписана левая метка «1;», которая указывает, что оценки были получены на шаге 1. На шаге 3 к оценке $((p_2 \wedge \sim p_1) \geq 0.3))$ было применено правило 16, в результате чего к текущей ветви дерева были приписаны подряд оценки $(p_2 \geq 0.3)$ и $(\sim p_1 \geq 0.3)$. На шаге 9 к оценкам $(p_1 \leq 0.7)$ и $(p_1 \geq 0.8)$ было применено бинарное правило 1** (из табл. 2), которое дало знак противоречия χ .

В общем случае, если к оценке λ применяется альтернативное правило вывода, то ко всем проходящим через λ ветвям присоединяется вилка из компонент оценки λ ; если применяется безальтернативное правило, то присоединяются подряд все компонентные оценки λ .

Напомним, что *состоятельные* правила вывода в любой пропозициональной логике, где каждая формула истинна или ложна, это такие правила, которые сохраняют истинность формул. Легко видеть, что благодаря Лемме 1 состоятельны все правила вывода из табл. 1 и табл. 2.

Таблица 1. Правила для построения дерева вывода

Номер правила	Антецедент	Консеквент
1	$\sim \sim \lambda$	Λ
2	$(\lambda \wedge \mu)$	λ и μ
3	$\sim (\lambda \wedge \mu)$	$\sim \lambda$ или $\sim \mu$
4	$(\lambda \vee \mu)$	λ или μ
5	$\sim (\lambda \vee \mu)$	$\sim \lambda$ и $\sim \mu$
6	$(\lambda \rightarrow \mu)$	$\sim \lambda$ или μ
7	$\sim (\lambda \rightarrow \mu)$	λ и $\sim \mu$
8	$\sim (\varphi \geq a)$	$(\varphi < a)$
9	$(\varphi > a)$	$(\varphi \leq a)$
10	$(\varphi \leq a)$	$(\varphi > a)$
11	$\sim (\varphi < a)$	$(\varphi \geq a)$
12	$(\sim \varphi \geq a)$	$(\varphi \leq 1 - a)$
13	$(\sim \varphi > a)$	$(\varphi < 1 - a)$
14	$(\sim \varphi \leq a)$	$(\varphi \geq 1 - a)$
15	$(\sim \varphi < a)$	$(\varphi > 1 - a)$
16	$((\varphi \wedge \psi) \geq a)$	$(\varphi \geq a)$ и $(\psi \geq a)$
17	$((\varphi \wedge \psi) > a)$	$(\varphi > a)$ и $(\psi > a)$
18	$((\varphi \wedge \psi) \leq a)$	$(\varphi \leq a)$ или $(\psi \leq a)$
19	$((\varphi \wedge \psi) < a)$	$(\varphi < a)$ или $(\psi < a)$
20	$((\varphi \vee \psi) \geq a)$	$(\varphi \geq a)$ или $(\psi \geq a)$
21	$((\varphi \vee \psi) > a)$	$(\varphi > a)$ или $(\psi > a)$
22	$((\varphi \vee \psi) \leq a)$	$(\varphi \leq a)$ и $(\psi \leq a)$
23	$((\varphi \vee \psi) < a)$	$(\varphi > a)$ и $(\psi > a)$
24	$((\varphi \rightarrow \psi) \geq a)$	$(\varphi \leq 1 - a)$ или $(\psi \geq a)$
25	$((\varphi \rightarrow \psi) > a)$	$(\varphi < 1 - a)$ или $(\psi > a)$
26	$((\varphi \rightarrow \psi) \leq a)$	$(\varphi \geq 1 - a)$ и $(\psi \leq a)$
27	$((\varphi \rightarrow \psi) < a)$	$(\varphi > 1 - a)$ и $(\psi < a)$

Таблица 2. Бинарные правила вывода

Номер правила	Антецеденты	Консеквент
1*	$(\varphi \geq a), (\varphi \geq b),$ $(\varphi \geq a) \text{ --- } (\varphi \geq b)$	$(\varphi \geq \max\{a, b\})$
2*	$(\varphi \geq a), (\varphi > b),$ $(\varphi \geq a) \text{ --- } (\varphi > b)$	$(\varphi > \max\{a, b\})$
3*	$(\varphi > a), (\varphi \geq b),$ $(\varphi > a) \text{ --- } (\varphi \geq b)$	$(\varphi > \max\{a, b\})$
4*	$(\varphi > a), (\varphi > b),$ $(\varphi > a) \text{ --- } (\varphi > b)$	$(\varphi > \max\{a, b\})$
5*	$(\varphi \leq a), (\varphi \leq b),$ $(\varphi \leq a) \text{ --- } (\varphi \leq b)$	$(\varphi \leq \min\{a, b\})$
6*	$(\varphi \leq a), (\varphi < b),$ $(\varphi \leq a) \text{ --- } (\varphi < b)$	$(\varphi < \min\{a, b\})$
7*	$(\varphi < a), (\varphi \leq b),$ $(\varphi < a) \text{ --- } (\varphi \leq b)$	$(\varphi < \min\{a, b\})$
8*	$(\varphi < a), (\varphi < b),$ $(\varphi < a) \text{ --- } (\varphi < b)$	$(\varphi < \min\{a, b\})$

$\alpha \text{ --- } \beta$ означает, что оценки α и β лежат на одной ветви.

Таблица 3. Правила для получения противоречий

Номер правила	Антецеденты	Консеквент
1**	$\varphi > a, \varphi < a$ $\varphi > a \text{ --- } \varphi < a$	х
2**	$\varphi > a, \varphi \leq a$ $\varphi > a \text{ --- } \varphi < a$	х
3**	$\varphi \geq a, \varphi < a$ $\varphi > a \text{ --- } \varphi < a$	х
4**	$(\varphi \geq a), (\varphi \leq b), a > b,$ $(\varphi \geq a) \text{ --- } (\varphi \leq b)$	х
5**	$(\varphi > a), (\varphi \leq b), (a \geq b),$ $(\varphi \geq a) \text{ --- } (\varphi \leq b)$	х
6**	$(\varphi > a), (\varphi \leq b), (a \geq b),$ $(\varphi \leq b) \text{ --- } (\varphi \geq a)$	х
7**	$(\varphi > a), (\varphi < b), (a \geq b)$ $(\varphi \geq a) \text{ --- } (\varphi \leq b)$	х
8**	$(\varphi > a), (\varphi < b), (a \geq b)$ $(\varphi \leq b) \text{ --- } (\varphi \geq a)$	х

Очевидно, что дерево T (рис. 2) имеем следующие полные атомарные ветви:

$$B1 = \{(p_3 \leq 0.6), (p_1 \geq 0.4), (p_1 \geq 0.8), (p_1 \geq 0.8)\};$$

$$B2 = \{(p_3 \leq 0.6), (p_1 \geq 0.4), (p_2 \geq 1)\};$$

$$B3 = \{(p_2 \geq 0.3), (p_1 \leq 0.7), (p_1 \geq 0.8), x\};$$

$$B4 = \{(p_2 \geq 0.3), (p_1 \leq 0.7), (p_2 \geq 1), (p_2 \geq 1)\}$$

Применяя к этим ветвям, за исключением замкнутой ветви $B3$, правила из табл. 2, получаем эквивалентные ветви

$$\underline{B1} = \{(p_3 \leq 0.6), (p_1 \geq 0.4), (p_1 \geq 0.8), (p_1 \geq 0.8)\}.$$

Изображенное на рис. 1 дерево вывода можно использовать для доказательства логических следствий, полученных из множества Kb («базы знаний»). Например, докажем, что из базы знаний Kb логически следует оценка $((p_1 \geq 0.8) \vee (p_1 \geq 1))$. Для этого в начало дерева T запишем отрицание этой оценки, т.е. оценкой $\sim((p_1 \geq 0.8) \vee (p_2 \geq 1))$, к которой теперь применим правило номер 5. В результате к каждой открытой ветви дерева T будут приписаны подряд оценки $\sim(p_1 \geq 0.8)$ и $\sim(p_2 \geq 1)$, из которых затем будут получены оценки $(p_1 < 0.8)$ и $(p_2 < 1)$. Наконец, к парам оценок $((p_1 < 0.8), (p_1 \geq 0.8))$ и $((p_2 < 1), (p_2 \geq 1))$ применяем правило номер 2* и получаем противоречия x по всем ветвям дерева, что свидетельствует о невыполнимости базы знаний Kb . Следовательно, имеем логическое следствие $Kb \models ((p_1 \geq 0.8) \vee (p_1 \geq 1))$. На самом деле (как легко показать) знак \models можно заменить на знак \models^* оптимального логического следствия.

4. СОСТОЯТЕЛЬНОСТЬ И ПОЛНОТА СИСТЕМЫ Σ ПРАВИЛ ВЫВОДА ДЛЯ ЛОГИКИ ОЦЕНОК LE

Напомним, что *состоятельные* правила вывода в любой пропозициональной логике (где каждая формула истинна или ложна) это такие правила, которые сохраняют истинность формул. Легко видеть, что благодаря Лемме 1 состоятельны все правила вывода из табл. 1 и табл. 2. Таким образом, состоятельна система Σ этих правил вывода.

Доказательство полноты системы Σ можно выполнить по схеме Хинтикки, используя понятие хинтиккова множества для системы правил Σ . Множество N оценок назовем *хинтикковым*, если оно обладает следующими свойствами:

- 1: N не содержит контрарных пар (т.е. пар оценок, представленных как antecedentes правил из табл. 3);
- 2: N замкнуто относительно бинарных правил из табл. 2;
- 3: $\sim \lambda \in N \Rightarrow \lambda \in N$;
- 4: $(\lambda \wedge \mu) \in N \Rightarrow \lambda \in N$ и $\mu \in N$;

- 5: $\sim (\lambda \wedge \mu) \in E \Rightarrow \sim \lambda \in H$ или $\sim \mu \in H$;
6: $(\lambda \vee \mu) \in H \Rightarrow \lambda \in H$ или $\mu \in H$;
7: $\sim (\lambda \vee \mu) \in H \Rightarrow \sim \lambda \in H$ и $\sim \mu \in H$;
8: $(\lambda \rightarrow \mu) \in H \Rightarrow \sim \lambda \in E$ или $\mu \in H$;
9: $\sim (\lambda \rightarrow \mu) \in H \Rightarrow \lambda \in H$ и $\sim \mu \in H$;
10: $\sim (\varphi \geq a) \in H \Rightarrow (\varphi < a) \in H$;
11: $\sim (\varphi > a) \in H \Rightarrow (\varphi \leq a) \in H$;
12: $\sim (\varphi \leq a) \in H \Rightarrow (\varphi > a) \in H$;
13: $\sim (\varphi < a) \in H \Rightarrow (\varphi \geq a) \in H$;
14: $(\sim \varphi \geq a) \in H \Rightarrow (\varphi \leq 1-a) \in H$;
15: $(\sim \varphi > a) \in H \Rightarrow (\varphi < 1-a) \in H$;
16: $(\sim \varphi \leq a) \in H \Rightarrow (\varphi \geq 1-a) \in H$;
17: $(\sim \varphi < a) \in H \Rightarrow (\varphi > 1-a) \in H$;
18: $((\varphi \wedge \psi) \geq a) \in H \Rightarrow (\varphi \geq a) \in H$ и $(\psi \geq a) \in H$;
19: $((\varphi \wedge \psi) > a) \in H \Rightarrow (\varphi > a) \in H$ и $(\psi > a) \in E$;
20: $((\varphi \wedge \psi) \leq a) \in H \Rightarrow (\varphi \leq a) \in H$ или $(\psi \leq a) \in H$;
21: $((\varphi \wedge \psi) < a) \in H \Rightarrow (\varphi < a) \in H$ или $(\psi < a) \in H$;
22: $((\varphi \vee \psi) \geq a) \in H \Rightarrow (\varphi \geq a) \in H$ или $(\psi \geq a) \in H$;
23: $((\varphi \vee \psi) > a) \in H \Rightarrow (\varphi > a) \in H$ или $(\psi > a) \in H$;
24: $((\varphi \vee \psi) \leq a) \in H \Rightarrow (\varphi \leq a) \in H$ и $(\psi \leq a) \in H$;
25: $((\varphi \vee \psi) < a) \in H \Rightarrow (\varphi \geq a) \in H$ и $(\psi \geq a) \in H$;
26: $((\varphi \rightarrow \psi) \geq a) \in H \Rightarrow (\varphi \geq 1-a) \in H$ или $(\psi \geq a) \in H$;
27: $((\varphi \rightarrow \psi) > a) \in H \Rightarrow (\varphi > 1-a) \in H$ или $(\psi > a) \in H$;
28: $((\varphi \rightarrow \psi) \leq a) \in H \Rightarrow (\varphi \leq 1-a) \in H$ и $(\psi \leq a) \in H$;
29: $((\varphi \vee \psi) < a) \in H \Rightarrow (\varphi < 1-a) \in H$ и $(\psi < a) \in H$.

Обозначим $At(H)$ множество всех входящих в H атомарных оценок. Используя свойства в пунктах п.1 и п.2, легко установить, что для каждого p имеет место следующее:

- А) существует ровно одно значение $a \in [0,1]$, что $(p \geq a) \in At(H)$;
В) существует ровно одно значение $a \in [0,1]$, что $(p < a) \in At(H)$;
С) существует единственная пара значений $c, d \in [0,1]$, что $(p \geq c), (p \leq d) \in At(H)$.

Лемма 2 (лемма Хинтикки для системы правил вывода, содержащихся в табл. 1). Всякое хинтикково множество выполнимо.

Эта лемма доказывается с использованием леммы 1 и применением индукции по структуре входящих в множество H оценок.

Следующим образом определим интерпретацию “ $_$ ”, при которой все оценки из H окажутся истинны:

“ p ” = a (в случае A); “ p ” = b (в случае B);

{“ p ” = c , “ p ” = a } (в случае C).

Полнота системы Σ правил вывода для логики оценок **LE** (**LE_h**) следует из следующего утверждения.

Утверждение 1. Пусть T – завершенная таблица, построенная для базы знаний Kb . Тогда Kb является невыполнимым множеством в том и только том случае, если таблица T замкнута.

1. Kb невыполнимо $\Rightarrow T$ замкнута.

Предположим, что таблица T не замкнута. Тогда она имеет открытую ветвь V . Ясно, что ветвь является хинтиковым множеством, и по лемме 1 тогда ветвь V истинна в некоторой интерпретации. Поэтому в этой интерпретации истинной будет и множество Kb (как часть V). Следовательно, множество Kb выполнимо.

2. T замкнута $\Rightarrow Kb$ невыполнимо. В силу состоятельности правил из Σ при построении дерева T не может возникнуть невыполнимая ветвь, в частности такая, которая содержит знак противоречия X .

5. ЗАКЛЮЧЕНИЕ

Мы определили пропозициональную логику **LE**, предложениями которой служат оценки истинностных значений для утверждений нечеткой пропозициональной логики Заде **L**. Логику оценок **LE** можно рассматривать как металогику для логики Заде. Мы построили для логики оценок **LE** состоятельную и полную систему логического вывода, основанную на идеях метода аналитических таблиц.

Литература

1. Agostino M., Gabbay D., Hahnle R., Possega J. Handbook of tableaux methods. – Springer, 2001.
2. Chen J., Kundu S. A sound and complete fuzzy logic system using Zadeh’s implication operator. In Ras Z.W., & Maciek M. (eds.) // Proc. of the 9th Int. Sym. on Methodologies for Intelligent Systems. No. 1079. Lecture Notes in Artificial Intelligence. – Springer, 1996. – P. 233-242.
3. Fitting M. First-order logic and automated theorem proving. – Springer, 1996.
4. Kundu S. An improved method for fuzzy-inferencing using Zadeh’s implication operator // Proceedings of IJCALWorkshop on Fuzzy logic in AI. 117-125. – Springer, 1995.

ТЕМПОРАЛЬНЫЕ НЕЙРО-НЕЧЕТКИЕ СЕТИ ПЕТРИ ДЛЯ МОДЕЛИРОВАНИЯ ИНФОРМАЦИОННО-ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ

Прокопенко С.А.

Белорусско-Российский университет

e-mail: puss95@yandex.by

Бобряков А.В., д.т.н., профессор

Национальный исследовательский университет «МЭИ»

e-mail: avbob@mail.ru

1. ВВЕДЕНИЕ

Процессы в современных производственных информационно-технологических системах все в большей степени приобретают комбинированный характер информационно-аналитических и производственно-технологических процессов, представляющих собой конвергенцию решаемых производственно-технологических и информационно-аналитических задач.

При этом производственно-технологические задачи определяют следующие особенности информационно-технологических процессов:

- могут быть реализованы различными совокупностями технологических операций;
- динамичное изменение структуры и параметров процесса;
- функциональные подсистемы и их элементы взаимозависимы, кроме того, элементы одной подсистемы могут являться одновременно и элементами другой подсистемы [1];
- процессы отличаются сложностью, которая возрастает по мере повышения требований к эффективности функционирования;
- являются энерго- и ресурсоемкими;
- могут относиться к категории вредных и опасных (с точки зрения техногенных нарушений и аварийных ситуаций);
- ограниченные возможности проведения экспериментальных исследований;
- отсутствие точной информации о состоянии оборудования;
- большинство процессов характеризуются непрерывным циклом.

В свою очередь, информационно-аналитические задачи характеризуются:

- неполнотой информации о функционировании и состоянии процессов;
- наличием сложных нелинейных зависимостей между параметрами;
- необходимостью обработки и анализа больших объемов разнородной информации;
- высокой динамикой изменения системных и внешних факторов, влияющих на эффективность управления [2, 3].

Сети Петри хорошо зарекомендовали себя как удобный, наглядный аппарат для моделирования и анализа сложных процессов. Однако их использование для моделирования информационно-технологических процессов без использования расширений затруднительно. Это связано с особенностями функционирования таких процессов.

Зачастую на практике применяют разновидности сетей Петри, которые позволяют учитывать особенности конкретных процессов. Так, нечеткие сети Петри позволяют учитывать нестохастическую неопределенность, нейро-нечеткие сети Петри – учитывать накопленный опыт ([4]), а временные сети Петри – моделируемых временные аспекты этих процессов.

2. ТЕМПОРАЛЬНЫЕ НЕЙРО-НЕЧЕТКИЕ СЕТИ ПЕТРИ

Предлагаемая темпоральная нейро-нечеткая сеть Петри (ТННСП) может быть представлена следующим образом:

$$TNFPN = (P, T, I, O, F, Al, D, m_0, \alpha, R),$$

где $P = \{P_1, P_2, \dots, P_{np}\}$ – конечное непустое множество вершин сети Петри, np – общее количество вершин; $T = \{T_1, T_2, \dots, T_{nt}\}$ – конечное непустое множество переходов сети Петри, nt – общее количество переходов; $I = \{I_1, I_2, \dots, I_{ni}\}$ – входные функции переходов; $O = \{O_1, O_2, \dots, O_{no}\}$ – выходные функции переходов; $F = \{F_1, F_2, \dots, F_{nf}\}$ – конечное непустое множество максимального времени выполнения переходов сети Петри; $Al = \{Al_1, Al_2, \dots, Al_{na}\}$ – множество темпоральных правил логики Алена для переходов сети, na – общее число правил; $D = \{D_1, D_2, \dots, D_{nd}\}$ – конечное множество значений меток, необходимое для срабатывания перехода при переходе по дуге, nd – общее количество дуг; m_0 – вектор начальной маркировки, каждый компонент которого определяется значением функции принадлежности нечеткого наличия маркера в соответствующей позиции; $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_{nr}\}$ – вектор значений срабатывания перехода, $R = \{R_1, R_2, \dots, R_{nr}\}$ – конечное множество ресурсов.

На рис. 1 представлена обобщенная структура ТННСП.

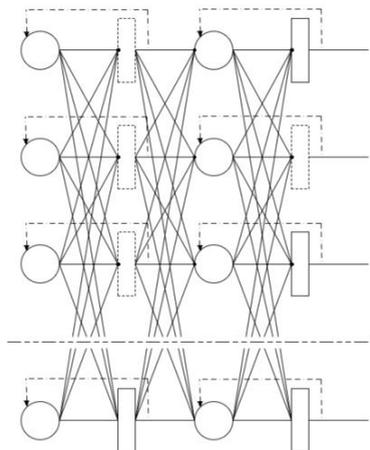


Рис. 1. Обобщенная структура темпоральной нейро-нечеткой сети Петри

Применение нейро-нечетких переходов в сетях Петри, реализуемых с использованием нечетких нейронов [5] позволяет учитывать неопределенность и ранее полученные результаты структурно-параметрической настройки.

Моделирование информационно-технологических процессов может характеризоваться различными траекториями, построение которых затруднено для классических или нейро-нечетких сетей Петри, а выбор актуальной траектории является полнопереборной задачей [6].

Использование же для предлагаемой модели темпоральной логики Аллена позволяет описывать последовательности моделируемых событий их взаимосвязи по временной шкале и учитывать временные причинно-следственные связи [7, 8]. Особенностью темпоральной логики Аллена является использование временных интервалов. Это также позволяет резервировать и адаптивно распределять ресурсы на временных интервалах в рамках моделируемого информационно-технологического процесса.

На рис. 2 показан пример диаграммы использования ресурса, с возможностью его гибкого перераспределения с использованием ТННСП без изменения других параметров информационно-технологического процесса.

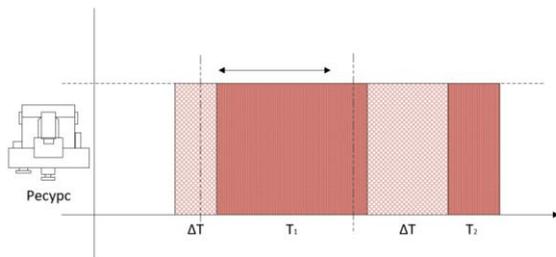


Рис. 2. Пример перераспределения ресурса с использованием ТНСП

В итоге удастся существенно снизить количество возможных траекторий реализации информационно-технологического процесса [9]. На рис. 3 показан пример структуры ТНСП, моделирующая выбранную траекторию информационно-технологического процесса с учетом ограничений, накладываемых временными причинно-следственными связями.

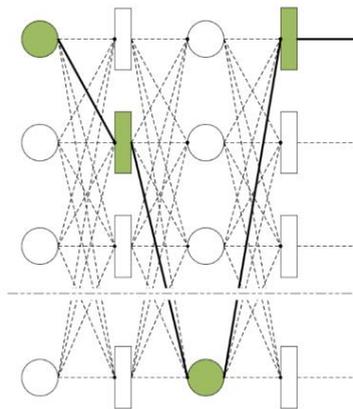


Рис. 3. Пример структуры ТНСП, моделирующая траекторию информационно-технологического процесса

На рис. 4 представлен фрагмент ТНСП, иллюстрирующий совместное использование нейро-нечетких элементов и временных причинно-следственных связей (реализуемых с использованием правил темпоральной логики Аллена) между последовательностями событий при моделировании информационно-технологического процесса в условиях неопределенности.

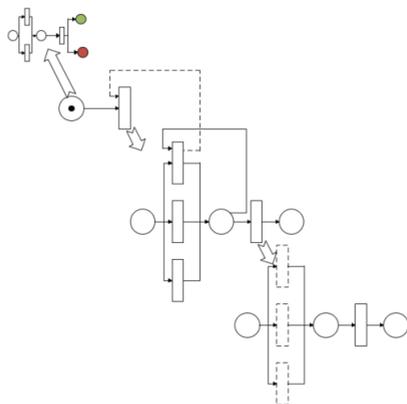


Рис. 4. Фрагмент ТНСП, иллюстрирующий совместное использование нейро-нечетких элементов и правил темпоральной логики Аллена

3. ПРОГРАММНОЕ ПРИЛОЖЕНИЕ

Разработано программное приложение для моделирования информационно-технологических процессов на основе ТНСП, обладающая следующими возможностями (рис. 5):

- формирование онтологической модели сложной системы, включающей классы информационных сущностей и квазиерархические связи между ними;

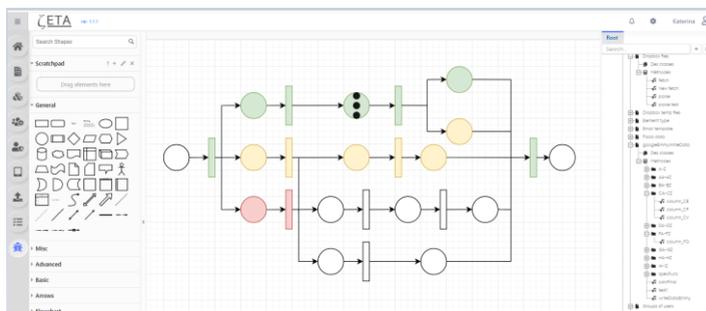


Рис. 5. Экранная форма разработанного приложения

- ввод и хранение пользовательских и получаемых из внешних источников данных;
- использование нейросетевого супервизора для валидации внешних и аналитических данных;

- построение информационно-технологического процесса с использованием конструкторов на основе ТННСП;
- гибкая настройка управления приоритетами конкурирующих за ресурсы информационно-технологических процессов на основе ТННСП.

4. ЗАКЛЮЧЕНИЕ

Предложены темпоральные нейро-нечеткие сети Петри, ориентированные на моделирование информационно-технологических процессов, представляющих собой конвергенцию решаемых производственно-технологических и информационно-аналитических задач.

Использование предлагаемых моделей позволяет моделировать информационно-технологические процессы с учетом временных причинно-следственных взаимосвязей между последовательности событий в условиях неопределенности, и обеспечивает существенное снижение числа возможных траекторий реализации этих процессов.

Литература

1. Bobryakov A.V., Borisov V.V., Misnik A.E., Prokopenko S.A. Approaches to the Implementation of Information-Analytical Processes in Complex Technical-Organizational Systems // V International Conference on Information Technologies in Engineering Education (Inforino). – 2020. – DOI: 10.1109/Inforino48376.2020.9111833.
2. Kutuzov V.V., Misnik A.E., Prakapenka S.A. Development of Information-Analytical Processes in Cyber-Physical Systems Based on Neural-Fuzzy Petri Nets // International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon). 2020. 10.1109/FarEastCon50210.2020.9271400.
3. Borisov V.V., Misnik A.E., Krutalevich S.K., Prokopenko S.A. Automation Methodology for Complex Technical-Organizational Systems // V International Conference on Information Technologies in Engineering Education (Inforino). – 2020. – DOI: 10.1109/Inforino48376.2020.9111813.
4. Борисов В.В., Захарченков К.В., Кутузов В.В., Мисник А.Е., Прокопенко С.А. Моделирование образовательных процессов на основе нейро-нечетких темпоральных сетей Петри // Прикладная информатика. – 2021. – № 4. – С. 35-47.
5. Kwan H., Cai L. A fuzzy neural network and its application to pattern recognition // IEEE Transactions on Fuzzy System. – 1994. – Vol. 2, No. 3. – P. 185-193.
6. Andrade E., Maciel P., Callou G., Araujo C. An Approach Based in Petri Net for Requirement Analysis. – 2010. – DOI: 10.5772/7495.
7. Griffin J., Lcsani M., Shadab N., Yin X. TLC: temporal logic of distributed components // Proceedings of the ACM on Programming Languages. – 2020. – Vol. 4, (ICFP), article 23. – DOI: 10.1145/3409005.
8. Balbiani P., Boudou J., Diczguz M., Fernandez-Duque D. Intuitionistic linear temporal logics // ACM Transactions on Computational Logic. – 2019. – Vol. 21, No. 2. – P. 1-32. – DOI: 10.1145/3365833.
9. Wu J., Yan S. Reliability Evaluation for Mechanical Systems by Petri Nets // Petri Nets in Science and Engineering. – 2018. – DOI: 10.5772/intechopen.79624.

ИССЛЕДОВАНИЕ ЗАДАЧИ УПРАВЛЕНИЯ ПРОЦЕССОМ ПРОИЗВОДСТВА АЦЕТИЛЕНА НА ОСНОВЕ НЕЧЁТКОЙ ЛОГИКИ*

*Санаева Г.Н., к.т.н., доцент
НИ РХТУ им. Д.И. Менделеева*

e-mail: gsanaeva@nirhtu.ru

*Богатиков В.Н., д.т.н., профессор
ТвГТУ*

e-mail: vnbgtk@mail.ru

1. ВВЕДЕНИЕ

При производстве ацетилена окислительным пиролизом природного газа обязательным условием является строгое соблюдение требований технологической безопасности его протекания, что является особенно актуальным из-за высокой взрыво- и пожароопасности и требует создания системы управления процессом с учётом его специфических особенностей.

2. ПОСТАНОВКА ЗАДАЧИ

Процесс окислительного пиролиза природного газа требует неукоснительного соблюдения ряда технологических ограничений:

- максимально допустимая температура газораспределительной решетки реактора, $T_{\text{реш}}^{\text{max}}$ (710°C);
- ограничения на содержание метана и кислорода в газе пиролиза на выходе из реактора, $C_{\text{CH}_4}^{\text{max}}$ и $C_{\text{O}_2}^{\text{max}}$ (9% и 0,8 (об.));
- минимальный допустимый расход воды на быстрое охлаждение («закалку») продуктов окислительного пиролиза, $G_{\text{B}}^{\text{min}}$ ($10 \text{ м}^3/\text{ч}$);
- ограничения на диапазон изменения температуры газа пиролиза на выходе из реактора, $G_{\text{ГП}}^{\text{min}}$ и $G_{\text{ГП}}^{\text{max}}$ ($50\text{--}100^{\circ}\text{C}$).

Таким образом, для процесса производства ацетилена окислительным пиролизом необходимо обеспечение его протекания в условиях технологической безопасности с учетом поддержания требуемого содержания ацетилена в газе пиролиза.

* Работа выполнена при финансовой поддержке РФФИ, проект №20-07-00914.

3. ОПРЕДЕЛЕНИЕ ЦЕНТРА БЕЗОПАСНОСТИ

Наивысший уровень технологической безопасности процесса окислительного пиролиза соответствует точке максимальной безопасности (*opt*), т.е. его центру безопасности (ЦБ). На основе математической модели процесса центр безопасности можно определить как точку, максимально удаленную от границ области безопасности с учетом технологических ограничений, накладываемых на процесс (рис. 1) [1].

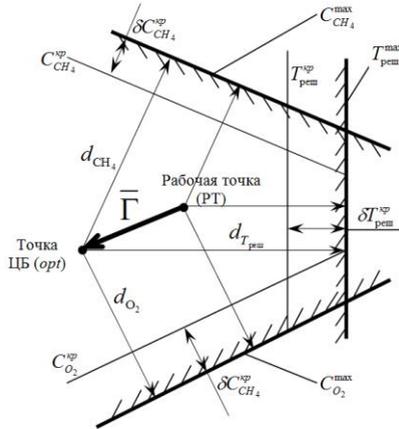


Рис. 1. Положение рабочей точки процесса

При этом определение центра безопасности как точки обеспечения максимально безопасного состояния процесса сводится к решению задачи нелинейного программирования:

$$\sum_{i=1}^N d_i^2 \rightarrow \max$$

где d_i – расстояние от рабочей точки процесса до границ области его безопасного протекания [2–4].

На практике поддержание процесса окислительного пиролиза в точке центра безопасности может сопровождаться уменьшением содержания ацетилена в газе пиролиза. Поэтому допустимо нахождение рабочей точки процесса в иной точке внутри области безопасности, определяемой технологическими ограничениями процесса. Уровень технологической безопасности при этом может несколько снизиться, но при этом будет обеспечено требуемое содержание ацетилена в газе пиролиза.

4. КРИТЕРИЙ УПРАВЛЕНИЯ

Для следующих переменных, определяемых технологическими ограничениями, накладываемыми на процесс: температура газораспределительной решетки $T_{грш}$, содержание метана C_{CH_4} и кислорода C_{O_2} в газе пиролиза, расход воды на «закалку» продуктов окислительного пиролиза G_B , температура газа пиролиза $T_{гп}$ – установлены диапазоны, определяющие состояние процесса с точки зрения обеспечения его технологической безопасности. Основным для этих переменных этом является диапазон *LOW* (рис. 2).

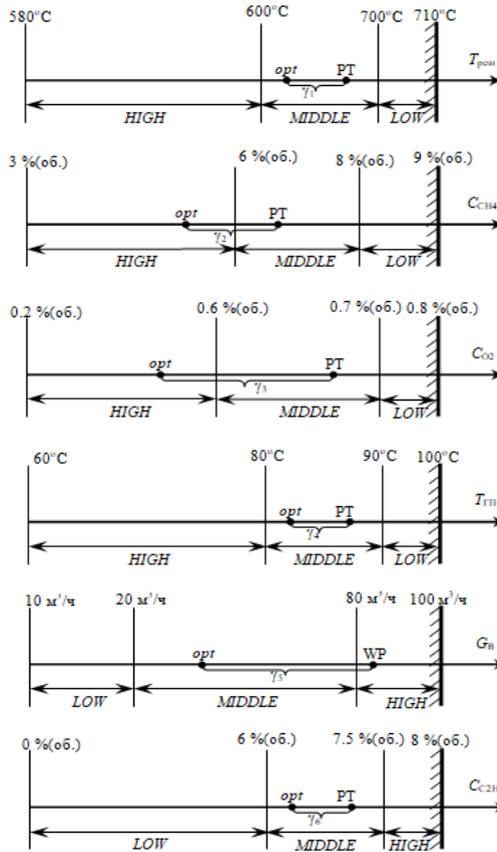


Рис. 2. Диапазоны протекания процесса окислительного пиролиза

В области *LOW* процесс ещё протекает безопасно, но из-за возникновения различного рода возмущений может перейти в аварийную ситуацию. Поэтому, если рабочая точка процесса находится в диапазоне *LOW* для переменных, определяющих его технологическую безопасность независимо от содержания ацетилена в газе пиролиза, необходимо осуществить как можно более быстрый переход в безопасную область (диапазоны *MIDDLE* и *HIGH*).

Если содержание ацетилена достаточное (диапазон *HIGH*), процесс протекает в условиях технологической безопасности (переменные $T_{\text{реш}} - T_{\text{ГП}}$ находятся в диапазонах *MIDDLE* и *HIGH*), то достаточно поддержание имеющихся значений управляемых переменных локальных контуров управления.

При управлении процессом окислительного пиролиза необходимо минимизировать сумму квадратов расстояний $\Gamma(\gamma_i)$ от текущей точки процесса до точки, соответствующей центру безопасности (рис. 1 и 2):

$$\Gamma(\gamma_i) = \sum_i k_i \cdot \gamma_i^2 \rightarrow \min$$

при соблюдении следующих ограничений:

$$\bar{\delta} > \bar{\delta}^{\text{кр}},$$

где δ_i и $\delta_i^{\text{кр}}$ – отклонение реальных измеренных значений от границы области безопасности и критическое значение этого отклонения по данной переменной i , k_i – нормирующий коэффициент.

5. СИНТЕЗ СИСТЕМЫ УПРАВЛЕНИЯ

Поскольку непосредственное регулирование концентрации ацетилена как результирующего продукта реакции окислительного пиролиза затруднительно, целесообразным является обеспечение условий протекания рассматриваемого процесса с целью поддержания максимально возможного содержания ацетилена в газе пиролиза путем стабилизации основных технологических переменных в области технологической безопасности [5].

В данной работе предлагается двухуровневая система управления (рис. 3). Нижний уровень представлен следующими локальными контурами управления:

- стабилизация расходов метана и кислорода на реакцию окислительного пиролиза G_{CH_4} и G_{O_2} (регуляторы P1 и P2);

- регулирование температуры $T_{ГП}$ изменением расхода воды на «закалку» продуктов реакции $G_{В}$ (регулятор P3);
- регулирование температуры $T_{реш}$ изменением расхода природного газа на горелку подогревателя $G_{гор}$ (регулятор P4).

В соответствии с регламентом ведения процесса окислительного пиролиза контроль перечисленных температур производится автоматически постоянно, контроль состава газа пиролиза по содержанию метана (C_{CH_4}), кислорода (C_{O_2}) и ацетилена ($C_{C_2H_2}$) производится периодически (1 раз в час).

На верхнем уровне системы управления Блок коррекции производит расчёт заданий для регуляторов P1-P4 с учётом положения рабочей точки процесса с использованием правил нечёткого вывода [6, 7] следующего вида:

IF (δC_{O_2} is «HIGH») AND ($C_{C_2H_2}$ is «HIGH») AND (δC_{CH_4} is «LOW») THEN (G_{CH_4} is «LOW»);

IF (δC_{O_2} is «LOW») AND (δC_{CH_4} is «MIDDLE») AND ($C_{C_2H_2}$ is «LOW») THEN (G_{CH_4} is «MIDDLE»);

IF (δC_{O_2} is «MIDDLE») AND (δC_{CH_4} is «LOW») AND ($C_{C_2H_2}$ is «LOW») THEN (G_{CH_4} is «LOW»)

...

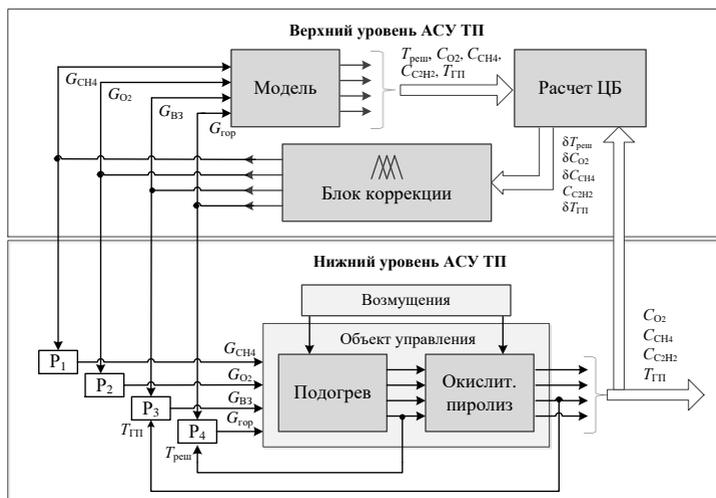


Рис. 3. Система управления процессом окислительного пиролиза

С учётом того, что значения C_{CH_4} , C_{O_2} и $C_{C_2H_2}$ определяются периодически, работа верхнего уровня заключается в следующем.

1. В начале часа текущего часа поступают сведения с объекта управления, и производится расчёт положения рабочей точки процесса: если она оказывается в области недопустимого приближения к предельным значениям переменных, определяющих технологическую безопасность процесса (диапазон *LOW*), необходимо как можно быстрее перевести процесс в более безопасную область (диапазоны *MIDDLE* и *HIGH*). Иначе для переменных $T_{реш} - T_{ГП}$ рассчитываются отклонения от их критических значений.

2. Для согласования значений переменных, полученных с объекта управления, и переменных, определяемых по математической модели процесса, рассчитывается вектор расхождений α , как отношение соответствующих параметров.

3. Для расчёта по математической модели процесса в качестве начальных значений переменных принимаются значения, полученные с объекта управления.

4. Исходя из правил нечёткого вывода, Блок коррекции верхнего уровня определяет значения заданий регуляторов P1-P4 нижнего уровня.

При отсутствии данных о составе газа пиролиза для расчета заданий регуляторов P1-P4 используется математическая модель процесса. При получении новых данных с объекта управления производится корректировка заданий регуляторов P1-P4 нижнего уровня управления.

Моделирование предлагаемого варианта системы управления позволяет сделать вывод о том, что значения переменных, определяющих технологическую безопасность процесса окислительного пиролиза, не выходят за пределы, определяемые технологическими ограничениями, накладываемыми на процесс.

6. ЗАКЛЮЧЕНИЕ

Проведённое моделирование позволяет сделать вывод о том, что использование аппарата нечёткой логики уменьшает вероятность перехода процесса окислительного пиролиза в штатную или аварийную ситуацию, что сопровождается снижением числа остановок процесса из-за возможного перехода в аварийную ситуацию. Это, в свою очередь, позволяет уменьшить непродуктивное использование всех видов ресурсов процесса и уменьшает вероятность загрязнения окружающей среды.

Литература

1. Антонов В.Н., Лapidус А.С. Производство ацетилена. – М.: Химия, 1970. – 416 с.
2. Мелихов А.Н., Бернштейн Л.С., Коровин С.Я. Ситуационные советующие системы с нечеткой логикой. – М.: Наука, 1990. – 272 с.
3. Палюх Б.В., Богатиков В.Н., Пророков А.Е., Алексеев В.В. Приложение метода разделения состояний для управления технологической безопасностью промышленных процессов на основе нечетко определенных моделей. – Тверь: ТГТУ, 2009. – 348 с.
4. Поспелов Д.А. Ситуационное управление: Теория и практика. – М.: Наука, 1986. – 288 с.
5. Виноградов Г.П., Богатиков В.Н., Кузнецов В.Н. Возможностная оптимизация в задачах принятия решений в мотивированных системах // Вестник Астраханского государственного технического университета: Управление, вычислительная техника и информатика. – 2019. – № 4. – С. 131-140.
6. Яхьяева Г.Э. Нечеткие множества и нейронные сети. – М.: Интернет-университет информационных технологий, 2014. – 320 с.
7. Вент Д.П., Ерофеев В.И., Богатиков В.Н., Лопатин А.Г., Пророков А.Е., Иляхинский А.В., Санаева Г.Н., Брыков Б.А., Мурашев П.М. Управление технологической безопасностью промышленных процессов на основе нечетко-определенных моделей. – Новомосковск: РХТУ им. Д.И. Менделеева, Новомосковский институт (филиал), 2019. – 209 с.

ПРИМЕНЕНИЕ МОДЕЛИ ПЯТЕН К РЕШЕНИЮ ОБРАТНЫХ ЗАДАЧ И ИСКУССТВЕННОМУ ИНТЕЛЛЕКТУ*

*Симонов Н.А., к.ф.-м.н., с.н.с.
ФТИАН им. К.А. Валиева РАН
e-mail: nsimonov@ftian.ru*

1. ВВЕДЕНИЕ

Необходимость развития новых подходов для моделирования и решения задач искусственного интеллекта (ИИ) связана с существенными ограничениями для применения традиционных методов при решения практических задач. По мнению автора, одним из фундаментальных недостатков существующей модели нейрона и нейронных сетей является моделирование процесса обучения и распознавания образов с помощью математических операций с действительными числами и применение численных методов. Это, в принципе, неадекватно природе человеческого мышления, которое основано на образном представлении и логической обработке информации с использованием ментальных образов в качестве базисных элементов.

Современный числовой подход к построению глубинных нейронных сетей (ГНС) с применением оптимизационных и стохастических подходов приводит к неустраняемым и неожиданным (глупым) ошибкам распознавания (классификации) образов [1]. Этим обусловлена и особенность этих ошибок, когда, даже при наличии четких изображений для нейронных сетей существует их вероятность, что представляет особую опасность. По-видимому, этим можно объяснить и тот факт, что до сих пор не удалось поднять точность классификации заметно выше 90% для базы изображений ImageNet в процессе соревнования лучших систем ГНС [2].

Автор статьи [1] говорит о принципиальных недостатках ГНС, которые обусловлены тем, что они в реальности не понимают окружающего мира. Поэтому появляется возможность неожиданных ошибок и большой потенциал обмана системы распознавания. Например, простого вращения изображения достаточно для того, чтобы обмануть

* Работа выполнена в рамках Государственного задания ФТИАН им. К.А. Валиева РАН Минобрнауки РФ по теме № FFNN-2022-0019.

обученную ГНС. Наклейка четырех небольших прямоугольных стикеров на дорожный знак «СТОП» приводит сеть к ошибочному распознаванию знака как «ограничение скорости 45 км/час». Изображение объектов природы на определенном природном или текстурном фоне также приводит к глупым ошибкам распознавания. Следует еще раз подчеркнуть, что эти ошибки отражают принципиальные недостатки ГНС [1], а не связаны с отдельными ошибочными алгоритмами. Специалисты говорят, что область ГНС во многом эмпирическая, и нет теории, объясняющей, почему система перестает работать. Никто не знает, как это улучшить, – говорит Джефф Клун из Вайомингского университета в Ларами [1].

Эти недостатки не позволяют использовать нейронные сети в критических областях, связанных с безопасностью и здоровьем людей. В статье [3] говорится, что Мишель Бачелет, Верховный комиссар ООН по правам человека сказала, что правительства должны прекратить сканирование черт лица людей в режиме реального времени, пока они не смогут показать, что технология точна, не дискриминирует и соответствует определенным стандартам конфиденциальности и защиты данных. Она привела несколько судебных дел в США и Австралии, где искусственный интеллект применялся ошибочно. В свою очередь, европейские регуляторы уже приняли правила, запрещающие сканирование черт лица в режиме реального времени, и другое использование ИИ, которое может угрожать безопасности или правам людей [3].

По мнению автора, для преодаления указанных трудностей нейронных сетей необходима разработка нового математического аппарата, который бы позволил более адекватно описывать такие аспекты естественного интеллекта как ментальные образы, смысловое содержание информации и т.п., с помощью которых может быть заложено «понимание окружающего мира» для ГНС [1]. С этой целью автором была предложена альтернативная математическая модель представления и обработки качественной и семантической информации, которая была названа моделью пятен [4–7]. Пятнами являются математические объекты с элементарными пространственными свойствами, для которых существуют их внутренние области и внешние окружения, а также определены понятие логической связи между этими областями для различных пятен. Модель пятен строится на базе представления и обработки только качественной информации об элементарных отношениях между ними, таких как пересечение, раздельность, включение и неразличимость. Важно отметить, что четкие геометрические фигуры рассматриваются как частный, предельный случай пятен.

Вместо действительных чисел, введены L4 числа, представляющие 2×2 матрицы логических величин для кодирования элементарных пространственных отношений между пятнами. Также определены L4 векторы и L4 матрицы. Разработана основа математического аппарата над L4 матрицами и векторами, что позволяет моделировать рассуждения и обучение на уровне, свойственном человеческому интеллекту, не используя действительные числа. Этот подход может быть альтернативной базой для моделирования интеллектуальных процессов, включая классификацию и рассуждения.

Автором был изучен обширный материал российских и зарубежных научных публикаций [5], относящимися к другим математическим моделям аспекта неопределенности, включая исчисления связи регионов [8–13], грубые (приближенные) множества [14–16], анализ формальных понятий [17–19], нечеткие множества [20], концепции гранул и гранулярных вычислений [21–23] и мягкие множества [24–26]. Сравнение с указанными концепциями позволяет сделать вывод, что несмотря на наличие общих идей, предложенная модель пятен является наиболее общим и гибким подходом, имеющим преимущество, по сравнению с другими. Это связано, в частности с тем, что модель пятен, в отличие от перечисленных выше теорий, не опирается на теорию множеств, хотя отношения между пятнами и операции с ними применимы и для множеств, и для нечетких множеств. Но, в отличие от последних, пятна не состоят из элементов, точнее эти элементы заранее не известны. С другой стороны, с помощью концепции пятен можно выразить другие концепции, например, множества, нечеткие множества, грубые множества, гранулы [4, 5]. Заметим, что каждому пятну можно поставить в соответствие бесконечную совокупность множеств или нечетких множеств, неразличимых с ним по имеющейся качественной информации.

Предполагается, что с помощью модели пятен можно представить любую качественную и количественную информацию. Важным свойством модели пятен является то, что на ее базе могут быть представлены ментальные образы, которые позволяют строить смысловое, или семантическое, пространство. Предлагаемый математический аппарат L4 чисел, L4 векторов и L4 матриц, в принципе, позволяет моделировать рассуждения, а также – обучение (см. раздел 5). Важным свойством такого вида рассуждений является их близость к особенности человеческого мышления, а именно, выводы делаются на базе имеющихся знаний, а рассуждения обладают свойством немонотонной логики, когда получение новых знаний может изменить выводы. Все это является основанием для возможности рассматривания разрабатываемой теории пятен как перспективной математической базы универсального (сильного) искусственного интеллекта.

2. ОПЕРАЦИИ С L4 МАТРИЦАМИ И ВЕКТОРАМИ

Элементарные отношения двух пятен, такие как пересечение, раздельность, включение и неразличимость, могут быть формализованы с помощью логических L4 чисел [4–7]. Для пятен a , b и их окружений \tilde{a} , \tilde{b} , которые тоже являются пятнами, L4 число $\langle a|b \rangle$ определяется в виде таблицы:

$$\langle a|b \rangle = \begin{bmatrix} ab & a\tilde{b} \\ \tilde{a}b & \tilde{a}\tilde{b} \end{bmatrix}, \quad (1)$$

где ab , $a\tilde{b}$... означает логическую связь между соответствующими пятнами. Такие логические таблицы, которые мы называем L4 числами, позволяют различать 16 различных элементарных отношений между пятнами. Примеры элементарных соотношений, описываемых L4 числами, приведены в табл. 1. В рассматриваемой концепции элементарные отношения пятен несут качественную информацию о пятнах, но большой объем таких данных позволяет извлекать информацию более высокого уровня, в том числе численную.

Таблица 1. Пример элементарных отношений между пятнами

Элементарные отношения	$\langle a b \rangle$
Пересечение, $a \succ b$	$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$
Раздельность, $a \prec b$	$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$
Включение (больше), $a \succ b$	$\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$
Включение (меньше), $a \prec b$	$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$
Неразличимость, $a = b$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

Предлагаемая теория использует аналогию с линейной алгеброй и матричным анализом. Например, элементарные отношения пятен аналогичны скалярному произведению числовых векторов, где вместо действительных чисел используются L4 числа. Следуя этому пониманию, определим L4 вектор для пятна, где элементами являются L4 числами, представляющие элементарные отношения этого пятна с *базисными* пятнами. Заметим, что L4 вектор с «координатами» пятна на

базисе представляет образ (изображение) пятна на этом базисе. Например, изображение пятна a на базисе $\{x_i\}$ можно представить в виде вектора a с L4 координатами:

$$\mathbf{a} \equiv [\alpha_1; \alpha_2; \dots \alpha_n], \quad (2)$$

где L4 числа $\alpha_i = \langle a|x_i \rangle$, а символ “;” обозначает представление в форме вектор-столбец.

Для любого атомарного базиса $A = \{u_i\}$ [4] можно определить элементарное отношение $\langle a|b \rangle_A$ между пятнами a и b относительно этого базиса и скалярное произведение векторов $(\mathbf{a} \cdot \mathbf{b})$ по правилу

$$\langle a|b \rangle_A = (\mathbf{a} \cdot \mathbf{b}) = \left[\begin{array}{cc} \sum_{i=1}^n a u_i \cdot b u_i & \sum_{i=1}^n a u_i \cdot \tilde{b} u_i \\ \sum_{i=1}^n \tilde{a} u_i \cdot b u_i & \sum_{i=1}^n \tilde{a} u_i \cdot \tilde{b} u_i \end{array} \right], \quad (3)$$

где символ « \cdot » обозначает логическую конъюнкцию. Это же правило будем применять для ортогонального базиса U , то есть состоящего из множества $\{u_i\}$ отдельных пятен.

Для произвольного базиса $X = \{x_i\}$, где пятна x_i могут пересекаться, определение величины $\langle a|b \rangle_X$ более сложно. В этом случае, сначала нужно рассмотреть базис $U = \{u_i\}$ всех пересечений пятен x_i и определить векторы $\mathbf{a}_U = [\langle a|u_k \rangle]$ и $\mathbf{b}_U = [\langle b|u_k \rangle]$ на базисе U , который является ортогональным. Затем $\langle a|b \rangle_X$ определим с помощью правила (3).

Формально векторы \mathbf{a}_U и \mathbf{b}_U можно определить с помощью следующего матричного равенства:

$$\mathbf{a}_U = \langle U|X \rangle \cdot \mathbf{a}_X,$$

где $\langle U|X \rangle$ является L4 матрицей, которая преобразует векторы между базисами X и U .

Например, отношение $\langle a|u_k \rangle$ для

$$u_k = x_1 \wedge \tilde{x}_2 \wedge x_3 \dots \wedge \tilde{x}_n, \quad (4)$$

где $k = 101 \dots 0_2$ определим с помощью правила [5]:

$$\langle a|u_k \rangle = \left[\begin{array}{cc} a x_1 \cdot a \tilde{x}_2 \cdot \dots \cdot a \tilde{x}_n & a \tilde{x}_1 + a x_2 + \dots + a x_n \\ \tilde{a} x_1 \cdot \tilde{a} \tilde{x}_2 \cdot \dots \cdot \tilde{a} \tilde{x}_n & \tilde{a} \tilde{x}_1 + \tilde{a} x_2 + \dots + \tilde{a} x_n \end{array} \right]. \quad (5)$$

Для устранения неопределенности в форме размытой границы, которое дает алгоритм (5), нужно применить дополнительные правила для ортогональных пятен u_k , принадлежащих границе, что позволяет скорректировать соотношения $\langle a|u_k \rangle$ в (5):

$$\begin{aligned} \forall x_i \in X: \{x_i \langle \rangle a, u_k x_i = 0\} &\Rightarrow a \rangle u_k \\ \forall x_i \in X: \{x_i \langle a, u_k x_i = 0\} &\Rightarrow a \langle \rangle u_k' \end{aligned} \quad (6)$$

где символы $\langle \rangle$ и \rangle обозначают отношения разделение и включение (больше), соответственно (см. табл. 1).

Рассмотрим правило умножения для L4 матрицы $\langle Y|X \rangle$, где $X = \{x_i\}$ и $Y = \{y_j\}$ – произвольные базисы. Тогда определим сначала базис $U = \{u_i\}$ всех пересечений пятен x_i и y_j , а затем – умножение на L4 вектор \mathbf{a}_X в следующем виде:

$$\mathbf{a}_Y = \langle Y|X \rangle \mathbf{a}_X = \langle Y|U \rangle \langle U|X \rangle \mathbf{a}_X \equiv \langle Y|U \rangle (\langle U|X \rangle \mathbf{a}_X). \quad (7)$$

Отметим, что умножение матрицы $\langle Y|U \rangle$ на L4 вектор $\langle U|X \rangle \mathbf{a}_X$ можно определить с учетом правила (3).

3. ПРИМЕР РЕКОНСТРУКЦИИ ФОРМЫ ПЛОХО СТРУКТУРИРОВАННОГО ОБЪЕКТА

Правильность предложенного аппарата, а точнее – правил (5)–(6), продемонстрирована при решении обратной задачи – восстановления формы четких фигур, являющихся предельным случаем пятен, с использованием только качественных данных по результату сканирования. Пример неполно заданного плохо структурированного изображения изображения звезды показан на рис. 3,а, а результат реконструкции ее формы по данным сканирования кругом показан на рис. 3,б. Реконструкция проводилась с помощью обработки качественных данных (пересечение, раздельность или включение) относительно результатов сканирования круга диаметра 10. Размер пикселей 0,25, период сканирования круга – 0,5.

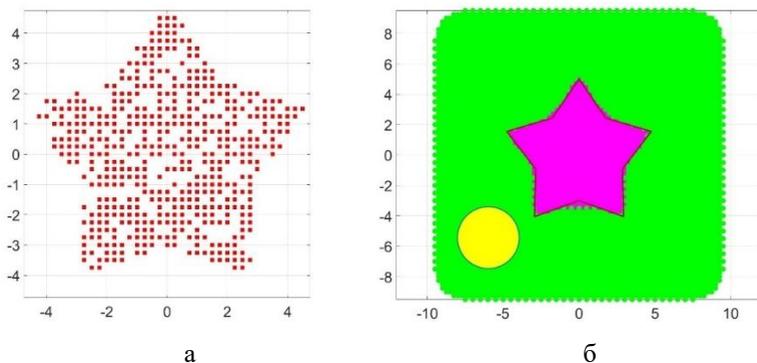


Рис. 1. Пример реконструкции звезды:
а – Плохо структурированное изображение звезды,
б – Реконструкция звезды по формулам (5) и (6)

4. ОБРАТНЫЕ L4 МАТРИЦЫ

Из определения L4 чисел, векторов и матриц следует что L4 матрица $\langle X|Y \rangle$, которая обратна матрице $\langle Y|X \rangle$, должна всегда существовать, и получается из $\langle Y|X \rangle$ путем транспонирования с дополнительным транспонированием ее элементов в виде L4 чисел (2x2 таблиц). Поэтому, решение уравнения

$$\mathbf{a}_Y = \langle Y|X \rangle \mathbf{a}_X$$

можно записать в виде

$$\mathbf{a}_X = \langle X|Y \rangle^{-1} \mathbf{a}_Y = \langle X|Y \rangle \mathbf{a}_Y = \langle X|U \rangle \langle U|Y \rangle \mathbf{a}_Y$$

где, как и для равенства (7), $U = \{u_i\}$ – базис всех пересечений пятен x_i и y_j .

5. МОДЕЛИРОВАНИЕ РАССУЖДЕНИЙ И ОБУЧЕНИЯ

На базе пятен возможно представление ментальных образов для понятий, идей, концепций или высказываний. Кроме того, с помощью разработанного аппарата можно моделировать процессы рассуждений и обучения.

По аналогии с естественным языком, понятие, представляемое пятном, можно описать с помощью его элементарных отношений с другими, базисными пятнами, которые соответствуют некоторым известным понятиям. Для любого ментального образа, который представлен пятном a , L4 вектор \mathbf{a} , определенный на этом базисе, будет характеризовать смысл этого понятия (рис. 2). Таким образом, L4 векторы позволяют моделировать ментальные образы (смыслы).

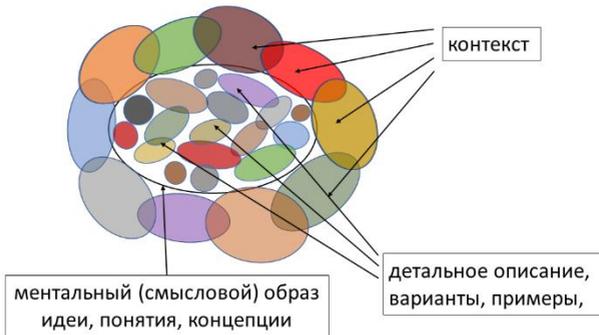


Рис. 2. Диаграмма Эйлера-Венна для представления ментального (смыслового) образа с помощью модели пятен

Используя L4 матрицы, можно моделировать рассуждения. Например, для L4 матрицы \mathbf{A} равенство

$$\mathbf{b} = \mathbf{A} \cdot \mathbf{a}$$

в теории пятен имеет смысл трансформации изображения пятна из одного базиса в другой. В применении к ментальным образам, это равенство моделирует рассуждение: из смысла высказывания \mathbf{a} следует вывод \mathbf{b} на основании информации (знаний) \mathbf{A} :

$$\mathbf{a} \rightarrow_{\mathbf{A}} \mathbf{b}.$$

Таким образом, в области обработки смысловой информации L4 матрицы соответствуют определенным знаниям. Важным свойством такого вида рассуждений является их близость к особенностям человеческого мышления, а именно:

- Выводы делаются на базе имеющихся знаний.
- Рассуждения обладают свойством *немонотонной логики*, когда получение новых знаний может изменить выводы.

Обучение нейронной сети для распознавания образов с использованием обучающих примеров можно трактовать как объединение различных изображений исследуемого объекта в один класс, который связан с ментальным образом объекта. Этот образ соответствует некоторому пятну, а различные его изображения – отображениям этого образа на разные базисы пятен.

Процесс обучения в предлагаемой модели можно отождествить с задачей нахождения неизвестной матрицы знаний \mathbf{A} . Допустим, мы имеем ряд обучающих примеров $\{\mathbf{x}_i, \mathbf{y}_i\}$, которые соответствуют равенствам:

$$\mathbf{y}_i = \mathbf{A} \cdot \mathbf{x}_i, \quad (10)$$

где матрица \mathbf{A} неизвестна. Для ее нахождения составим из L4 векторов $\mathbf{x}_i, \mathbf{y}_i$ новые L4 матрицы – \mathbf{X}, \mathbf{Y} , соответственно, и запишем уравнения, следующие из (10):

$$\mathbf{Y} = \mathbf{A} \cdot \mathbf{X}. \quad (11)$$

Его решение в виде

$$\mathbf{A} = \mathbf{Y} \cdot \mathbf{X}^{-1} \quad (12)$$

и представляет формальную интерпретацию процесса обучения. Для нахождения этого решение можно использовать разработанный математический аппарат для L4 матриц.

6. ЗАКЛЮЧЕНИЕ

Предлагаемая модель пятен и математический аппарат L4 чисел, L4 векторов и L4 матриц позволяют отображать и обрабатывать семантическую (смысловую) информацию, содержащуюся в данных различной природы. Эта важная особенность предлагаемой теории дает возможность рассматривать введенный математический аппарат как основу для представления знаний и моделирования человеческого мышления в области универсального (сильного) искусственного интеллекта.

На базе модели пятен могут быть представлены ментальные образы, которые являются элементами для построения смыслового, или семантического, пространства. Разработанный математический аппарат может быть применен во многих других областях искусственного интеллекта, например, в представлении знаний, компьютерном мышлении, распознавании образов, машинном обучении и для таких сложных задач как интеллектуальный анализ данных.

Литература

1. Heaven D. Deep trouble for deep learning // Nature. – 2019. – Vol. 574, No. 7777. – P. 163-166.
2. Image Classification on ImageNet// Paperswithcode.com. – <https://paperswithcode.com/sota/image-classification-on-imagenet>.
3. Keaten J. and O'Brien M. UN urges moratorium on use of AI that imperils human rights // AP News. – 2021. Sep. 15. – <https://apnews.com/article/technology-business-laws-united-nations-artificial-intelligence-efafd7b1a5bf47afb1376e198842e69d>.
4. Симонов, Н. А. Концепция пятен для задач искусственного интеллекта и алгоритмов нейроморфных систем // Микроэлектроника. – 2020. – Т. 49, № 6. – С. 459-473.
5. Симонов Н.А. Сравнение модели пятен с другими моделями качественных данных и размытых объектов // Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов X-й Международной научно-технической конференции (ИММВ-2021, Коломна, 17-20 мая 2021 г.). В 2-х т. Т.1. 2021. – Смоленск: Универсум. – С. 337-347.
6. Симонов, Н.А. Модель пятен как основа для качественных рассуждений и гранулярных вычислений // 19 Национальная конференция по искусственному интеллекту с международным участием (КИИ-2021). – 2021. – С. 267-284.
7. Simonov N. The Spot Model for Representation and Processing of Qualitative Data and Semantic Information / V. V. Borisov, B. A. Kobrinskii (Eds. //, Proceedings of Selected Contributions to the Russian Advances in Artificial Intelligence Track at RCAI 2021, Taganrog, Russia, October 11-16. 2021. CEUR-WS. US. 16-Dec-2021. – P. 55-69. – Urn:nbn:de:0074-3044-4.

8. Clarke B.L. A calculus of individuals based on connection // *Notre Dame Journal of Formal Logic*. – 1981. – Vol. 22. – P. 204-218.
9. Egenhofer M.J. and Franzosa R. Point-set topological spatial relations // *International Journal of Geographical Information Systems*. – 1991. – No. 5. – P. 161-174.
10. Egenhofer M.J. and Franzosa R. On the equivalence of topological relations// *International Journal of Geographical Information Systems*. – 1995. – No. 9. – P. 133-152.
11. Cohn A.G. and Gotts N.M. The ‘Egg–Yolk’ representation of regions with indeterminate boundaries // *Geographic Objects with Indeterminate Boundaries, GISDATA Series. Vol. 2.* / eds. P.A. Burrough and A.U. Frank (Taylor & Francis). – 1996. – P. 171-187.
12. Cohn A.G., Bennett B., Gooday J.M., and Gotts N. RCC: a calculus for region based qualitative spatial reasoning // *GeoInformatica*. – 1997. – Vol. 1, No. 3. – P. 275-316.
13. Stell J.G. Part and complement: Fundamental concepts in spatial relations// *Annals of Mathematics and Artificial Intelligence*. – 2004. – No. 41. – P. 1-17.
14. Pawlak Z. Rough sets// *International Journal of Computer and Information Sciences*. – 1982. – No. 11. – P. 341-356.
15. Pawlak Z. Rough set theory and its applications to data analysis // *Cybernetics and Systems*. – 1998. – Vol. 29, No. 7. – P. 661-688.
16. Yao Y.Y. Rough set approximations: A concept analysis point of view// *Computational Intelligence*; Ishibuchi H. Ed. – 2015. - Vol. I. *Encyclopedia of Life Support Systems (EOLSS)*. – P. 282-296.
17. Wille R. Restructuring lattice theory: An approach based on hierarchies of concepts // *Ordered Sets*. Rival, I. Ed.; Reidel: Dordrecht, Netherland. – 1982. – P. 445-470.
18. Ganter B., Wille R. *Formal Concept Analysis: Mathematical Foundations*. – New York, US: Springer, 1999.
19. Wille R. *Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies* // *Formal Concept Analysis, LNAI 3626*; Ganter, B. et al. Eds. – Springer-Verlag: Berlin, Heidelberg. – 2005. – P. 1-33.
20. Zadeh L.A. Fuzzy sets// *Inf. Control*. – 1965. – Vol. 8, No. 3. – P. 338-35.
21. Zadeh L.A. Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic // *Fuzzy Sets Syst*. – 1997. – No. 19. – P. 111-127.
22. Yao Y.Y. Information granulation and rough set approximation// *International Journal of Intelligent Systems*. – 2001. – Vol. 16, No. 1. – P. 87-104.
23. Yao J.T., Vasilakos A.V., and Pedrycz W. Granular Computing: Perspectives and Challenges // *IEEE Trans. on Cybernetics*. – 2013. – Vol. 43, No. 6. – P. 1977-1989.
24. Molodtsov D. Soft set theory—first results // *Computers & Mathematics with Applications*. – 1999. – Vol. 37, No. 4-5. – P. 19-31.
25. Молодцов Д.А. Теория мягких множеств. – М.: УРСС, 2004.
26. Молодцов Д.А. Структура мягких множеств// *Нечеткие системы и мягкие вычисления*. – 2017. – Т. 12, № 1. – С. 5-18.

МЕТОДЫ ВЫВОДА ДЛЯ СИСТЕМ ТИПА МАМДАНИ И ТАКАГИ–СУГЕНО С НЕСИНГЛТОННОЙ ФАЗЗИФИКАЦИЕЙ НА ОСНОВЕ НЕЧЕТКОГО ЗНАЧЕНИЯ ИСТИННОСТИ*

*Синюк В.Г., к.т.н., доцент
БГТУ им. В.Г. Шухова
e-mail: vgsinuk@mail.ru
Кулабухов С.В., аспирант
БГТУ им. В.Г. Шухова
e-mail: qlba@ya.ru*

1. ВВЕДЕНИЕ

Нечеткие системы рассматриваются как композиция фаззификатора, базы знаний, модуля вывода и дефаззификатора. Разнообразные комбинации методов вывода, фаззификации и дефаззификации приводят к многочисленным конструкциям нечетких систем. Наиболее популярная база знаний получена из первых исследований Мамдани [1], где как в антецедентной, так и в консеквентной частях правил используются нечеткие множества. Эта форма эволюционирует до формы типа Такаги-Сугено [2], где в консеквентной части правил нечеткие множества заменяются четкими функциями.

В последнее время при разработке нечетких систем уделяется внимание изменению метода фаззификации. Фаззификация – это отображение из точки входного пространства в нечеткое множество. Основным методом является синглтонная фаззификация [3]. Она обычно применяется всегда при любом характере входных данных. Основным преимуществом такого подхода является существенное упрощение реализации методов вывода нечетких систем, особенно для нейро-нечетких систем. Альтернативным решением является несинглтонная (NS) фаззификация [4–6].

NS-фаззификация используется в нечетких системах на основе правил, когда измерения, которые их активируют, несовершенны или неопределенны (из-за шума измерений, дефектов или ухудшения качества датчиков и т.д.), или когда их входными данными являются слова (как при вычислениях со словами). Она моделирует такие измерения или слова нечеткими числами или более общими нечеткими множест-

* Работа выполнена при финансовой поддержке РФФИ, проект №20-07-00030.

вами, так что, независимо от причины несовершенства или неопределенности измерения или слова, они рассматриваются в рамках нечетких множеств и систем.

Фаззификация NS была введена и исследована для нечетких систем типа Мамдани и Такаги-Сугено в частном случае в [6–8]. Данная статья посвящена развитию данного подхода для более общей постановки задачи. Проблема состоит в том, что в этом случае для нечетких систем MISO-структуры реализация нечеткого вывода с использованием известных методов приводит к экспоненциальной вычислительной сложности. В статье рассмотрены методы вывода на основе нечеткого значения истинности, снижающего вычислительную сложность до полиномиальной.

В разделе 2 осуществлена постановка задачи и оценка сложности при NS-фаззификации нечеткими числами при традиционном подходе. В разделе 3 представлены методы вывода для нечетких систем Мамдани и Такаги-Сугено с применением нечеткого значения истинности, что позволило снизить вычислительную сложность до полиномиальной и обобщить задачу вывода при NS-фаззификации. Раздел 4 посвящен применению методов дефаззификации к результатам, полученным в разделе 3.

2. ПОСТАНОВКА ЗАДАЧИ

Лингвистическая модель представляет собой базу нечетких правил R_k , $k = \overline{1, N}$ вида

$$R_k : \text{Если } x_1 \text{ есть } A_{1k} \text{ и } x_2 \text{ есть } A_{2k} \text{ и } \dots \text{ и } x_n \text{ есть } A_{nk}, \text{ то } y \text{ есть } B_k, \quad (1)$$

где N – количество нечетких правил, $A_{ik} \subseteq X_i$, $i = \overline{1, n}$, $B_k \subseteq Y$ – нечеткие множества, которые характеризуются функциями принадлежности $\mu_{A_{ik}}(x_i)$ и $\mu_{B_k}(y)$ соответственно; x_1, x_2, \dots, x_n – входные переменные лингвистической модели, причем

$$[x_1, x_2, \dots, x_n]^T = \mathbf{x} \in X_1 \times X_2 \times \dots \times X_n.$$

Символами X_i , $i = \overline{1, n}$ и Y обозначаются соответственно пространства входных и выходной переменных. Если ввести обозначения $X = X_1 \times X_2 \times \dots \times X_n$ и $A_k = A_{1k} \times A_{2k} \times \dots \times A_{nk}$, причем

$$\mu_{A_k}(\mathbf{x}) = T_1 \mu_{A_{ik}}(x_i),$$

где T_1 – произвольная t-норма, то правило (1) представляется в виде нечеткой импликации

$$R_k : A_k \rightarrow B_k, \quad k = \overline{1, N}.$$

Правило R_k можно формализовать как нечеткое отношение, определенное на множестве $X \times Y$, т.е. $R_k \subseteq X \times Y$ – нечеткое множество с функцией принадлежности

$$\mu_{R_k}(\mathbf{x}, y) = \mu_{A_k \rightarrow B_k}(\mathbf{x}, y).$$

Модель Мамдани определяет задание функции $\mu_{A_k \rightarrow B_k}(\mathbf{x}, y)$ на основе известных функций принадлежности $\mu_{A_k}(\mathbf{x})$ и $\mu_{B_k}(y)$ следующим образом [3,9]:

$$\mu_{A_k \rightarrow B_k}(\mathbf{x}, y) = T_2(\mu_{A_k}(\mathbf{x}), \mu_{B_k}(y)) = \mu_{A_k}(\mathbf{x})^{T_2} * \mu_{B_k}(y),$$

где T_2 – произвольная t-норма.

Ставится задача определить нечеткий вывод $B'_k \subseteq Y$ для системы, представленной в виде (1), если на входах – нечеткие множества

$$A' = A'_1 \times A'_2 \times \dots \times A'_n \subseteq X \quad \text{или} \quad x_1 \text{ есть } A'_1 \text{ и } x_2 \text{ есть } A'_2 \text{ и } \dots \text{ и } x_n \text{ есть } A'_n$$

с соответствующей функцией принадлежности $\mu_{A'}(\mathbf{x})$, которая определяется как

$$\mu_{A'}(\mathbf{x}) = T_3 \prod_{i=1, n} \mu_{A'_i}(x_i).$$

Несинглтонный фаззификатор отображает измеренное $x_i = x'_i$, $i = \overline{1, n}$ в нечеткое число, для которого $\mu_{A'_i}(x'_i) = 1$ и $\mu_{A'_i}(x_i)$ уменьшается от единицы по мере удаления от x'_i .

Концептуально [8] NS фаззификация подразумевает, что данное входное значение x'_i есть значение, которое будет верным из всех значений, находящихся рядом; однако, так как вход является неопределенным, соседние значения также могут быть верными, но в меньшей степени.

В соответствии с обобщенным нечетким правилом *modus ponens* [3], нечеткое множество B'_k определяется композицией нечеткого множества A' и отношения R_k , т.е.

$$B'_k = A' \circ (A_k \rightarrow B_k),$$

или, на уровне функций принадлежности,

$$\mu_{B'_k}(y | \mathbf{x}') = \sup_{\mathbf{x} \in X} \left\{ \mu_{A'_k}(\mathbf{x}) \overset{T_1}{*} \left(\mu_{A_k}(\mathbf{x}) \overset{T_2}{*} \mu_{B_k}(y) \right) \right\}, \quad (2)$$

где T_2 может быть любым оператором в классе t -норм.

В (2) применена условная нотация, так как ввод в нечеткую систему происходит при определенном значении \mathbf{x} , а именно \mathbf{x}' . Обозначение $\mu_{B'_k}(y | \mathbf{x}')$ показывает, что $\mu_{B'_k}$ изменяется с каждым значением \mathbf{x}' .

Сложность выражения (2) составляет $O(|X_1| \cdot |X_2| \cdot \dots \cdot |X_n| \cdot |Y|)$.

3. МЕТОДЫ ВЫВОДА С ИСПОЛЬЗОВАНИЕМ НЕЧЕТКОГО ЗНАЧЕНИЯ ИСТИННОСТИ (НЗИ)

Для снижения экспоненциальной сложности (2) рассмотрим следующие методы вывода.

3.1. МЕТОД ВЫВОДА, КОГДА Т-НОРМЫ РАЗЛИЧНЫ

Применяя правило истинностной модификации [10]

$$\mu_{A'}(\mathbf{x}) = \tau_{A_k|A'}(\mu_{A_k}(\mathbf{x})),$$

где $\tau_{A_k|A'}(\cdot)$ – нечеткое значение истинности (нзи) нечеткого множества A_k относительно A' , представляющее собой функцию принадлежности совместимости $CP(A_k, A')$ A_k по отношению к A' , причем A' рассматривается как достоверное [11]:

$$\tau_{A_k|A'}(v) = \mu_{CP(A_k, A')}(v) = \sup_{\substack{\mu_{A_k}(\mathbf{x})=v \\ \mathbf{x} \in X}} \left\{ \mu_{A'}(\mathbf{x}) \right\},$$

перейдем от переменной \mathbf{x} к переменной v , обозначив $\mu_{A_k}(\mathbf{x}) = v$.

Получим:

$$\mu_{A'}(\mathbf{x}) = \tau_{A_k|A'}(\mu_{A_k}(\mathbf{x})) = \tau_{A_k|A'}(v), \quad (3)$$

тогда (2) примет вид:

$$\mu_{B'_k}(y | \mathbf{x}') = \sup_{v \in [0;1]} \left\{ \tau_{A_k|A'}(v) \overset{T_1}{*} \left(v \overset{T_2}{*} \mu_{B_k}(y) \right) \right\}. \quad (4)$$

При вербализации «инженерной» импликации в (4) она представится в виде:

$$\text{Если нзи есть истинно, то } u \text{ есть } B'_k. \quad (5)$$

Таким образом, (5) представляет собой еще одну структуру правил в отличие от канонических структур Заде [12] и Такаги-Сугено [2]. Применение данного правила не зависит от количества входов в нечетких системах.

Как следует из (4), вывод B'_k есть результат композиции полученного значения *нзи* между антецедентом правила (1) A_k относительно входов A' и «инженерной» импликации (5). Такое преобразование реализует вывод в рамках единого пространства истинности, упрощает вычисление составной функции *нзи* (3) и лишено проблем многомерного анализа (2).

Выражение (4) характеризуется сложностью порядка $O(|v| \cdot |Y|)$. Как следует из [13, 14]:

$$\begin{aligned} \mu_{CP(A_k, A')}(v) &= \tilde{T}_1 \mu_{CP(A_{k1}, A'_1)}(v_1) = \\ &= (\dots ((\mu_{CP(A_{k1}, A'_1)}(v_1) \tilde{T}_1 \mu_{CP(A_{k2}, A'_2)}(v_2)) \tilde{T}_1 \mu_{CP(A_{k3}, A'_3)}(v_3)) \tilde{T}_1 \dots \\ &\quad \dots) \tilde{T}_1 \mu_{CP(A_{kn}, A'_n)}(v_n), \end{aligned}$$

где \tilde{T}_1 – расширенная по принципу обобщения n -местная t -норма и

$$\mu_{CP(A_{ki}, A'_i)}(v_i) = \sup_{\substack{\mu_{A'_i}(x_i) = v_i \\ x_i \in X_i}} \{ \mu_{A_{ki}}(x_i) \}.$$

Например, двухместная расширенная t -норма имеет вид:

$$\mu_{CP(A_k, A')}(v) = \tilde{T}_1 \mu_{CP(A_{k1}, A'_1)}(v_1) = \sup_{\substack{v_1 \tilde{T}_1 v_2 = v \\ (v_1, v_2) \in [0,1]^2}} \{ \mu_{CP(A_{k1}, A'_1)}(v_1) \tilde{T}_3 \mu_{CP(A_{k2}, A'_2)}(v_2) \}.$$

Последнее соотношение характеризуется сложностью порядка $O(|v|^2)$.

3.2. МЕТОД ВЫВОДА, ЕСЛИ $T_4 = T_2 = T$

С учетом свойства ассоциативности t -норм выражение (4) примет вид:

$$\begin{aligned} \mu_{B'_k}(y|x') &= \sup_{v \in [0,1]} \{ \tau_{A_k|A'}(v) \overset{T}{*} (v \overset{T}{*} \mu_{B_k}(y)) \} = \sup_{v \in [0,1]} \{ (\tau_{A_k|A'}(v) \overset{T}{*} v) \overset{T}{*} \mu_{B_k}(y) \} = \\ &= \sup_{v \in [0,1]} \{ \tau_{A_k|A'}(v) \overset{T}{*} v \} \overset{T}{*} \mu_{B_k}(y) = \Pi_{A_k|A'} \overset{T}{*} \mu_{B_k}(y), \end{aligned} \quad (6)$$

где

$$\Pi_{A_k|A'} = \sup_{v \in [0;1]} \{ \tau_{A_k|A'}(v)^{*} v \}^T$$

есть скалярная величина, которая является мерой возможности [15], т.е. насколько antecedent правила A_k соответствует входам A' . Возможность здесь определяется как мера совместимости полученного значения *нзи* (3) со значением *истинно*.

Как следует из [7], $\Pi_{A_k|A'} \in [0;1]$ есть уровень срабатывания (firing level) правила R_k , который меняется с каждым значением x' .

Таким образом, применение нечеткого значения истинности при методе вывода (6) получен при различных T_1 , T_3 и $T = T_2 = T_4$. Данный результат обобщает результат, полученный в [7,8], когда имеет место ограничение на равенство всех t-норм.

3.3. МЕТОД ВЫВОДА ДЛЯ НЕЧЕТКИХ СИСТЕМ ТИПА ТАКАГИ-СУГЕНО (ТС)

Описание нечетких систем ТС реализуется с помощью правил (1), консеквент которых имеет вид:

$$\mu_{B_k}(y) = \begin{cases} 1, & \text{если } y = f_k(x), \\ 0, & \text{если } y \neq f_k(x), \end{cases} \quad (7)$$

где $f_k(x)$ – функция, имеющая одинаковую для всех правил $k = \overline{1, N}$ структуру и различающаяся параметрами.

Учитывая (7) следует, что для правил ТС (6) преобразуется к виду:

$$\mu_{B'_k}(y|x') = \mu_{B'_k}(x') = \Pi_{A_k|A'}, \quad \text{если } y = f_k(x').$$

Отметим, что для нечеткой системы ТС, $\mu_{B'_k}(x')$ является явной функцией x' и неявной функцией y , где $x' \in X_1 \times X_2 \times \dots \times X_n$. Явное применение консеквента $f_k(x')$ будет реализовано на этапе дефаззификации (см. подраздел 4.4).

Таким образом, с учетом результатов подразделов 3.2 и 3.3:

Теорема. Пусть x' является входом в нечеткую систему. Тогда функция принадлежности B'_k с учетом уровня срабатывания правила R_k , $k = \overline{1, N}$ равна:

$$\left\{ \begin{array}{l} \text{для систем Мамдани: } \mu_{B'_k}(y|\mathbf{x}') = \Pi_{A_k|A'}^{\top} \mu_{B_k}(y); \\ \text{для систем ТС: } \mu_{B'_k}(\mathbf{x}') = \Pi_{A_k|A'}, \text{ если } y = f_k(\mathbf{x}'). \end{array} \right. \quad (8)$$

4. ДЕФАЗЗИФИКАЦИЯ

Наиболее часто используются в реальных приложениях [7] для нечеткой системы Мамдани дефаззификаторы: по среднему центру (Center Average, CA), по центру множеств (Center-of-Sets, CoS) и метод центра тяжести (Center of Area, CoA).

Два вида дефаззификации для нечеткой системы ТС описаны в подразделе 4.4.

4.1. ДЕФАЗЗИФИКАЦИЯ ПО СРЕДНЕМУ ЦЕНТРУ

Выходное значение нечеткой системы с учетом (8) определяется [3]:

$$\bar{y}_{CA}(\mathbf{x}') = \frac{\sum_{k=1, \bar{N}} \bar{y}_k \cdot \mu_{B'_k}(\bar{y}_k|\mathbf{x}')}{\sum_{k=1, \bar{N}} \mu_{B'_k}(\bar{y}_k|\mathbf{x}')} = \frac{\sum_{k=1, \bar{N}} \bar{y}_k \cdot \Pi_{A_k|A'}^{\top} \mu_{B_k}(\bar{y}_k)}{\sum_{k=1, \bar{N}} \Pi_{A_k|A'}^{\top} \mu_{B_k}(\bar{y}_k)}. \quad (9)$$

В (9), \bar{y}_k – центры функций принадлежности $\mu_{B_k}(y)$, $k = \bar{1}, \bar{N}$, т.е. значения в которых:

$$\mu_{B_k}(\bar{y}_k) = \sup_{y \in \bar{Y}} \{\mu_{B_k}(y_k)\} = 1, \quad (10)$$

тогда (9) примет вид:

$$\bar{y}_{CA}(\mathbf{x}') = \frac{\sum_{k=1, \bar{N}} \bar{y}_k \cdot \Pi_{A_k|A'}}{\sum_{k=1, \bar{N}} \Pi_{A_k|A'}} \quad (11)$$

т.к. t-норма по определению удовлетворяет граничному условию $T(a, 1) = a$.

Проблема дефаззификации по среднему центру возникает, когда функция принадлежности консеквента правила R_k имеет максимальное значение при $y = 0$ (например, функция принадлежности самого левого терма B_k). Такое правило будет обнуляться следуя (11). Эта проблема устраняется дефаззификатором CoS.

4.2. ДЕФАЗЗИФИКАЦИЯ ПО ЦЕНТРУ МНОЖЕСТВ

При этой дефаззификации [16] каждая функция принадлежности консеквента заменяется синглтоном, расположенным в его центре тяжести (CoA) с амплитудой, равной уровню срабатывания (firing level) правила R_k , после чего определяется центроид данных синглтонов.

Выражение для вывода с CoS дефаззификацией задается:

$$\bar{y}_{\text{CoS}}(\mathbf{x}') = \frac{\sum_{k=1, N} \text{CoA}(B_k) \cdot \Pi_{A_k|A'}}{\sum_{k=1, N} \Pi_{A_k|A'}} = \frac{\sum_{k=1, N} c_k \cdot \Pi_{A_k|A'}}{\sum_{k=1, N} \Pi_{A_k|A'}}. \quad (12)$$

В (12), c_k есть центроид функции принадлежности консеквента правила R_k . Они могут быть определены предварительно.

Если каждая функция принадлежности B_k является симметричной, нормальной и выпуклой, то $c_k = \bar{y}_k$, но для несимметричной функции принадлежности $c_k \neq \bar{y}_k$. Поэтому в общем случае $\bar{y}_{\text{CoS}}(\mathbf{x}') \neq \bar{y}_{\text{CA}}(\mathbf{x}')$.

4.3. ДЕФАЗЗИФИКАЦИЯ НА ОСНОВЕ МЕТОДА ЦЕНТРА ТЯЖЕСТИ

Дефаззификатор CoA определяет четкий вывод нечеткой системы [3]:

$$\bar{y}_{\text{CoA}}(\mathbf{x}') = \frac{\sum_{k=1, N} \bar{y}_k \cdot \mu_{B'}(\bar{y}_k | \mathbf{x}')}{\sum_{k=1, N} \mu_{B'}(\bar{y}_k | \mathbf{x}')}. \quad (13)$$

Нечеткое множество B' получается как результат объединения нечетких множеств B'_k , $k = \overline{1, N}$ с помощью оператора \max или других s-норм, т.е.

$$\mu_{B'}(y | \mathbf{x}') = \bigvee_{j=1, N} \mu_{B'_j}(y | \mathbf{x}'). \quad (14)$$

Тогда (13) с учетом (14) и (8) примет вид:

$$\bar{y}_{\text{CoA}}(\mathbf{x}') = \frac{\sum_{k=1, N} \bar{y}_k \cdot \bigvee_{j=1, N} \mu_{B'_j}(\bar{y}_k | \mathbf{x}')}{\sum_{k=1, N} \bigvee_{j=1, N} \mu_{B'_j}(\bar{y}_k | \mathbf{x}')} = \frac{\sum_{k=1, N} \bar{y}_k \cdot \bigvee_{j=1, N} \{ \Pi_{A_j|A'}^T * \mu_{B_j}(\bar{y}_k) \}}{\sum_{k=1, N} \bigvee_{j=1, N} \{ \Pi_{A_j|A'}^T * \mu_{B_j}(\bar{y}_k) \}}. \quad (15)$$

Обозначим $\mu_{B_j}(\bar{y}_k) = b_{jk}$. Согласно (10), $b_{kk} = \mu_{B_k}(\bar{y}_k) = 1$. Тогда (15):

$$\bar{y}_{\text{CoA}}(\mathbf{x}') = \frac{\sum_{k=1, N} \bar{y}_k \cdot \bigwedge_{j=1, N} \{ \Pi_{A_j|A'} * b_{jk} \}}{\sum_{k=1, N} \bigwedge_{j=1, N} \{ \Pi_{A_j|A'} * b_{jk} \}}. \quad (16)$$

Если $b_{jk} = 0$ для $j, k = \overline{1, N}$, $j \neq k$, то (16) примет вид выражения (11). Условия $b_{jk} = 0$ соответствуют условиям, в которых консеквенты правил R_k представляют собой нечеткие множества-синглтоны, либо функции принадлежности B_k не пересекаются.

4.4. ДЕФАЗЗИФИКАЦИЯ НЕЧЕТКИХ СИСТЕМ ТИПА ТС

Один вид дефаззификатора для нечетких систем ТС приводит к ненормированной нечеткой системе [2], для которой:

$$\bar{y}_{\text{TS}}(\mathbf{x}') = \sum_{k=1, N} \Pi_{A_k|A'} \cdot f_k(\mathbf{x}').$$

Второй вид дефаззификатора приводит к нормированной нечеткой системе ТС:

$$\bar{y}_{\text{TS}}(\mathbf{x}') = \frac{\sum_{k=1, N} \Pi_{A_k|A'} \cdot f_k(\mathbf{x}')}{\sum_{k=1, N} \Pi_{A_k|A'}}.$$

В частном случае, когда $f_k(\mathbf{x}') = f_k$, т.е. является константой, структурно $\bar{y}_{\text{CoS}}(\mathbf{x}')$ и $\bar{y}_{\text{TS}}(\mathbf{x}')$ совпадают.

5. ЗАКЛЮЧЕНИЕ

Приведенные в статье методы вывода для нечетких систем типа Мамдани и Такаги-Сугено дают возможность применять их с несинглотонной фаззификацией и с полиномиальной вычислительной сложностью. Данные результаты достигнуты благодаря подходу, основанному на нечетком значении истинности. Следствием этого подхода является также новая структура лингвистических правил (5), применение которого не зависит от размерности входов в нечетких системах.

Литература

1. Mamdani E.H. Applications of Fuzzy Algorithm for Control a Simple Dynamic Plant // Proc. IEEE. – 1974. – Vol. 121. – No. 12. – P.1585-1588.
2. Takagi T., Sugeno M. Fuzzy Identification of Systems and Its Application to Modeling and Control // IEEE Transactions on Systems, Man and Cybernetics. – 1985. – Vol. 15. – No. 1. – P.116-132.
3. Рутковский Л. Методы и технологии искусственного интеллекта. – М.: Горячая линия – Телеком, 2010.
4. Pourabdollah A., John R., Garibaldi J. M. A new dynamic approach for non-singleton fuzzification in noisy time-series prediction // Proc. of FUZZ-IEEE. – 2017. – P. 1-6.
5. Fu C., Sarabakha A., Kayacan E., Wagner C., John R., Garibaldi J.M. Input Uncertainty Sensitivity Enhanced Nonsingleton Fuzzy Logic Controllers for Long-Term Navigation of Quadrotor UAVs // IEEE-ASME Transactions on Mechatronics. – 2018. – Vol. 23, No. 2. – P. 725-734.
6. Mendel J.M. Non-Singleton Fuzzification Made Simpler // Information Sciences. – 2021. – Vol. 559. – P. 286-308.
7. Mendel J.M. Uncertain Rule-Based Fuzzy Systems: Introduction and New Directions. – Second edition. – Cham, Switzerland: Springer, 2017.
8. Mouzouris G.C., Mendel J.M. Non-Singleton Fuzzy Logic Systems: Theory and Application // IEEE Transactions on Fuzzy Systems. – 1997. – Vol. 5, No. 1. – P. 56-71.
9. Пегат А. Нечеткое моделирование и управление. – М.: БИНОМ, Лаборатория знаний, 2009.
10. Борисов А.Н., Крунберг О.А., Федоров И.П. Принятие решений на основе нечетких моделей. – Рига: Зинатне, 1990.
11. Дюбуа Д., Прад А. Теория возможностей. Приложение к представлению знаний в информатике. – М.: Радио и связь, 1990.
12. Zadeh L.A. Outline of a New Approach to the Analysis of Complex Systems and Decision Processes // IEEE Transactions on Systems, Man and Cybernetics. – 1973. – Vol. 3, No. 1. – P. 28-44.
13. Синюк В.Г., Михелев В.В. Методы вывода для систем логического типа на основе нечеткой степени истинности // Известия РАН. Теория и системы управления. – 2018. – № 3. – С. 108-115.
14. Sinuk V.G., Polyakov V.M., Kutsenko D.A. New Fuzzy Truth Value Based Inference Methods for Non-singleton MISO Rule-Based Systems // Proceedings of the First International Scientific Conference “Intelligent Information Technologies for Industry” (IITI '16). – 2016.
15. Dubois D., Esteve F., Godo L., Prade H. Fuzzy-set Based Logics - An History-Oriented Presentation of Their Main Developments // The Many Valued and Nonmonotonic Turn in Logic. Eds: Gabbay M.D., John W. – Vol.8. Handbook of the History of Logic book series. – Elsevier, 2007. – Chapter 2.3 Fuzzy Truth-Values – Degree of Truth vs. Degree of Uncertainty. – P. 325-449.
16. Sugeno M., Yasukawa T. A Fuzzy-Logic-Based Approach to Qualitative Modeling // IEEE Transactions on Fuzzy Systems. – 1993. – No. 1. – P. 7-31.

ВЫЯВЛЕНИЕ ПРЕДВЕСТНИКОВ БИФУРКАЦИЙ ДИНАМИЧЕСКОЙ СИСТЕМЫ НА ОСНОВЕ АНАЛИЗА СТРУКТУРЫ ЕЕ НЕЧЕТКОЙ МОДЕЛИ*

Суханов А.В., к.т.н., зам. начальника отделения

АО «НИИАС»

ФГБОУ ВО РГУПС

ЧОУ ВО ЮУ (ИУБиП)

e-mail: a.suhanov@rfniias.ru

Ковалев С.М., д.т.н., профессор, г.н.с.

АО «НИИАС»

ФГБОУ ВО РГУПС

e-mail: ksm@rfniias.ru

Акперов И.Г., д.э.н., к.т.н., профессор, ректор

ЧОУ ВО ЮУ (ИУБиП)

e-mail: rector@iubip.ru

Ольгейзер И.А., к.т.н., начальник отделения

АО «НИИАС»

ФГБОУ ВО РГУПС

e-mail: iohan@rfniias.ru

1. ВВЕДЕНИЕ

Свойство двойной пластичности связано с работой иммунных систем [1, 2]. В настоящее время иммунные системы являются одним из наименее развитых реализаций биологических сред в области искусственного интеллекта (в сравнении с нейронными сетями и генетическими алгоритмами) [3] так как их первоначальное назначение – защитная функция [4].

Двойная пластичность связана с другой функцией иммунных систем, а именно с саморегуляцией. Система с саморегуляцией плавно изменяет свои параметры под воздействием онтогенетических изменений организма и в ответ на изменение внешней среды. Другими словами, иммунные модели могут успешно аппроксимировать поведение моделируемой системы даже при постоянных ее изменениях.

* Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 20-07-00100.

С такой точки зрения иммунные системы можно рассматривать как отдельную область искусственного интеллекта.

Саморегуляция обеспечивает плавные параметрические изменения, ведущие к эндогенным изменениям структуры. Параметрические изменения называются параметрической адаптацией (или параметрической пластичностью) и реализуют адаптивный механизм, позволяющий системе менять параметры (например, синаптические веса искусственной нейронной сети) в ходе выполнения своей задачи. Эндогенные изменения структуры называются структурной адаптацией (или структурной пластичностью) и обеспечивают системе адаптивность под значительные изменения внешней среды [5]. Структурная пластичность означает добавление новых и удаление избыточных элементов в системе. С точки зрения нейронных сетей это означает способность нейронов исчезать и возникать в нужное время [6]. Структурная пластичность позволяет изменять архитектуру всей системы. Отличительной особенностью структурных изменений как функции двойной пластичности является то, что они возникают только при «коллективном» требовании, например, для «помощи слабым элементам», «управления плотностью», «заполнения пустых мест» и пр.

Двойная пластичность возникает в процессе постоянных взаимодействий с эволюционирующей внешней средой. Она реализует постоянную эндогенную реорганизацию системы, позволяющую должным образом моделировать внешнюю среду. Таким образом, если иммунная система правильно настроена, она изменяется с изменением среды. В настоящее время, для анализа внешней среды используют анализ реакции иммунной системы на внешнее воздействие. Однако, за счет точной подстройки системы под внешнюю среду сложно заметить, а тем более предсказать развитие аномального поведения.

В настоящей работе предложен подход к превентивному обнаружению аномального поведения динамических систем, ведущего к смене ее аттрактора (точке бифуркации). Представлена актуальность использования анализа внутренней структуры модели для обнаружения нетипичного поведения моделируемой системы. Описан базовый способ построения базы нечетких правил, положенный в основу подхода к превентивному обнаружению смены аттрактора. В конце представлен иллюстративный пример со странным аттрактором Лоренца и приведены последующие этапы развития систем двойной пластичности.

2. ПОСТАНОВКА ЗАДАЧИ И АКТУАЛЬНОСТЬ ПРОБЛЕМЫ

В настоящей работе предлагается представить задачу упреждающего обнаружения нетипичного поведения (или аномальных событий) в развитии динамических процессов не в виде их прогнозирования их в будущем, а как задачу обнаружения в настоящем неких предвестников этих событий, представленных в виде характерных темпоральных паттернов временных рядов. Очевидно принципиальное различие в постановках этих задач, как и, естественно, в подходах к их решению. И при этом большая практическая значимость второго подхода обусловлена тем обстоятельством, что гарантированное прогнозирование будущего в нестационарных процессах и нелинейных системах принципиально невозможно. Для реализации подхода к упреждающему прогнозированию на основе анализа предикторов в первую очередь надо ответить на вопрос о принципиальном существовании таких предвестников-предикторов.

Доказательство актуальности заключалось в серии математических экспериментов, подтверждающих наличие особых паттернов во временных рядах в преддверии смены аттракторов или бифуркации. Рассмотрим в качестве примера временной ряд, описываемый логистическим отображением [7]:

$$x(t+1) = a \cdot x(t) \cdot (1 - x(t)), \quad (1)$$

где $a \in [1, \infty]$ – параметр скорости роста популяции, описываемой $x(t)$.

Для имитации точки бифуркации в экспериментах было использовано переменное значение $a(t)$ в виде ступенчатой функции от времени.

Для оценки временного ряда (1) была использована модель эволюционирующей нейронечеткой сети (eNFN, рис. 1) [8], обученной для разных значений a .

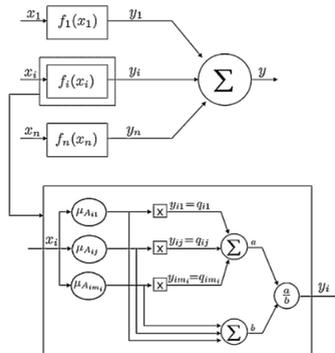


Рис. 1. Модель eNFN

В качестве «индикатора» бифуркации использовалось модальное значение правил. На рис. 2 представлена зависимость модальных значений нечетких правил от параметра скорости роста популяции.

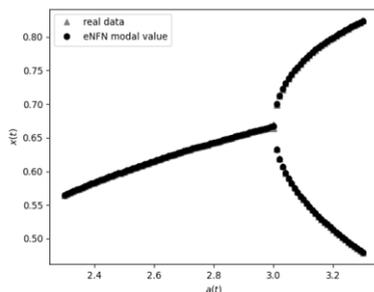


Рис. 2. Модальные значения eNFN для зашумленного логистического отображения

Очевидно, что в силу синтетической природы (1) логистическое отображение не имеет предвестников бифуркации, наступающей при $a = 3$. Поэтому необходимым являлось поиск и добавление дополнительных условий для внесения факторов порождения предвестников и приближения искусственных временных рядов к реальным. Как показали эксперименты, для провокации появления предвестников в синтетических временных рядах всего лишь необходимо было добавить белый шум в параметры порождающей системы. В случае логистического отображения на параметр a наложен шум $\varepsilon \in [-0.01; 0]$ (рис. 3).

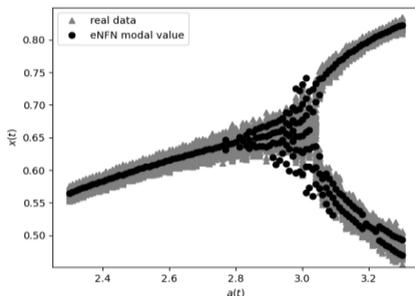


Рис. 3. Модальные значения eNFN для зашумленного логистического отображения

Результаты, представленные на рис. 3, доказывают появление предвестников бифуркации системы (при $a > 2.8$) в то время как реальные данные (ровно как и отклик eNFN) находятся в первом аттракторе.

3. ИНКРЕМЕНТНО ОБУЧАЕМАЯ МОДЕЛЬ ТАКАГИ-СУГЕНО

Вторая цель настоящего исследования заключалась в поиске моделей, способных обучаться инкрементно (или переобучаться в реальном времени) и реагировать на появление паттернов-предвестников аномальных событий.

Для ее достижения было решено использовать инкрементно изменяемую базу знаний в виде нечетких правил типа Сугено. Такие модели выбраны в виду их интерпретируемости, т.е. возможности понимания результатов обучения модели, ее структуры, и, следовательно, возможности анализа структурных изменений на предмет наличия особых паттернов.

В качестве способа обучения модели Сугено использован эволюционирующий способ обучения на потоковых данных, предложенный в [9] для создания адаптивной нейро-нечеткой сети (ets+, рис. 4).

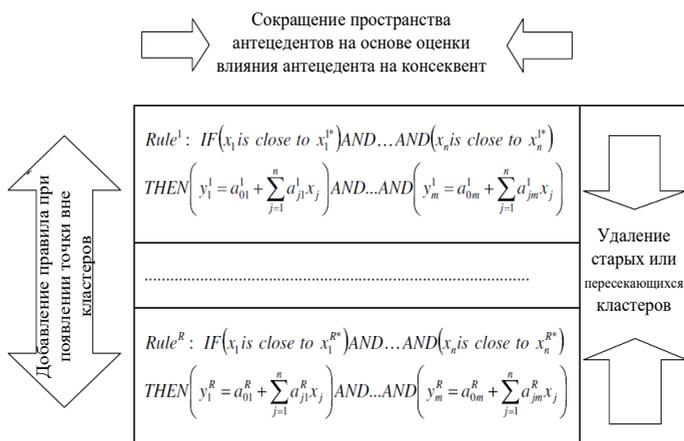


Рис. 4. Краткое представление ets+ модели

4. ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ

Для выявления способности ets+ модели к выявлению паттернов-предвестников точек бифуркации динамических систем был проведен эксперимент на динамической системе, описывающей странный аттрактор Лоренца:

$$\begin{cases} \dot{x} = \sigma(y - x) \\ \dot{y} = x(r - z) - y, \\ \dot{z} = xy - bz \end{cases} \quad (2)$$

где $\sigma=10$, $r=28$, $b=8/3$.

Для проявления бифуркации было решено плавно изменять параметр r в пределах $[20; 24,55]$ ($r(t)=\sin(f(t))$), когда решение все еще находится в двух устойчивых траекториях, но уже приближается к неустойчивым циклам аттрактора Лоренца (рис. 5):

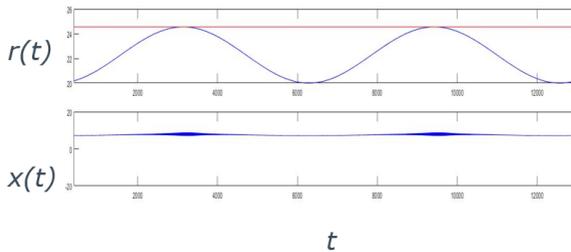


Рис. 5. Динамическая система (2) с незашумленным параметром r

Добавление белого шума $\varepsilon \in [-4; 0]$ инициирует случайный переход системы (2) в странный аттрактор до $r = 24.55$ (рис. 6).

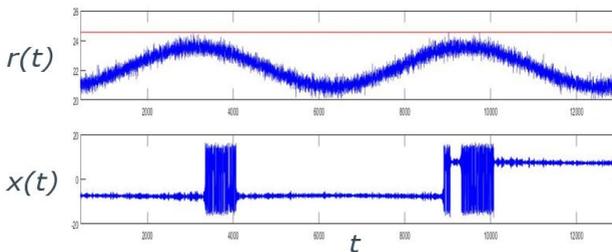


Рис. 6. Динамическая система (2) с зашумленным параметром r

Таким образом, задача эксперимента – анализ внутренней структуры модели на предмет выявления структурной адаптации при переходе системы из устойчивого состояния в странный аттрактор.

В качестве антецедентов ets+ модели были использованы 20 значений $x(t - \Delta t)$ с временным лагом 0.06с. При анализе изменения модальных значений правил было отмечено, что вблизи точки бифур-

кации число antecedентов в одной области сокращается, а в другой, наоборот, увеличивается (рис. 7), что может служить сигнализатором приближения системы к смене аттракторов. При этом отклик системы на входные данные не менялся.

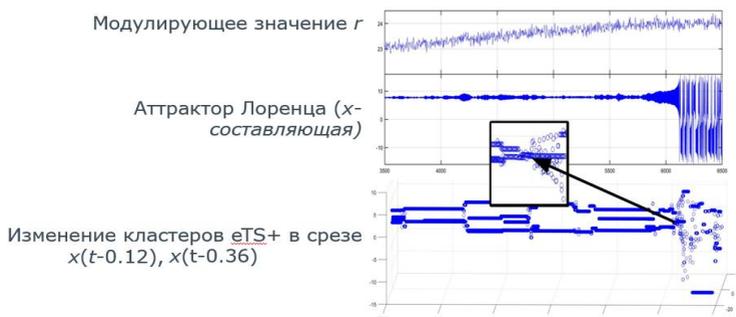


Рис. 7. Динамическая система (2) и срез моделирующей системы ϵ_{TS+}

Проведенный эксперимент в частном виде доказывает актуальность применения нечеткой обучаемой модели Сугено в задачах выявления паттернов-предвестников бифуркаций динамических систем.

5. ЗАКЛЮЧЕНИЕ

В настоящем исследовании решены 2 задачи:

Доказана актуальность обнаружения предвестников при прогнозировании целевых событий

Предложен подход к обнаружению предвестников точек бифуркации на основе нечетких моделей типа Сугено, способных инкрементно обучаться и реагировать на появление паттернов-предвестников целевых событий.

В дальнейшем планируется апробация разработанного подхода к упреждению нештатных ситуаций на железнодорожном транспорте, в частности, выявление неисправностей вагонных замедлителей на основе анализа динамических характеристик процесса расформирования железнодорожных составов. В теоретическом развитии планируется разработка метода автоматического выявления изменений структуры модели динамической системы с целью упреждения ее бифуркаций.

Литература

1. Bersini H. The endogenous double plasticity of the immune network and the inspiration to be drawn for engineering artifacts // *Artificial immune systems and their applications*. – Springer, Berlin, Heidelberg, 1993. – P. 22-44.
2. Rutkowski L. *Computational intelligence: methods and techniques*. – Springer Science & Business Media, 2008.
3. Брюхомицкий Ю.А. Модель адаптивной самоорганизующейся искусственной иммунной системы для решения задач компьютерной безопасности // *Известия ЮФУ. Технические науки*. – 2013. – № 12 (149).
4. Corus D., Oliveto P.S., Yazdani D. Artificial immune systems can find arbitrarily good approximations for the NP-hard number partitioning problem // *Artificial Intelligence*. – 2019. – Vol. 274. – P. 180-196.
5. Kovalev S., Kolodenkova A., Sukhanov A. Incremental structure-evolving intelligent systems with advanced interpretational properties // *Russian Conference on Artificial Intelligence*. – Springer, Cham, 2020. – P. 134-151.
6. Hirose Y., Yamashita K. and Hijiya S. Back-propagation Algorithm Which Varies the Number of Units // *Neural Networks*. – 1991. – Vol. 4. – P. 61-66
7. Weisstein E.W. Logistic map // <https://mathworld.wolfram.com/>. – 2001.
8. Silva A.M. et al. A fast learning algorithm for evolving neo-fuzzy neuron // *Applied Soft Computing*. – 2014. – Vol. 14. – P. 194-209.
9. Angelov P. Evolving takagi-sugeno fuzzy systems from streaming data (ets+) // *Evolving intelligent systems: methodology and applications*. – 2010. – Vol. 12. – P. 21.

ИССЛЕДОВАНИЕ ДИНАМИЧЕСКИХ СВОЙСТВ СИСТЕМЫ УПРАВЛЕНИЯ С НЕЧЕТКИМИ РЕГУЛЯТОРАМИ ПРОЦЕССА СУШКИ ЗЕЛЕНОГО ЧАЯ*

Фам К.Б., аспирант

Тверской государственной технической университет

e-mail: phambang79520897405@gmail.com

Богатиков В.Н., д.т.н., профессор

Тверской государственной технической университет

1. ВВЕДЕНИЕ

Чай является наиболее часто потребляемым безводным напитком в мире. Он охватывает большинство регионов и в последнее время фиксирует высокие темпы роста потребления [1]. Почти две трети чая в мире относятся к категории черного чая, а остальная часть в основном состоит из зеленого чая и другие чаи, такие как улун, жасмин и пуэр, в балансе. Для того, чтобы донести до потребителя качественный чайный продукт, свежие листья зеленого чая должны пройти множество сложных стадий обработки, в которой одним из важных этапов является сушка. Он не только продлевает срок хранения, предотвращая рост микроорганизмов, вредных для здоровья человека и качества сушеных чайных продуктов, но и придает чаю форму, цвет и вкус, характерные для зеленого чая.

Этот процесс характеризуется сложностью поддержания технологических параметров и малых рабочих диапазонов ведения процесса, высоким энергопотреблением и рядом неопределенностей и из-за повышенных энергозатрат требуется постоянное совершенствование для снижения энергозатрат (уменьшение эксплуатационных расходов) при сохранении или улучшении параметров качества количества конечного продукта [2–4]. Этого можно достичь за счет автоматизации и эффективного управления такими процессами.

* Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (проект № 20-07-00914).

В данной работе предложена автоматизированная система управления (АСУ) на основе нечеткого-ПИД контроллера для управления процессом сушки зеленого чая

2. СТРАТЕГИЯ УПРАВЛЕНИЯ ПРОЦЕССОМ СУШКИ ЗЕЛЕНОГО ЧАЯ

Основные задачи управления процессом сушки включают поддержание желаемой температуры продукта при наличии нарушений в процессе; регулирование скорости конвейерной ленты соответствующим образом, чтобы обеспечить правильное содержание влаги в выходящем чае и избежать чрезмерной сушки (что увеличивает затраты на энергию и может привести к термическому повреждению термочувствительных твердых веществ) и стабилизация общего процесса [5].

Основными критериями оптимизации процесса сушки зеленого чая являются качество конечного продукта (конечная влажность зеленого чая), удельные энергозатраты (на единицу массы испаренной влаги) и продолжительность сушки зеленого чая. В процессе конвекционной сушки эти критерии сильно зависят от температуры сушильной камеры и скорости сушильного конвейера. Таким образом, проблема управления процессом сушки зеленого чая сводится к управлению нагревателем, так что слой зеленого чая быстро нагревается до заданной температуры и затем остается постоянной в течение всего процесса и регулирование скорости конвейерных роликов для соответствия влажности продукта на выходе.

3. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ

Конвейерно-ленточная сушилка для натуральных продуктов (овощей, фруктов, лечебных трав и др.), именно в данной работе для зеленого чая. Схематический вид данной сушилки представлен на рис. 1 [6, 7]. Он состоит из ленточного конвейера, который медленно перемещает мелкие частицы влажного слоя зеленого чая через поток сушильного агента. Мелкие частицы влажного материала имеют примерную среднюю толщину $h = 48$ мм. В качестве сушильного агента используется предварительно подогретый влажный воздух с точно заданными характеристиками. Сушильный агент поступает по ленточному транспортеру и через зазоры между ломтиками влажного материала снизу вверх, перпендикулярно направлению движения влажного материала.

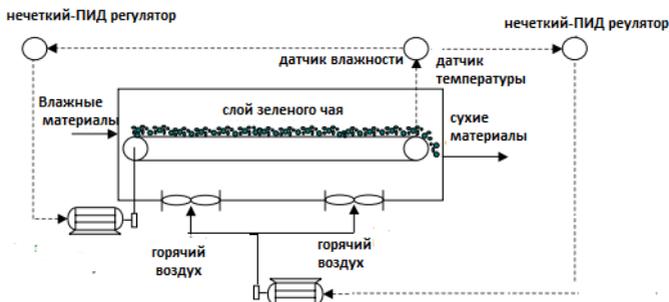


Рис. 1. Схема оцениваемой конвейерной сушилки

Установка состоит из конвейерной ленточной сушилки с поверхностью ленты 1 м x 5,4 м и набора датчиков, т.е. датчиков температуры, влажности. Для нагрева сушильного воздуха и скорости ленты используются регулируемые электронагреватель и конвейерных роликов (24 кВт). контролируется нечеткие-ПИД регуляторы, которые поддерживает температуру с обратной связью от термопары К-типа внутри слоя зеленого чая в камере и обеспечивает соответствие влажности продукта на выходе.

Контролируется нечеткие-ПИД регуляторами, которые поддерживает температуру с обратной связью от термопары К-типа внутри слоя зеленого чая в камере и корректирует скорость ленты с обратной связью от гигрометр в камере сушилки. Термопары измеряют температуру слоя через равные промежутки времени и передают среднее измеренное значение в модуль контроллера через аналогово-цифровую плату. Блоки с удаленной адресацией (PAX2C Dual Line 1/8 DIN Intelligent Meter Series, Red Lion Controls Inc.) используются для получения значений уставок от управляющего компьютера.

Уравнение переноса энергии [8]:

$$c\rho \frac{\partial T(x,\tau)}{\partial \tau} = \frac{\partial}{\partial x} \left(\lambda \frac{\partial T(x,\tau)}{\partial x} \right). \quad (1)$$

Начальные и граничные условия:

$$T(x, 0) = T_0 \quad (2)$$

$$\frac{\partial T(0,\tau)}{\partial x} = 0 \quad (3)$$

$$\lambda \frac{\partial T(r,\tau)}{\partial x} = k (T_c - T(r,\tau)), \quad (4)$$

где $T(x,\tau)$ – температура слоя зеленого чая; r – толщина слоя; λ – коэффициент теплопроводности; T_c – температура сушильного агента, k – коэффициенты теплоотдачи.

Уравнение переноса массы:

$$\frac{\partial M}{\partial \tau} = -u_c \frac{\partial M}{\partial x} - \frac{Ka(M-M_e)}{(1-\varepsilon)p_s} \quad (5)$$

u_c – скорость конвейерной ленты, x – осевое положение вдоль конвейерной ленты-сушилки, k – коэффициент массообмена, a – суммарная площадь поверхности массо- и теплоотдачи к объему слоя, M_e – равновесная влажность мате листьев, ε – пористость слоя, p_s – плотность сухих листьев зеленого чая.

Для получения передаточной функции процесса линейная модель системы получается путем линеаризации уравнений (1)–(5) с использованием разложения в ряд Тейлора и нахождения преобразования Лапласа полученного линейного уравнения по методике из [10]. Полученное уравнение в области s затем упрощается с использованием метода матрицы передачи, как описано в [11], и тепловые параметры процесса (как показано в табл. 1) подставляются в окончательное уравнение для получения упрощенной передаточной функции процесса:

$$G_{p1} = \frac{400}{(450s + 1)(60s + 1)}$$

$$G_{p2} = \frac{-60}{(5s + 1)(10s + 1)}$$

где G_{p1} – передаточной функции для температуры; G_{p2} – передаточной функции для влажности

Таблица 1. Тепловые параметры процесса сушки зеленого чая

Параметр	Обозначение	Значение
удельная теплоемкость	$c(\text{J/ kg/K})$	1780
плотность зеленого чая	$\rho(\text{kg/m}^3)$	750
теплопроводность	$a(\text{W/m/K})$	0.17
пористость	$\varepsilon(\text{m}^2/\text{g})$	1241
Температура окружающей среды	$T_0(\text{K})$	298

4. РЕГУЛЯТОР

В блоке регулятора используется алгоритм управления нечеткие-ПИД регуляторами с отрицательной обратной связью (рис. 2). Регуляторы температуры и скорости слоя зеленого чая в камере сушилки автоматически определяются параметры по самокорректирующемуся

нечеткому закону, с достаточно большим диапазоном точной подстройки вокруг значения, полученного экспериментальным методом Циглера-Николса.

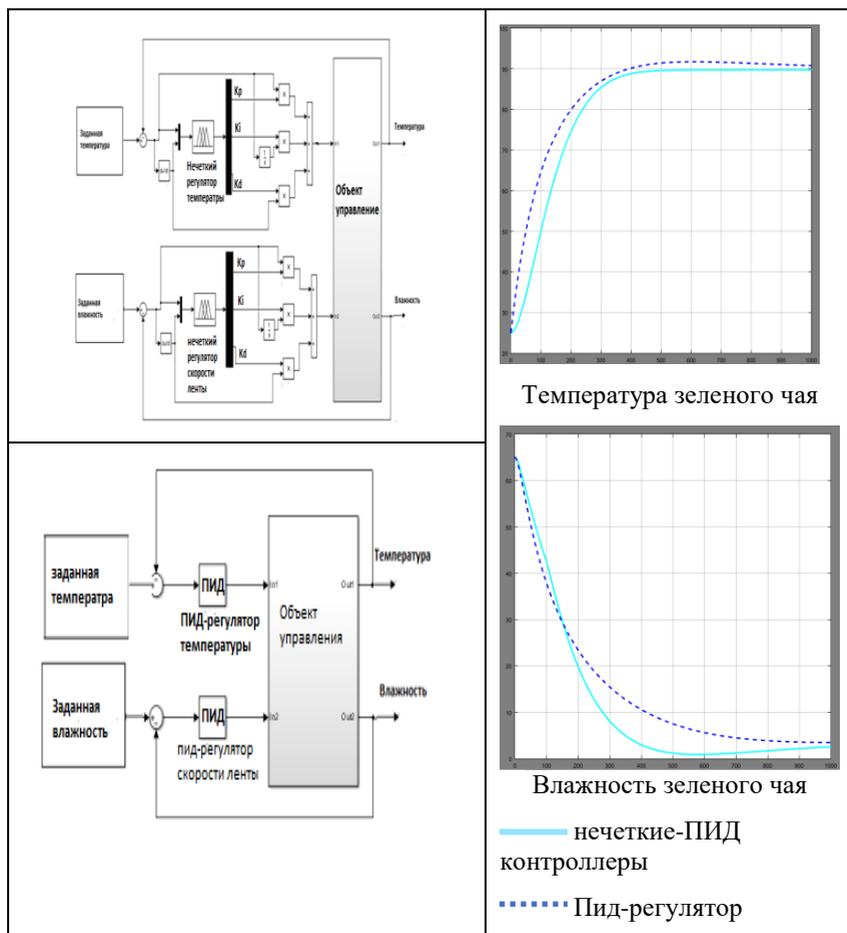


Рис. 2. Блок-схема регулятора обратной связи

В данном случае ПИД-регулятор играет роль основного регулятора, формирующего управляющие сигналы для объекта, а нечеткий регулятор [12] играет роль наблюдателя, выдающего подходящие параметры (пропорциональный – K_P , интегральный – K_I и производный – K_D) для ПИД-регулятора в режиме реального времени.

4.1. РЕГУЛЯТОР ПИД

Управляющее воздействие $U(\tau)$ для ПИД-алгоритма согласно [9] имеет вид:

$$U(\tau) = K_p \varepsilon + K_d \frac{\partial \varepsilon}{\partial \tau} + K_i \int_0^{\tau} \varepsilon d(\tau),$$

где K_p , K_i и K_d – пропорциональный, интегральный и производный коэффициенты; ε – ошибка = (заданная) – (измеряемая величина).

Передаточная функция для ПИД-регулятора может быть получена из преобразования Лапласа уравнения (9), чтобы получить [9]:

$$\frac{U(s)}{\varepsilon(s)} = K_p + K_d s + \frac{K_i}{s}.$$

4.2. НЕЧЁТКИЙ РЕГУЛЯТОР

4.2.1. Нечёткий регулятор для температуры

1.1. Определите переменные.

- Входная переменная:

Отклонение: $ET = T_{out} - T_{set}$

Где T_{out} – измеренная температура; T_{set} – заданная температура

Скорость изменения сигнала:

$DET = \frac{ET(i+1) - ET(i)}{T}$, где T — период выборки, i — время выборки

- Выходные переменные:

K_p – пропорциональный коэффициент

K_i – интегральный коэффициент

K_d – производный коэффициент

1.2. Лингвистические переменные

- Входная переменная:

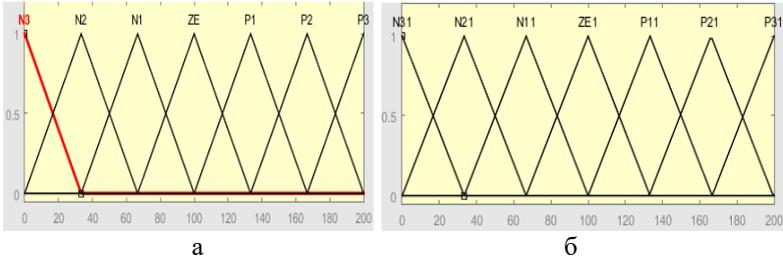
Со следующими соответствующими символами:

N3: очень мало. N2: умеренно мало. N1: мало.

ZE: средний. P1: много. P2: умеренно много. P3: очень много

→ Ошибка: $ET = \{N3, N2, N1, ZE, P1, P2, P3\}$

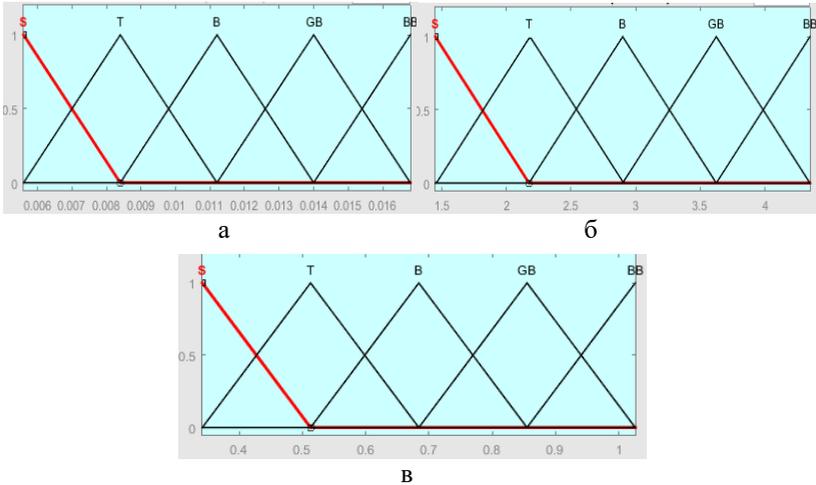
→ $DET = \{N3, N2, N1, ZE, P1, P2, P3\}$



**Рис. 3. Функция принадлежности входных переменных:
а – ET и б – DET**

- Выходная переменная:
Со следующими соответствующими символами:
S: мало. T: средний. B: много. GB: довольно много. BB: очень много.

- $Kp = \{ S, T, B, GB, BB \}$
- $Ki = \{ S, T, B, GB, BB \}$
- $Kd = \{ S, T, B, GB, BB \}$



**Рис. 4. Функция принадлежности выходных переменных:
а - Kp; б – Ki; в – Kd**

1.3. База правил нечеткого регулятора

Таблица 2. База правил Кр

Кр		DET						
		NB	NM	NS	ZE	PS	PM	PB
ET	NB	BB	BB	BB	BB	BB	BB	BB
	NM	GB	GB	GB	GB	GB	GB	GB
	NS	B	B	B	B	B	B	B
	ZE	T	T	S	S	S	T	T
	PS	B	B	B	B	B	B	B
	PM	GB	GB	GB	GB	GB	GB	GB
	PB	BB	BB	BB	BB	BB	BB	BB

Таблица 3. База правил Кд

Кд		DET						
		NB	NM	NS	ZE	PS	PM	PB
ET	NB	BB	BB	BB	BB	BB	BB	BB
	NM	GB	GB	B	B	B	GB	GB
	NS	B	B	B	B	B	B	B
	ZE	T	T	S	S	S	T	T
	PS	B	B	B	B	B	B	B
	PM	GB	GB	B	B	B	GB	GB
	PB	BB	BB	BB	BB	BB	BB	BB

Таблица 4. База правил Ки

Ки		DET						
		NB	NM	NS	ZE	PS	PM	PB
ET	NB	S	S	S	S	S	S	S
	NM	T	T	B	B	B	T	T
	NS	B	B	GB	GB	GB	B	B
	ZE	BB	BB	BB	BB	BB	BB	BB
	PS	B	B	GB	GB	GB	B	B
	PM	T	T	B	B	B	T	T
	PB	S	S	S	S	S	S	S

4.2.2. Нечёткий регулятор для скорости конвейерной ленты

2.1. Определите переменные.

- Входная переменная:

Отклонение: $ET = M_{out} - M_{set}$

где M_{out} – измеренная влажность; M_{set} – заданная влажность

Скорость изменения сигнала:

$DET = \frac{ET^{(i+1)} - ET^{(i)}}{T}$, где T – период выборки, i – время выборки

- Выходные переменные:
 K_p – пропорциональный коэффициент
 K_i – интегральный коэффициент
 K_d – производный коэффициент

2.2. Лингвистические переменные

- Входная переменная:
 Со следующими соответствующими символами:
 S: мало. M: средний. B: много.
 → Ошибка: $ET = \{S, M, B\}$
 → $DET = \{S, M, B\}$

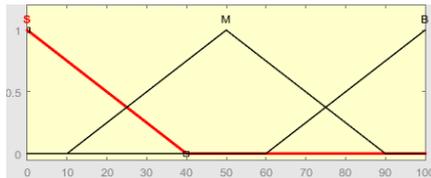


Рис. 5. Функция принадлежности входных переменных

- Выходная переменная:
 Со следующими соответствующими символами:
 S: мало. M: средний. B: много.
 → $K_p = \{S, M, B\}$
 → $K_i = \{S, M, B\}$
 → $K_d = \{S, M, B\}$

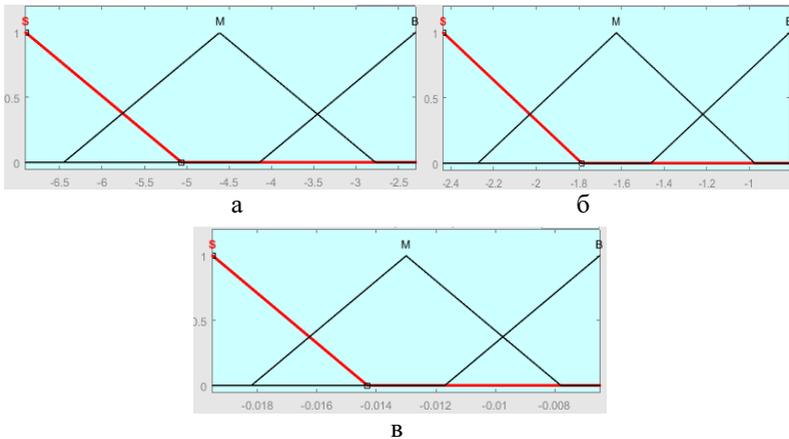


Рис. 6. Функция принадлежности выходных переменных:
а – K_p ; б – K_i ; в – K_d

2.3. База правил нечеткого регулятора

Таблица 5. База правил Кр

Кр		DET		
		S	M	B
ET	S	S	S	S
	M	M	M	B
	B	S	B	M

Таблица 6. База правил Ки

Ки		DET		
		S	M	B
ET	S	S	S	M
	M	S	S	S
	B	S	S	S

Таблица 7. База правил Кд

Кд		DET		
		S	M	B
ET	S	B	M	M
	M	M	M	M
	B	B	M	S

В результате внедрения нечеткого регулятора вносит коррективы в коэффициенты настроек ПИД-регулятора, изменение статического параметра питания не приводит к существенным отклонениям в работе и находится в пределах микрорегулировки и допустимого поддержания набора зеленых параметры сушки чая (рис. 7)

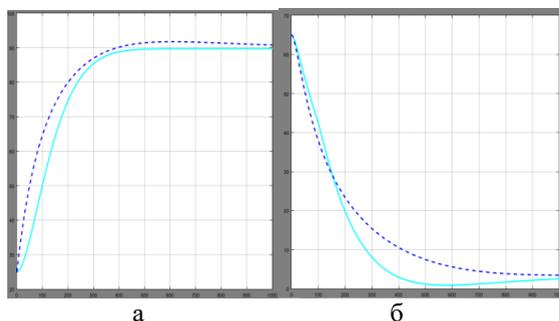


Рис. 7. Результаты моделирования различных регуляторов (ПИД и прогнозирующий нечеткий-ПИД контроллер) процесса сушки зеленого чая на основе Matlab

5. ЗАКЛЮЧЕНИЕ

В данной работе описаны проектирование, моделирование и реализация САУ на основе нечёткого-ПИД контроллера с отрицательной обратной связью для управления процессом сушки зеленого чая в конвейерной сушилке.

В целом результаты численного моделирования показывают, что нечеткий-ПИД контроллер работает стабильно и эффективно в условиях устранения входных шумов. Система обеспечивает относительно быстрое реагирование с точки зрения компенсированного устранения ошибок и устойчивого состояния при сушке.

Литература

1. International Tea Limited, 2008. Tea trade statistics and research, 1 Carlton House Terrace, London.
2. Mujumdar A.S. (Ed). Handbook of Industrial Drying. – 2nd ed. – New York: Marcel Dekker, 2006. – P. 20-31.
3. Mujumdar S. and Huang L.X. Global R&D Needs in Drying // Drying Technology: An International Journal. – 2010. – Vol. 25, No. 4. – P. 647-658.
4. Mujumdar A.S., R&D Needs, Challenges and Opportunities for Innovation in Drying Technology, 2010.
5. Areed F.F.G., El-Kasassy M.S. and Mahmoud K.A. Design of Neuro-Fuzzy Controller for a Rotary Dryer // International Journal of Computer Applications. – 2012. – Vol. 37, No. 5,
6. Salemović D.R., et al. A Mathematical Model and Simulation of the Drying Process of Thin Layers of Potatoes in a Conveyor-Belt Dryer // Thermal Science. – 19 (2015), 3. – P. 1107-1118.
7. Salemović D., Mathematical Modeling, Simulation and Identification of Drying Process of Natural Products in Striped Drying Chamber (in Serbian), Ph. D. thesis, University of Belgrade, Belgrade, Serbia, 1999
8. Дульнев Геннадий Николаевич. Теория тепло- и массообмена. – СПб: НИИУИТМО, 2012. – 195 с.
9. Mohd S.S., Hishamuddin J. and Intan Z.M.D. Implementation of PID controller tuning using differential evolution and genetic algorithms // Intenation journal of innovative computing, information and control. – November 2012. – Vol. 8, No. 11. – P. 7761-7769.
10. Vajta M. A New Model Reduction Technique for a Class of Parabolic Partial Differential Equations // in IEEE Int. Conference on Systems and Engineering, Dayton, Ohio, USA, 1991.
11. Vajta M. "yquist stability test for a parabolic partial differential equation // in IASTED Conf. on Modeling, Identification and Control, Innsbruck, Austria, 2000.
12. Кудинов Ю.И. 'Построение и настройка нечеткого адаптивного пид-регулятора // Интеллектуальные системы, Информатика и системы управления, – 2016. – № 3 (49). Работа выполнена при поддержке РФФИ, проект № 16-08-01279..

НЕЧЕТКИЙ ЛОГИКО-ЛИНГВИСТИЧЕСКИЙ АЛГОРИТМ ОБНАРУЖЕНИЯ ИНЦИДЕНТОВ В КИБЕРФИЗИЧЕСКИХ СИСТЕМАХ

Чикалов Н.В.

НИУ ИТМО

e-mail: nik.chikalow2011@yandex.ru

Березин М.Е.

*Санкт-Петербургский государственный университет
аэрокосмического приборостроения*

e-mail: adfpbm@gmail.com

Гатчин Ю.А., д.т.н.

НИУ ИТМО

e-mail: gatchin1952@mail.ru

Поляков В.И., к.т.н.

НИУ ИТМО

e-mail: v_i_polyakov@mail.ru

1. ВВЕДЕНИЕ

Факторами повышения вероятности возникновения инцидента являются: высокая степень износа основных производственных фондов, недостаточный уровень производственной и технологической дисциплины, недостаточный уровень подготовки специалистов, долгий период модернизации оборудования, неактуальное программное обеспечение, которое может содержать критические уязвимости, программные и аппаратные закладки.

Основной проблемой выявления инцидентов является нечеткость границы перехода между состоянием нормального функционирования и произошедшего инцидента. Поэтому возникают трудности в применении традиционных методов обнаружения инцидентов.

Нетривиальность задачи управления КФС обусловлена следующими факторами:

- большой объем “сырых” данных, что снижает производительность доступа к ним, а также доступа к системам и сетям связи предприятия;
- сложность администрирования информационных процессов;

- наличие большого числа промышленных технологий, протоколов, необходимых для информационного взаимодействия;
- разнородность фирм производителей оборудования;
- необходимость формирования контролируемой зоны, с целью недопущения физического деструктивного воздействия третьими лицами;
- территориально-распределенная структура обслуживаемых систем и оборудования;
- большое число элементов сети;
- атаки присущие только КФС;
- уязвимости нулевого дня.

2. АЛГОРИТМ ВЫЯВЛЕНИЯ И ПРОГНОЗИРОВАНИЯ ИНЦИДЕНТОВ НА ОСНОВЕ НЕЧЕТКОЙ ЛОГИКИ

В настоящей работе предлагается алгоритм для выявления и прогнозирования инцидентов на основе нечеткой логики. Во время функционирования КФС на промышленном предприятии анализируется состояние сенсоров, микроконтроллеров и управляющих механизмов на предмет возникновения инцидента. Блок-схема изображена на рис. 1.

Если анализ показал, что функционирование КФС отклоняется от нормальных значений, считается, что, вероятно, произошел инцидент. Следует отметить, что типовые ситуации строятся заранее перед началом работы системы управления, поэтому гарантировать высокую степень достоверности определения входных ситуаций во многих случаях нельзя. Если инцидент имеет место ($n=2$), но входная ситуация плохо определена, есть два решения.

Первое: доопределить плохо определенные в ней признаки, при наличии времени и возможности. После доопределения ($n=3$) инцидент регистрируется, формируется отчет, определяется управляющее воздействие.

Второе: получить характеристики инцидента в соответствии с наиболее сходной ситуацией с выдачей управляющих воздействий. При этом необходимо принимать во внимание снижение степени достоверности определения входной ситуации. Далее инцидент регистрируется, формируется отчет, определяется управляющее воздействие. Плохо определенная входная ситуация нечетко не равна ни одной из типовых ситуаций всего множества ранее определенных сценариев. В начальной стадии решения задач управления сложными объектами на первый план выступает процесс абстрагирования, сужения перцептивной зоны, с концентрацией внимания лишь на узловых, хорошо определенных признаках.

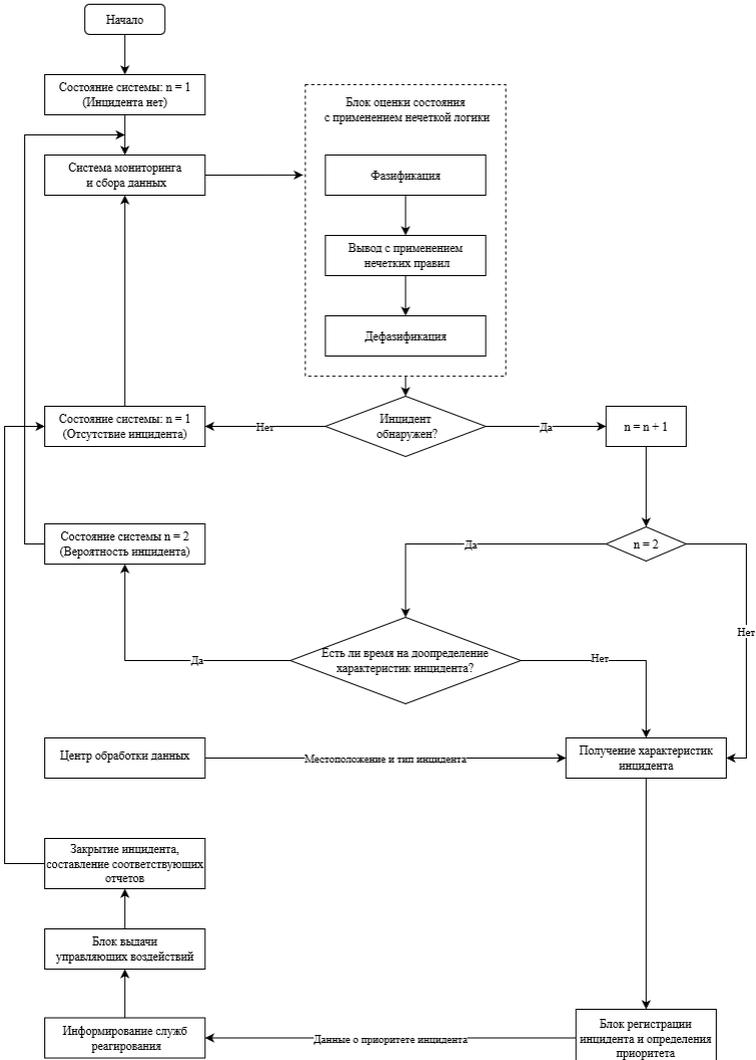


Рис. 1. Обнаружение инцидентов и определение их приоритетов с применением алгоритма на основе нечеткой логики

Ограниченный набор нечетких ситуаций может описывать практически бесконечное число состояний объекта управления. Состояния объекта управления оцениваются блоком оценки состояния через некоторые дискретные промежутки времени. Состояние объекта

представляется в виде нечеткой ситуации. Полученная входная нечеткая ситуация сравнивается с правилами. Определяется типовая нечеткая ситуация, наиболее близкая входной. Далее информация поступает в блок регистрации и определения приоритета. После предлагаются варианты управляющих воздействий.

3. ФОРМАЛИЗАЦИЯ ЛИНГВИСТИЧЕСКИХ ПЕРЕМЕННЫХ

Введем лингвистические переменные с соответствующими термножествами в следующем виде:

$$\langle x, T, X, G, M \rangle,$$

где x – лингвистическая переменная; T – множество наименований лингвистической переменной (так называемое термножество), областью определения каждой из которых является множество X ; G – синтаксическая процедура, которая позволяет оперировать элементами термножества T , в том числе генерировать новые термы; M – семантическая процедура, которая позволяет превратить каждое новое значение лингвистической переменной, образуемой процедурой G , в нечеткую переменную, т.е. сформировать соответствующее нечеткое множество.

В качестве входных переменных используем:

1. $\langle x_1, T_1, X_1, G_1, M_1 \rangle$, где x_1 – «время реакции оператора дежурной смены промышленного предприятия»; T_1 – {«низкое», «среднее», «высокое»}; X_1 – $[0, 1000]$ (мс); G_1 – синтаксическое правило образования новых термов (в этой процедуре применяем связки «и», «или» и следующие модификаторы «очень», «не», «немного»); M_1 – процедура задания нечетких подмножеств (A_{11} = «низкое», A_{12} = «среднее», A_{13} = «высокое»).

2. $\langle x_2, T_2, X_2, G_2, M_2 \rangle$, где x_2 – «пропускная способность инфокоммуникационной среды КФС при воздействии деструктивного воздействия»; T_2 – {«низкая», «средняя», «высокая»}; X_2 – $[0, 100]$ (Мбит/с); G_2 – синтаксическое правило образования новых термов (в этой процедуре применяем связки «и», «или» и следующие модификаторы «очень», «не», «немного»). M_2 – процедура задания нечетких подмножеств (A_{21} = «низкая», A_{22} = «средняя», A_{23} = «высокая»).

3. $\langle x_3, T_3, X_3, G_3, M_3 \rangle$, где x_3 – «число узлов КФС (потенциальных точек вхождения злоумышленника)»; T_3 – {«малый», «средний», «большой»}; X_3 – $[1, 100]$ (узлов); G_3 – синтаксическое правило образования новых термов (в этой процедуре применяем связки «и», «или» и следующие модификаторы «очень», «не», «немного»). M_3 – процедура задания нечетких подмножеств (A_{31} = «малый», A_{32} = «средний», A_{33} = «большой»).

4. $\langle x_4, T_4, X_4, G_4, M_4 \rangle$, где x_4 – «доступность КФС промышленного предприятия» доступность описывает возможность того, что КФС промышленного предприятия будет доступна для использования, когда это будет необходимо; T_4 – {«неудовлетворительная», «удовлетворительная», «хорошая», «отличная»}; X_4 – [0, 100] (процентов); G_4 – синтаксическое правило образования новых термов; M_4 – семантическая процедура, которая позволяет задать вид функции принадлежности нечетких подмножеств (A_{41} = «неудовлетворительная», A_{42} = «удовлетворительная», A_{43} = «хорошая», A_{44} = «отличная»).

5. $\langle x_5, T_5, X_5, G_5, M_5 \rangle$, где x_5 – «возможность возникновения неисправности КФС промышленного предприятия»; T_5 – {«низкая», «средняя», «высокая»}; X_5 – [1, 90] (дней); G_5 – синтаксическое правило образования новых термов; M_5 – семантическая процедура, которая позволяет задать вид функции принадлежности нечетких подмножеств (A_{51} = «низкая», A_{52} = «средняя», A_{53} = «высокая»)

6. $\langle x_6, T_6, X_6, G_6, M_6 \rangle$, где x_6 – «возможность возникновения отказа на сенсоре»; T_6 – {«4 сенсора, находящихся в горячем резерве», «3 сенсора, находящихся в горячем резерве», «2 сенсора, находящихся в горячем резерве», «1 сенсор, находящийся в горячем резерве»}; X_6 – [1, 4] (номер); G_6 – синтаксическое правило образования новых термов; M_6 – семантическая процедура, которая позволяет задать вид функции принадлежности нечетких подмножеств (A_{61} = «4 сенсора, находящихся в горячем резерве», A_{62} = «3 сенсора, находящихся в горячем резерве», A_{63} = «2 сенсора, находящихся в горячем резерве», A_{64} = «1 сенсор, находящийся в горячем резерве»).

7. $\langle x_7, T_7, X_7, G_7, M_7 \rangle$, где x_7 – «среда передачи информации между элементами КФС»; T_7 – {«кабель медный», «оптоволокну», «беспроводная среда»}; X_7 – [1, 3] (номер); G_7 – синтаксическое правило образования новых термов; M_7 – семантическая процедура, которая позволяет задать вид функции принадлежности нечетких подмножеств (A_{71} {«кабель медный», A_{72} = «оптоволокну», A_{73} = «беспроводная среда»}) В качестве выходной переменной используем:

$\langle y, T_y, X_y, G_y, M_y \rangle$, где y – «возможность возникновения инцидента»; T_y – {«низкая», «средняя», «высокая»}; X_y – [0, 100] (процентов); G_y – синтаксическое правило образования новых термов; M_y – семантическая процедура, которая позволяет задать вид функции принадлежности нечетких подмножеств (B_1 = «низкая», B_2 = «средняя», B_3 = «высокая»).

На рис. 2 отображены формализации лингвистических переменных и обучение выборки на основе нечетких правил возникновения инцидентов.

4. ПРИМЕР ФОРМИРОВАНИЯ БАЗЫ НЕЧЕТКИХ ПРАВИЛ

Составление нечетких правил базируется на следующих типах данных: априорные и эмпирические. Первый тип данных поступает от экспертов и основывается на их опыте. Второй тип данных получаем в результате проведенных измерений. Считаем, что данные от экспертов заранее обладают достаточной полнотой и непротиворечивостью. Следует отметить, что внесение в базу всех возможных правил может привести к существенной потере смысла логического вывода. Для верификации правил нечеткого вывода целесообразно использовать ситуации с известным результатом.

Пример обучающей выборки представлен в формуле (1):

$$(x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, x_4^{(k)}, x_5^{(k)}, x_6^{(k)}, x_7^{(k)}, y^{(k)}), k = 1, \dots, K, \quad (1)$$

где $x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, x_4^{(k)}, x_5^{(k)}, x_6^{(k)}, x_7^{(k)}$ – значения входных переменных $x_1, x_2, x_3, x_4, x_5, x_6, x_7$ в k -м примере; $y^{(k)}$ – значение выходной переменной y в k -м примере; K – общее количество примеров в обучающей выборке. Каждый образец обучающей выборки ставится в соответствие каждому продукционному правилу.

Определим для каждого образца функции принадлежности заданных значений нечетких лингвистических переменных:

- для $y^{(1)}$: $\mu_{B1}(y^{(1)}) = 0,9; \mu_{B2}(y^{(1)}) = 0,05; \mu_{B3}(y^{(1)}) = 0;$
- для $x_7^{(1)}$: $\mu_{A11}(x_7^{(1)}) = 1; \mu_{A72}(x_7^{(1)}) = 0; \mu_{A73}(x_7^{(1)}) = 0;$
- для $x_6^{(1)}$: $\mu_{A61}(x_6^{(1)}) = 0; \mu_{A62}(x_6^{(1)}) = 1; \mu_{A63}(x_6^{(1)}) = 0; \mu_{A64}(x_6^{(1)}) = 0;$
- для $x_5^{(1)}$: $\mu_{A51}(x_5^{(1)}) = 0; \mu_{A52}(x_5^{(1)}) = 0,3; \mu_{A53}(x_5^{(1)}) = 1;$
- для $x_4^{(1)}$: $\mu_{A41}(x_4^{(1)}) = 0; \mu_{A42}(x_4^{(1)}) = 0; \mu_{A43}(x_4^{(1)}) = 0,2;$
 $\mu_{A44}(x_4^{(1)}) = 0,8;$
- для $x_3^{(1)}$: $\mu_{A31}(x_3^{(1)}) = 0,7; \mu_{A32}(x_3^{(1)}) = 0; \mu_{A33}(x_3^{(1)}) = 0;$
- для $x_2^{(1)}$: $\mu_{A21}(x_2^{(1)}) = 0; \mu_{A22}(x_2^{(1)}) = 0; \mu_{A23}(x_2^{(1)}) = 1;$
- для $x_1^{(1)}$: $\mu_{A11}(x_1^{(1)}) = 0,5; \mu_{A12}(x_1^{(1)}) = 0,1; \mu_{A13}(x_1^{(1)}) = 0.$

Далее необходимо: во-первых - для каждого образца функции определить конкретные значения нечетких переменных; во-вторых – определить функции принадлежности, которые принимают max значение.

Ниже приведена часть правил нечеткого продукционного алгоритма:

Π_1 : ЕСЛИ «время реакции на инцидент оператора дежурной смены промышленного предприятия» есть «низкое» И «пропускная способность инфокоммуникационной среды КФС при воздействии деструктивного

воздействия» есть «высокая» И «число узлов КФС» есть «мало» И «доступность КФС промышленного предприятия» доступность описывает возможность того, что КФС промышленного предприятия будет доступна для использования, когда это будет необходимо» есть «отличная» И «возможность возникновения неисправности КФС промышленного предприятия» есть «низкая» И «возможность возникновения отказа на сенсоре» есть «4 сенсора, находящихся в горячем резерве» И «среда передачи информации между элементами КФС» есть «кабель медный», ТО «возможность возникновения инцидента» есть «низкая».

(Π_1 : ЕСЛИ x_1 есть A_{11} И x_2 есть A_{23} И x_3 есть A_{31} И x_4 есть A_{44} И x_5 есть A_{53} И x_6 есть A_{62} И x_7 есть A_{71} , ТО y есть B_1).

Π_2 : ЕСЛИ «время реакции на инцидент оператора дежурной смены промышленного предприятия» есть «среднее» И «пропускная способность инфокоммуникационной среды КФС при воздействии деструктивного воздействия» есть «средняя» И «число узлов КФС» есть «среднее» И «доступность КФС промышленного предприятия» есть «хорошая» И «возможность возникновения неисправности КФС промышленного предприятия» есть «средняя» И «возможность возникновения отказа на сенсоре» есть «2 сенсора, находящихся в горячем резерве» И «среда передачи информации между элементами КФС» есть «оптоволокно», ТО «возможность возникновения инцидента» есть «средняя» (рис. 2).

(Π_2 : ЕСЛИ x_1 есть A_{12} И x_2 есть A_{22} И x_3 есть A_{32} И x_4 есть A_{43} И x_5 есть A_{52} И x_6 есть A_{63} И x_7 есть A_{72} , ТО y есть B_2).

Π_{20} : ЕСЛИ «время реакции на инцидент оператора дежурной смены промышленного предприятия» есть «высокое» И «пропускная способность инфокоммуникационной среды КФС при воздействии деструктивного воздействия» есть «низкая» И «число узлов КФС» есть «большое» И «доступность КФС промышленного предприятия» есть «неудовлетворительная» И «возможность возникновения неисправности КФС промышленного предприятия» есть «низкая» И «возможность возникновения отказа на сенсоре» есть «4 сенсора, находящихся в горячем резерве» И «среда передачи информации между элементами КФС» есть «беспроводная», ТО «возможность возникновения инцидента» есть «высокая»

(Π_{20} : ЕСЛИ x_1 есть A_{13} И x_2 есть A_{21} И x_3 есть A_{33} И x_4 есть A_{41} И x_5 есть A_{51} И x_6 есть A_{61} И x_7 есть A_{73} , ТО y есть B_3).

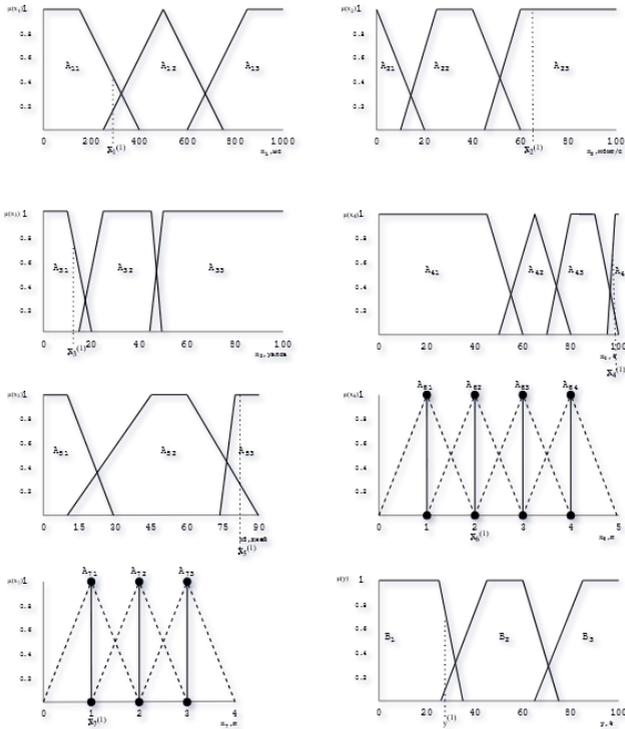


Рис. 2. Формализация лингвистических переменных и обучение выборки на основе нечетких правил возникновения инцидента (для Π_2)

Сформулированные правила могут служить первоначальной базой нечетких правил. Для исключения избыточности в первоначальной базе необходимо выявить логические противоречия. Оптимизация правил может быть достигнута за счет использования практического опыта экспертов, а также за счет адаптации набора обучающей выборки. Получить какие-либо нечеткие заключения о текущем состоянии нечеткой системы, в которой может возникнуть инцидент, на основе определенных нечетких условий и правил можно посредством реализации алгоритма нечеткого вывода.

Нечеткая логико-лингвистическая модель и механизм нечеткого вывода возникновения инцидента в КФС на промышленном предприятии могут быть описаны выражением 2:

$$\mu_{B_i}(y) = \bigcap_{i=1}^{20} \sup_{x \in X} \{T[\alpha_i, \mu_{\bar{A}_i \rightarrow \bar{B}_i}(x, y)]\} = \bigcap_{i=1}^{20} \sup_{x_1 \dots x_7 \in X} \{T(\mu_{\bar{A}_{i1}}(x_1), T(\mu_{\bar{A}_{i2}}(x_2), T(\mu_{\bar{A}_{i3}}(x_3), T(\mu_{\bar{A}_{i4}}(x_4), T(\mu_{\bar{A}_{i5}}(x_5), T(\mu_{\bar{A}_{i6}}(x_6), T(\mu_{\bar{A}_{i7}}(x_7), \mu_{\bar{B}_i}(y))))))\} \}. \quad (2)$$

Этап 1. Основываясь на переменных нечеткой логики, а именно на лингвистических переменных возникновения инцидента, $\mu_{\bar{A}_{ij}}(x_j')$, ($i = 1..20, j = 1..7$), определим достоверность всех правил, т.е. сформируем степень истинности.

Этап 2. По каждому из правил α_i уровневого множества осуществляем агрегирование степеней истинности предпосылок (выражения (3)-(5)):

$$\alpha_i = T\{\mu_{\bar{A}_{i1}}(x_1'), T\{\mu_{\bar{A}_{i2}}(x_2'), T\{\mu_{\bar{A}_{i3}}(x_3'), T\{\mu_{\bar{A}_{i4}}(x_4'), T\{\mu_{\bar{A}_{i5}}(x_5'), T\{\mu_{\bar{A}_{i6}}(x_6'), T\{\mu_{\bar{A}_{i7}}(x_7')\}\}\}\}\}\}, \quad (3)$$

$$\alpha_2 = T\{\mu_{\bar{A}_{21}}(x_1'), T\{\mu_{\bar{A}_{22}}(x_2'), T\{\mu_{\bar{A}_{23}}(x_3'), T\{\mu_{\bar{A}_{24}}(x_4'), T\{\mu_{\bar{A}_{25}}(x_5'), T\{\mu_{\bar{A}_{26}}(x_6'), T\{\mu_{\bar{A}_{27}}(x_7')\}\}\}\}\}\}. \quad (4)$$

$$\alpha_{20} = T\{\mu_{\bar{A}_{201}}(x_1'), T\{\mu_{\bar{A}_{202}}(x_2'), T\{\mu_{\bar{A}_{203}}(x_3'), T\{\mu_{\bar{A}_{204}}(x_4'), T\{\mu_{\bar{A}_{205}}(x_5'), T\{\mu_{\bar{A}_{206}}(x_6'), T\{\mu_{\bar{A}_{207}}(x_7')\}\}\}\}\}\}. \quad (5)$$

Этап 3. По каждому из правил α_i уровневого множества определяем степень истинности заключений $\mu_{\bar{B}_i}(y)$ (выражения (6)-(8)):

$$\mu_{\bar{B}_1}(y) = T\{\alpha_1, \mu_{\bar{B}_1}(y)\} \quad (6)$$

$$\mu_{\bar{B}_2}(y) = T\{\alpha_2, \mu_{\bar{B}_2}(y)\}, \quad (7)$$

$$\mu_{\bar{B}_{20}}(y) = T\{\alpha_{20}, \mu_{\bar{B}_{20}}(y)\}. \quad (8)$$

На рис. 3 представлен процесс определения степени заключений для двух лингвистических переменных x_1 и x_2 .

Этап 4. Для каждого конкретного правила α_i уровневого множества выполняется группировка заключений $\mu_{\bar{B}_{20}}(y)$, определяемых на 3 этапе. Получаем то, для возможности возникновения инцидента, т.е. в данном случае переменная y (выходная переменная), составляет нечеткое множество и соответствующая функция принадлежности $\mu_{B'}(y)$ (выражение (9)):

$$\mu_{B'}(y) = S\{\mu_{\bar{B}_1}(y), S\{\mu_{\bar{B}_2}(y), \dots, S\{\mu_{\bar{B}_{19}}(y), S\{\mu_{\bar{B}_{20}}(y)\}\}\}\}. \quad (9).$$

На рис. 4 представлена систематизация заключений по правилам. S-операция применяется для формирования единой группы усеченных нечетких множеств, содержащих информацию возникновения инцидента.

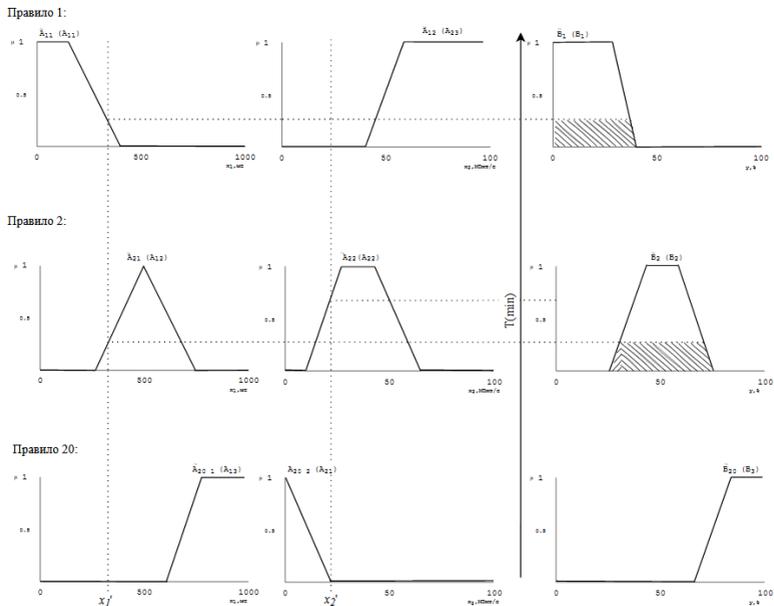


Рис. 3. Иллюстрация активизации заключений алгоритма нечеткого вывода

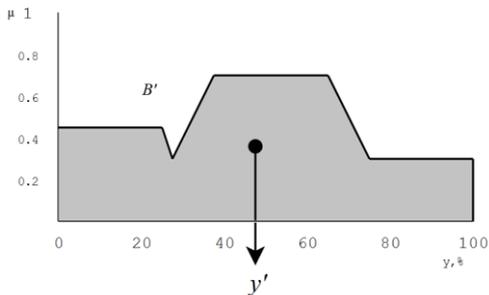


Рис. 4. Графическое представление систематизации заключений нечеткого вывода

Этап 5. Применение операции дефазификации к полученному нечеткому множеству для формирования четкости. Четкость y' определяется как центр тяжести для $\mu_{B'}(y)$

$$y' = \frac{\int_{Y_{min}}^{Y_{max}} y \mu_B(y) dy}{\int_{Y_{min}}^{Y_{max}} \mu_B(y) dy} . \quad (11)$$

где Y_{min} , Y_{max} – заданные границы интервала носителя нечеткого множества выходной переменной y – «возможность возникновения инцидента».

5. ЗАКЛЮЧЕНИЕ

Одной из наиболее значимых характеристик реагирования на инцидент является неопределенность в принятии решений. Следовательно, аппарат нечеткой логики является наиболее подходящим решением проблемы управления инцидентами. Алгоритм на основе нечеткой логики является эффективным методом при описании нечеткого определения условий возникновения инцидента и фактических суждений, используемых при обработке полученных данных о происшествии. Предлагаемый алгоритм является гибким благодаря возможности создать любое нечеткое правило под необходимые условия для определения происшествия, связанного с особенностями функционирования КФС промышленных предприятий.

Предложенный алгоритм позволяет уйти от вероятностного подхода к оценке возможности возникновения инцидента в КФС на промышленном предприятии, который обладает существенным недостатком – сложностью, а зачастую невозможностью определения инцидента в КФС на промышленном предприятии. Алгоритм базируется на теории возможностей, позволяя определить искомое значение возможности возникновения инцидента в КФС на промышленном предприятии, исходя из опыта экспертов, либо на основе проведенных экспериментов.

С помощью выбранного алгоритма нечеткого вывода для определенных значений характеристик системы управления КФС промышленных предприятий и операции дефазификации получено четкое значение возможности возникновения инцидента на объекте.

Использование нечеткой логики при оценке инцидентов и построение логических заключений в моделях управления сложными объектами, облегчает решение задач обеспечения общения с пользователем на профессионально-ориентированном языке, хранения, накопления и обработки качественной информации.

Литература

1. Anomaly detection techniques in cyber-physical systems // Dheorge Sebestyen, Anca Hangan.
2. Борисов В.В., Федулов А.С., Зернов М.М. Основы теории нечетких множеств: учеб. пособие для вузов. – М.: Горячая линия-Телеком, 2014. – 88 с.
3. Zimmermann H.J. Fuzzy set theory and its applications. – 2nd ed. – Boston/Dordrecht/London: Kluwer Academic Publishers, 1991. – 435 p.
4. Громов Ю.Ю., Иванова О.Г., Мосягина Н.Г., Соседов Г.А., Точка В.Н. Представление знаний в информационных системах. – Тамбов, М., СПб., Баку, Вена: Изд-во «Нобелистика», 2008. – 144 с.
5. Борисов А.Н., Алексеев А.В., Крумберг О.А. Модели принятия решений на основе лингвистической переменной. – Рига: Зинатне, 1982. – 256 с.,
6. Берштейн Л.С., Боженюк А.В. Нечеткие модели принятия решений: дедукция, индукция, аналогия: монография. – Таганрог: Изд-во ТРТУ, 2001. – 110 с.
7. Wang L.X., Mendel J.M. Generating fuzzy rules by learning from examples // IEEE Transaction on Systems, Man and Cybernetics. – 1992. – Vol. 22, No. 6. – P. 1414-1427
8. Борисов В.В., Федулов А.С. Нечеткие продукционные модели и сети. – Смоленск: Изд-во ВА ВПИО ВС РФ, 2005.

СЕКЦИЯ 2 | **МАШИННОЕ ОБУЧЕНИЕ,
НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ**

УДК 004.8

**ПРЕДИКТИВНЫЙ АНАЛИЗ ПОВЕРХНОСТНОЙ
ДЕФОРМАЦИИ МАТЕРИАЛОВ НА ОСНОВЕ
ДАННЫХ АКУСТИЧЕСКАЯ ЭМИССИИ***

Александров А.А.

МИИ ИМ ЮФУ

e-mail: alea@sfedu.ru

Чернов А.В., д.т.н., профессор

МИИ ИМ ЮФУ

e-mail: cherno@sfedu.ru

Поляниченко Д.С.

МИИ ИМ ЮФУ

e-mail: poliani@sfedu.ru

Карташов О.О., к.т.н.

МИИ ИМ ЮФУ

e-mail: okartashov@sfedu.ru

Бутакова М.А., д.т.н., профессор

МИИ ИМ ЮФУ

e-mail: mbutakova@sfedu.ru

1. ВВЕДЕНИЕ

В условиях эксплуатации деталей, технических устройств и объектов, все без исключения, в период применения теряют свои технические характеристики и могут быть разрушены. Изнашивание поверхностного слоя материала или деталей происходит на всей поверхности материала и в местах фактического контакта деталей, которые

* Работа выполнена при поддержке проекта Старт №71351.

подвергаются значительной упругой и пластической деформации. Вследствие этого, в материале происходит его разупрочнение и разрыхление. Упрочнение поверхностей деталей повышает сопротивление усталости и максимальному напряжению на поверхности детали, поэтому дефекты, возникающие внутри, не всегда могут проявляться на поверхности. При статических, циклических и ударных нагрузках прочность материала должна противостоять возможным разрушениям, в том числе и возникающих под действием нагрузок деформациям, которые нарушают работоспособность деталей, приводя к процессу возникновения и развития трещин в материале как поверхностных, так и внутренних [1].

В материалах металлов в начальный момент приложения внешней нагрузки на поверхность наблюдается упругая деформация, далее материал поверхности начинает пластически деформироваться. При этом после снятия приложенной нагрузки, в металле останется пластическая деформация сопровождающейся увеличением числа дефектов и изменением субструктуры.

Для выявления, подобных поверхностных, внутренних и опасных дефектов применяют метод акустической эмиссии (АЭ). Он является одним из глобальных методов не разрушающего контроля, позволяющий заблаговременно получать данные об ухудшении целостности материала, появлении дефектов и надежности изделия. Данный метод контроля позволяет находить очаги напряжений в местах нарушающие целостность материала. Напряжения возникают в месте дефекта, где в поврежденном участке объекта происходит генерации волн напряжений, в то время как основная бездефектная часть не вызывает генерации напряжений, что позволяет идентифицировать место разрушения конструкции.

В исследовании, с помощью метода акустической эмиссии рассматривается способ предиктивного анализа возникновения поверхностной деформации материала, происходящим под воздействием соударения инструмента с материалом и приводящих к возникновению, и развитию дефектов.

2. ОБНАРУЖЕНИЕ ДЕФЕКТОВ МАТЕРИАЛА АКУСТИКО-ЭМИССИОННЫМИ СПОСОБАМИ

Акустическая эмиссия [2] представляет собой явление, связанное с возникновением и распространением звуковых колебаний (упругих колебаний, звуковых волн) в твердой среде, в конструкционном материале, подвергающемся деформациям, вследствие механических разрушений, а также иных деформаций, возникающих вследствие горения, взрывов, испарения жидкостей и прочих процессов. Уникаль-

ной особенностью АЭ является устойчиво различаемая звуковая последовательность, имеющая характерные особенности для различных материалов и различных разновидностей деформаций в них. Этот факт и был положен в основу известного метода технической диагностики, который называется акустико-эмиссионный контроль (АЭК) [3]. В настоящее время он относится к распространенным пассивным видам неразрушающего контроля, позволяющего количественно оценивать большой спектр деформационных изменений материалов (трещины, разрывы, разломы, расслоения и т.п.) затрачивая при этом относительно небольшие измерительные ресурсы и время.

Пассивность контроля в данном случае непосредственно относится к ключевой особенности самой акустической эмиссии – источником сигнала служит сам материал, который находится под контролем. Стоит отметить, что практически все деформационные процессы в материале, включая также электрохимические и химические преобразования, пластические деформации из-за температуры, давления, трения и износа, сопровождаются возникновением и распространением АЭ. Это позволяет отслеживать динамику процессов преобразования материала, включая перемещение трещин, развитие расслоений, увеличение разломов, учитывая не только причины механического характера. Преобразователи АЭ могут использоваться для проверки объектов достаточно высокой сложности и в условиях применения разнородных материалов. При этом распространение АЭ сигналов может регистрироваться на удалении от мест возникновения дефектов на десятки метров. Вследствие таких особенностей АЭ контроль относится к интегральным методам контроля. Вместе с этим сигналы АЭ достаточно сложны для информативного анализа и подвержены влиянию сторонних звуковых волн и шумов, что несколько осложняет повсеместное использование методов АЭ контроля. Для повышения информативности АЭ измерений в зависимости от объекта и методики контроля могут применяться несколько АЭ датчиков, причем один из которых может искусственно генерировать эталонные АЭ колебания.

Физические процессы, приводящие к процессам АЭ можно описать следующим образом [4]. Твердое тело, материал, обладает кристаллической решеткой, каждый атом которой при недеформированном состоянии материала находится в равновесном состоянии. При возникновении разного рода деформаций, сопровождающихся нагружением кристаллической решетки материала начинает накапливаться потенциальная энергия, которая импульсно высвобождается при превышении пределов прочности кристаллической решетки, что сопровождается излучением акустических волн, имеющих достаточно широ-

кий диапазон. Вполне понятно, фиксация единичных импульсов акустических волн, возникающих в процессе разрыва локальных атомных связей единичного элемента кристаллической решетки твердого вещества, является достаточно проблематичной. Однако, при одновременных осцилляциях значительного числа элементов кристаллической решетки материала, генерируемые акустические волны накладываются и происходит процесс АЭ, поддающийся уверенной регистрации специализированными АЭ датчиками. Поля упругих волн источников АЭ, которые являются моделями развивающихся дефектов в материалах, до сих пор вызывают значительный интерес у исследователей [5].

3. МЕТОД ОПРЕДЕЛЕНИЯ ПОВЕРХНОСТНОЙ ДЕФОРМАЦИИ МАТЕРИАЛОВ

Предлагаемый в исследовании метод акустической эмиссии позволяет обнаруживать процесс достижения материалом предельного состояния и начала его разрушения. На рис. 1 представлен метод определения поверхностной деформации материалов с помощью нейронной сети.



Рис. 1. Определения поверхностной деформации материалов

Процесс нагрузки материала предлагается проводить с помощью кратковременного взаимодействия инструмента или соударением. Во время соударения между материалом и инструментом начинают действовать кратковременные ударные силы, сопровождающиеся возникновением волн напряжений, которые возникают и распространяются в обоих телах. Предлагаемый метод анализирует начальный момент приложения внешней нагрузки на поверхность и распространяющиеся напряжения в материале, которые затухают в результате упругой деформации. Следует отметить, распространение зависит от скорости, массы, формы и самого материала. Возникающее напряжения будут свидетельствовать о местах нарушения целостности материала, которые соответствуют поверхностным, внутренним и опасным дефектам [6].

Измерения с помощью датчиков акустической эмиссии позволяют регистрировать, впоследствии усиливать, ударные импульсы и напряжения в материале в широком диапазоне частот от 40 кГц до 800 кГц, прочие частоты подавляются. Обычно рабочие частоты виб-

раций механизмов находятся на значительно низких частотах, чем резонансная частота датчиков, что позволяет отфильтровывать рабочие частоты механизмов.

Сформированные за время измерения значения амплитуд импульсов сводятся воедино в виде таблицы, что облегчает их дальнейшее эффективное использование. Данные будут рассматриваться в виде многомерных временных рядов. Существует большое множество различных методов прогнозирования временных рядов, как с помощью статистических методов, так и глубокого машинного обучения. В исследование анализ и прогнозирование возникновения поверхностной деформации материала будет проводиться с помощью глубокого обучения, сетью с долгой краткосрочной памятью [7].

4. ПРОВЕДЕНИЕ ЭКСПЕРИМЕНТА

Пластическая деформация поверхностного слоя материала будет производиться с помощью инструмента в условиях циклического нагружения, которая соответствует большой степени наклепа. Вследствие данного процесса в материале будет проявляться разуплотнение и потеря несущей способности металла поверхностного слоя.

Для определения степени влияния циклического нагружения на степень изменения сигналов акустической эмиссии поверхностного слоя, были проведены исследования относительно равномерной деформации образцов металла с последующим их разрушением. В результате, для сбора данных в качестве деформирующегося материала использовалась заготовка из коррозионностойкой жаропрочной стали 08X18H10 изготовленной в виде трубы диаметром 25 мм и толщиной стенки 1,5 мм. Взаимодействующий инструмент изготовлен из закаленной инструментальной стали с заостренной частью под углом в 120°, закреплённый на подвижной платформе, общей массой 10 кг, на расстоянии 150 мм от деформирующегося материала.

Деформирующийся материал прочно закрепляется, не подвержен движениям и другим воздействиям. Процесс соударения материала и инструмента происходит с помощью продольного перемещения инструмента с силой большой степени наклепа, при этом удар получается центральным. Процесс соударения повторяется циклически, на протяжении 40 измерений. Образовавшиеся в процессе проведения эксперимента повреждение поверхности, представляет дефект в виде единичного углубления с пологими краями.

На рис. 2 представлена общая схема размещения компонентов и их взаимодействия. Для уверенной регистрации ударных импульсов и напряжения в материале датчик акустической эмиссии был распо-

жен в непосредственной близости от места соударения, на объект контроля, на расстояние равном 50 мм, через слой контактной жидкости с постоянным усилием прижима.

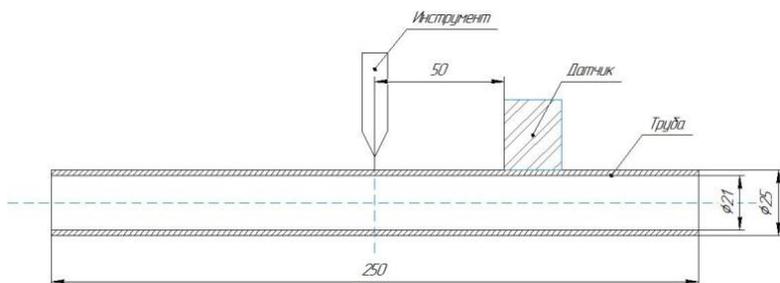


Рис. 2. Общая схема размещения компонентов

Регистрация и преобразование сигналов акустической эмиссии применялся датчик ZET 601 с полосой пропускания от 100 до 800 кГц и коэффициентом электроакустического преобразования более 45 дБ отн. 1 В/м/с, с последующим усилением сигнала. С датчика акустической эмиссии сигнал попадает на модуль усилителя состоящего из усилителя заряда и инструментального усилителя. Зарегистрированный датчиком и усиленный усилителем сигнал поступает на аналого-цифровой преобразователь, где преобразуется в цифровую форму сигнала с частотой дискретизации 2,4 МГц. Для получения данных с аналого-цифрового преобразователя, на компьютер, используется скрипт, написанный на языке программирования Python. Передача осуществляется по протоколу передачи данных TCP/IP. После передачи массива данных на компьютер, возвращается подтверждение об успешном получении данных, иначе передача повторяется, после этого данные сохраняются в файл. Исходный код доступен на <https://github.com/Kilmarnock13/surfaceDeformation.git>. Собранные данные временных рядов позволяют сохранить естественную хронологию воздействием соударения инструмента с материалом, что соответствует амплитуде сигналов акустической эмиссии в порядке её возникновения.

5. ПРИМЕНЕНИЕ СЕТИ С ДОЛГОЙ КРАТКОСРОЧНОЙ ПАМЯТЬЮ

Полученные в ходе эксперимента данные временных рядов процесса упругой деформации поверхности материала были предварительно обработаны для использования в модели долгой краткосрочной памяти. Для начала из исходных данных была удалены части, не не-

сущие информативности – это признак шага измерения и значения, соответствующие времени ожидания события. Абсцисса в эксперименте – это временные интервалы, которые всегда одинаковы и циклически на протяжении каждого конечного измерения, следовательно не несут информативной нагрузки для последующей модели. Так же и значения напряжения после последнего пересечения амплитуды сигнала и до полного затухания, которое можно обозначить как T_0 не обязательны для обучения построения прогнозных кривых акустической эмиссии. После проведения такой фильтрации разбиение данных по условным единицам представлены на рис. 3.

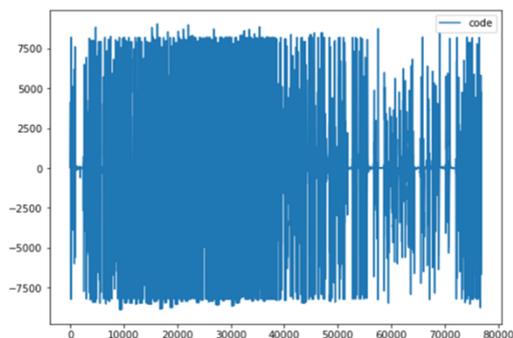


Рис. 3. Графическое представление данных

Далее выполняется стандартное масштабирование данных в виде нормализации. Из каждого измерения вычитается среднее значение, которое делится на стандартное отклонение. Таким образом каждое число помещается в диапазоне $[0,1]$. Для доступного количества измерений было выполнено разбиение на обучающую, для которой и выполнялся метод нормализации данных и валидационную выборки, где обучающая включала в себя 30000 значений амплитуды.

Для решения задачи прогнозирования временного ряда была составлена простая модель искусственной нейронной сети с долгой краткосрочной памятью, где входной слой – LSTM (Long short-term memory), включающий в себя 32 нейрона и способный обрабатывать последовательности, 2 слой – LSTM, включающий в себя 16 нейронов, использующий активационную функцию Relu, слой регуляризации с параметром исключения в 10 % и выходной плотно связанный слой на 24 значения. Оптимизатор – RMSprop, функция потерь – среднее значение абсолютной разницы между метками и прогнозами (Mean Absolute Error). Предварительно использовалось прогнозирование 24 значений временного ряда. Обучение производилось на протяжении

10 эпох с интервалом в 1000 значений. С данными настройками нейронная сеть обучилась 15 минут и 37 секунд. Графики обучающих и валидационных потерь представлены на рис. 4.

В результате точность прогнозирования составила 83.42%. Примеры представлены на рис. 5.

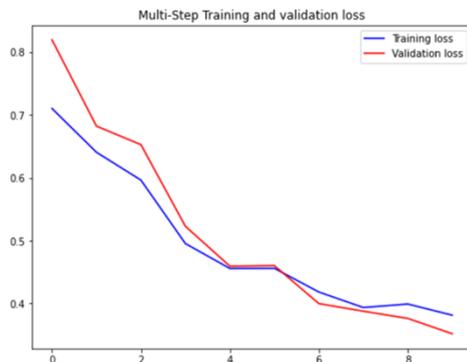


Рис. 4. Потери обучения и валидации

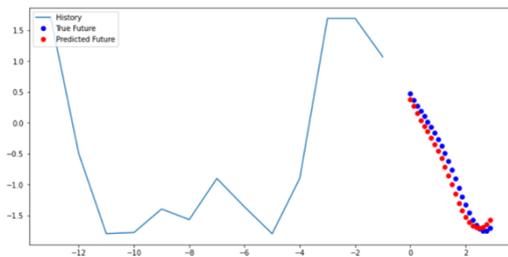


Рис. 5. Результат прогнозирования

6. ЗАКЛЮЧЕНИЕ

В докладе рассмотрено решение задачи предиктивного анализа поверхностной деформации материалов с помощью машинного обучения. С помощью метода акустической эмиссии получены данные позволяющие отслеживать динамику процессов преобразования материала возникающие при его поверхностной деформации. Для определения изменений силы сигналов АЭ был применен метод MARSE, представляющий относительную энергию излучения, распространяющуюся в материале и описывающей собой измеренную площадь под огибающей сигнала. Метод отличается от других своей чувствительностью по амплитуде и времени сигнала, но не использует пороговые

значения и подсчет количества чисел осцилляции. Данный метод был выбран по причине максимальной чувствительности и служит идентификатором ускоренного роста дефектов. Предложенный подход к прогнозированию данных временных рядов измерений акустической эмиссии на основе использования моделей рекуррентных нейронных сетей, позволил прогнозировать протекание процесса поверхностной пластической деформации с высокой точностью прогнозирования 83.42%. Благодаря прогнозу данных временных рядов и применяемому методу, определения энергии сигнала (MARSE), становится возможным предсказывать возрастание поверхностной деформации материала с высокой долей вероятности. Исходный код доступен на <https://github.com/Kilmarnock13/surfaceDeformation.git>.

Литература

1. Xu Y., Balint D.S., и Dini D. On the Origin of Plastic Deformation and Surface Evolution in Nano-Fretting: A Discrete Dislocation Plasticity Analysis // *Materials*. – Окт. 2021. – Т. 14. – Вып. 21. – С. 6511. – DOI: 10.3390/ma14216511.
2. Acoustic Emission. Sikorski, W. (Ed.). IntechOpen, 2012. – URL: <https://doi.org/10.5772/2070>.
3. Баранов В.М., Карасевич А.И. и др. Акустическая диагностика и контроль на предприятиях топливно-энергетического комплекса. – М.: Наука, 1998. – 304 с.
4. Оглезнева Л.А., Калининченко А.Н. Акустические методы контроля и диагностики. Ч. II. – Изд-во Томского политехнического университета, 2009. – 292 с.
5. Буденков Г.А., Недзвецкая О.В. Динамические задачи теории упругости в приложении к проблемам акустического контроля и диагностики. – М.: Физматлит, 2004. – 136 с.
6. Niinomi M. Fatigue failure of metallic biomaterials // в *Metals for Biomedical Devices*, Elsevier, 2010. – С. 122-156. – Doi: 10.1533/9781845699246.2.122.
7. Ensafi Y., Amin S.H., Zhang G., и Shah B. Time-series forecasting of seasonal items sales using machine learning – A comparative analysis // *International Journal of Information Management Data Insights*. – Апр. 2022. – Т. 2. – Вып. 1. – С. 100058. – doi: 10.1016/j.jjime.2022.100058.

ПОДХОД К АНАЛИЗУ ЛИЧНОСТНЫХ ХАРАКТЕРИСТИК ПОЛЬЗОВАТЕЛЕЙ СОЦИАЛЬНЫХ СЕТЕЙ С ПРИМЕНЕНИЕМ МАШИННОГО ОБУЧЕНИЯ*

Андреев И.А., аспирант

Ульяновский государственный технический университет

e-mail: ares-ilya@ya.ru

Мошкин В.С., к.т.н., доцент

Ульяновский государственный технический университет

e-mail: postforvadim@ya.ru

Ярушкина Н.Г., д.т.н., профессор

Ульяновский государственный технический университет

e-mail: jng@ulstu.ru

1. ВВЕДЕНИЕ

В современном мире социальные сети являются огромным источником информации о каждом человеке. Работодатель может узнать значительно больше информации из социальных сетей, нежели информация, поданная самим соискателем на конкретную должность в резюме. В настоящее время сбор и содержательный анализ собранной в социальных сетях информации проводится вручную специалистами кадровых служб, что требует больших затрат времени и ограничивает объём обрабатываемой информации.

В работах [1, 2] отмечается важность сбора информации о пользователях социальных сетей с целью предиктивного анализа и выявления психических расстройств и рассматривается влияние социальных сетей на разбиение пользователей на отдельные социальные группы в зависимости от их поведения в интернет-пространстве.

В работе [3] пользователи социальных сетей объединяются в аутентичную субкультуру, при этом личностные черты каждого пользователя определяются путем анализа открытых структурированных (анкета) и неструктурированных (сообщения, статусы) данных страницы.

* Работа выполнена при финансовой поддержке Минобрнауки России в рамках проекта №075-00233-20-05 от 03.11.2020 «Исследование интеллектуального предиктивного мульти-модального анализа больших данных и извлечения знаний из различных источников».

Ряд исследований [4, 5] показал, что личностные черты могут вступать как в качестве предикторов и коррелянтов различных психических отклонений, поэтому определение личностных черт может использоваться как часть диагностики личностных и психических расстройств.

2. СУЩЕСТВУЮЩИЕ ПОДХОДЫ К АНАЛИЗУ ЛИЧНОСТНЫХ ХАРАКТЕРИСТИК ПОЛЬЗОВАТЕЛЕЙ СОЦИАЛЬНЫХ СЕТЕЙ

Существует ряд работ, которые рассматривают возможность определения личностных характеристик пользователя на основе слабо-структурированной информации.

В статье [6] авторы провели исследование личностей авторов различных блогов, предлагая пройти им психологический опрос. Было показано, что использование некоторых слов может быть связано с личностными характеристиками автора. В [7] было доказано, что используемые слова и структура текста могут отражать те или иные черты личности автора блога.

Проводились исследования по анализу не только текстовой информации, но и изображений. В работах [8, 9] авторы показали возможность применения новейших методов интеллектуального анализа данных при работе с изображениями с целью получения личностных характеристик пользователя.

Основной пласт существующих работ в данной области построены на основе англоязычной текстовой информации, однако также существуют работы [10, 11], которые анализируют информацию из популярных русскоязычных сообществ.

3. АЛГОРИТМ ПСИХОЛИНГВИСТИЧЕСКОГО АНАЛИЗА ДАННЫХ СОЦИАЛЬНЫХ СЕТЕЙ

Задача психолингвистического анализа – это задача бинарной классификации. Схематично предлагаемый в рамках данной работы подход к построению и обучению классификатора, обеспечивающего психолингвистический анализ данных социальных сетей с использованием машинного обучения представлен на рис. 1.

В предлагаемом подходе в качестве входных данных используется текст из «постов», которые публикует на своих страницах пользователи социальных сетей. Используются только «оригиналы» текстов, т.е. написанные самим пользователем социальной сети, а копии данного текстов на других страницах (так называемые «репосты») не учитываются. Тексты содержат личные мнения, рассуждения и мысли авторов.

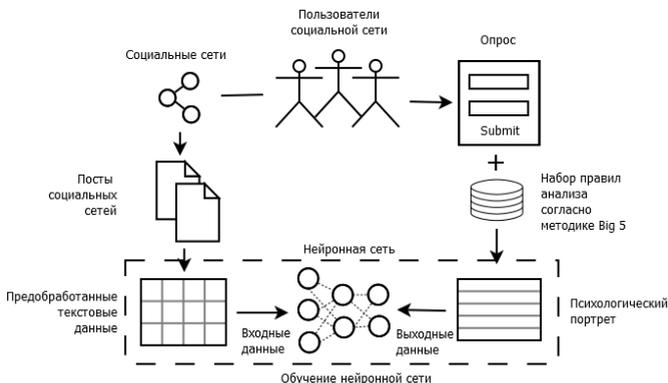


Рис. 1. Этапы построения и обучения классификатора

Основные источники текстов на страницах:

- сообщения (посты);
- текстовые статусы;
- комментарии к собственным и иным сообщениям.

Размеры текстов варьируются от одного-двух предложений до нескольких десятков предложений.

Для сбора выходных данных обучающей выборки в рамках предложенного подхода пользователям социальных сетей, тексты аккаунтов которых анализируются, предлагается пройти тест 5PFQ [12]. В данном тесте 75 вопросов. Опросник содержит парные высказывания, опрашиваемый определяет степень собственного согласия с тем или иным высказыванием по шкале 0 до 4.

Пример вопроса: «Всё новое вызывает у меня интерес» – «Часто новое вызывает у меня раздражение».

Полученные значения психологических характеристик пользователей составляют выходные значения для нейронной сети и соответствуют следующей модели:

$$Out = \{N, E, O, A, C\} \quad (1)$$

- N – Нейротизм, $|N|=2$;
- E – Экстраверсия, $|E|=2$;
- O – Открытость опыту, $|O|=2$;
- A – Согласие, или сотрудничество, $|A|=2$;
- C – Сознательность, или добросовестность, $|C|=2$.

Для определения психологических особенностей пользователя после обучения классификатора на вход подаются текстовые данные с его страницы в социальной сети, а на выходе получают его психоло-

гические характеристики. Таким образом, решается задача нахождения психологических особенностей пользователя по особенностям представления его мнения в социальных сетях.

4. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Были проведены эксперименты для оценки эффективности алгоритмов классификации текстов. Оцениваемые алгоритмы были основаны моделях SVM и RF, а входные данные были разбиты на обучающую и тестовую выборку в различных соотношении 70/30.

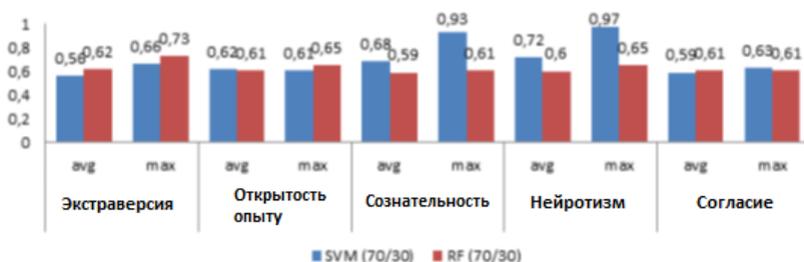


Рис. 2. Результаты экспериментов

Как видно из результатов экспериментов, метод опорных векторов показал лучшие результаты для всех текстов. Следует отметить, что в исследовании [13] поведение моделей на последних двух парах характеристик также показало наилучшие результаты.

5. ЗАКЛЮЧЕНИЕ

В рамках исследования был предложен подход к определению психологических характеристик пользователя социальных сетей посредством анализа текстовых сообщений в социальных сетях с использованием машинного обучения. Наилучшие средние результаты показал метод SVM. Для получения результатов с наивысшей эффективностью планируется применить иные алгоритмы классификации (LSTM, RNN), а также иные алгоритмы векторизации текста (в первую очередь, алгоритм BERT).

Литература

1. Польская Н.А., Якубовская Д.К. Влияние социальных сетей на самоповреждающее поведение у подростков // Консультативная психология и психотерапия. – 2019. – Т. 27, № 3. – С. 156-174. – Doi: 10.17759/cpp.20192703010
2. Наумов В.В. Анализ социальной структуры интернет-аудитории. Вестник Челябинского государственного университета. – 2012. – 35 (289). – С. 148-153.

3. Хайтун С.Д. Количественный анализ социальных явлений. – 3-е изд. – М.: КомКнига, 2010. – 280 с.
4. Widiger T.A., Mullins-Sweatt S.N. Clinical utility of a dimensional model of personality disorder // *Professional Psychology: Research and Practice*. – 2010. – 41 (6). – P. 488-494.
5. Widiger T.A., Costa P.T., Jr., R. R. McCrae. A proposal for Axis II: Diagnosing personality disorders using the five-factor model / In P.T. Costa, Jr. & T.A. Widiger (Eds.) // *Personality disorders and the five-factor model of personality*. – 2002. – P. 431-456. Washington, DC, US: American Psychological Association. – <http://dx.doi.org/10.1037/10423-025>.
6. Yarkoni T. Personality in 100,000 words: A large-scale analysis of personality and word use among bloggers // *Journal of research in personality*. – 2010. – Vol. 44, No. 3. – P. 363-373.
7. Soury A., Hosseinpour S., Rahmani A.M. Personality classification based on profiles of social networks' users and the five-factor model of personality // *Human-centric Computing and Information Sciences*. – 2018. – Vol. 8, No. 1. – P. 24.
8. Cristani M. Unveiling the multimedia unconscious: Implicit cognitive processes and multimedia content analysis // *Proceedings of the 21st ACM international conference on Multimedia*. – 2013. – P. 213-222.
9. Segalin C. The pictures we like are our image: continuous mapping of favorite pictures into self-assessed and attributed personality traits // *IEEE Transactions on Affective Computing*. – 2016. – Vol. 8, No. 2. – P. 268-285.
10. Павлыгин Э.Д., Подлобошников А.Г., Савинов Р.А., Ярушкина Н.Г., Наместников А.М., Филиппов А.А., Романов А.А., Мошкин В.С., Гуськов Г.Ю., Григоричева М.С. Разработка программного комплекса для интеллектуального анализа социальных медиа // *Автоматизация процессов управления*. – 2019. – № 2 (56). – С. 23-36. – DOI: 10.35752/1991-2927-2019-2-56-23-36.
11. Yarushkina N., Filipov A., Moshkin V., Namestnikov A., Guskov G. The social portrait building of a social network user based on semi-structured data analysis // *CEUR Workshop Proceedings/ 14th International Conference on Interactive Systems: Problems of Human-Computer Interaction, IS 2019*. – 2019. – Vol. 2475. – P. 119-129.
12. Хромов А.Б. Пятифакторный опросник личности: учебно-методическое пособие. – Курган: Изд-во Курганского гос. университета, 2000. – С. 23.
13. Wiggins J.S., Pincus A.L. Conceptions of personality disorders and dimensions of personality // *Psychological Assessment: A Journal of Consulting and Clinical Psychology*. – 1989. – 1 (4). – P. 305-316. – <http://dx.doi.org/10.1037/1040-3590.1.4.305>.

ОБРАБОТКА НОРМАТИВНЫХ ПРАВОВЫХ АКТОВ С ПРИМЕНЕНИЕМ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ*

*Аникин М.А., студент
МГТУ им. Н.Э. Баумана
e-mail: maxim.anikin1@yandex.ru*

*Королева М.Н., к.т.н., доцент
МГТУ им. Н.Э. Баумана
научный сотрудник
ИТЦ «Система»
e-mail: maria.svyatkina@gmail.com*

1. ВВЕДЕНИЕ

В настоящее время в отрасли права происходит переход от бумажной формы существования юридических норм к цифровой. Он берет свое начало от Концепции правовой информатизации России, утвержденной Указом Президента РФ от 28 июня 1993 г. N 966 [1]. Одной из главных задач, упомянутых в Концепции, является создание распределенной базы данных правовой информации и обеспечение интерактивного доступа юридических и физических лиц к ней. Данный процесс открывает возможности анализа оцифрованных нормативных правовых актов (НПА) с использованием методов обработки естественного языка [2, 3], что в свою очередь позволит автоматизировать стадии жизненного цикла НПА. Автоматизация работы с НПА является комплексной задачей, поэтому предлагается децентрализовать функции и параллельно создать систему, состоящую из нескольких модулей, взаимодействующих между собой.

В данной работе рассмотрены возможности применения современных методов машинного обучения в области обработки естественного языка для анализа нормативных правовых актов на русском языке, в частности, для получения векторного представления текста, а также определения именованных сущностей для упрощения поиска по документу и распознавания его структуры. Предложена функциональная модель системы обработки текста. Приведены результаты работы существующих предобученных моделей из проекта DeepPavlov [4] на выборке нормативных правовых актов с официального интернет-портала правовой информации [5].

* Исследование выполнено при частичной финансовой поддержке РФФИ и БРФФИ проект № 20-57-04002.

2. ОПИСАНИЕ ПРОБЛЕМЫ

Основная проблема – отсутствие учета контекста и использование вручную созданных свойств текста, приводящие в свою очередь к созданию систем, чувствительным к изменениям типового содержания документов.

На данный момент времени ведутся работы по повышению доступности и автоматизации действий в правовой сфере. Существующие методы поиска базируются на индексировании полей в базах данных. У индексов есть определенные недостатки: отсутствует учет контекста, из-за чего ухудшается способность к распознаванию близких по написанию слов, высокая чувствительность к опечаткам при запросе пользователя. Наиболее успешные системы поиска создаются на основе эмпирических формул, а также вручную созданных свойств: встречаемости слов в тексте, их расположения, системы приоритетов для них. Создание и сопровождение таких систем требует использования огромного количества рабочих ресурсов, и результатом является система, не обладающая гибкостью.

Изначально документ поступает в напечатанном виде с подписью, в противном случае он не имеет статуса официально опубликованного НПА. Автоматизация работ предполагает оцифровку документа. Для представления документа в виде файла необходимо выделять в нем структурные элементы. Данная задача также решается путем создания определенных правил, по которым происходит выделение. Правовая сфера отличительна тем, что используемые формулировки и стиль речи, используемый в тексте, четко определены. Тем не менее даже в этой сфере наблюдаются отступления от рекомендаций и учет всего их многообразия представляется крайне ресурсозатратным и приводит к созданию системы с высокой чувствительностью к ошибкам.

Работы, ведущиеся в области обработки естественного языка, предположительно могут решить проблему ответов на вопросы по тексту НПА, возникшие у пользователя. На данный момент напрямую ответы на вопросы по тексту можно получить только от специалистов или на определенных интернет-ресурсах, на которых обсужден интересующий пользователя вопрос.

Необходимо проводить эксперименты и создавать новые наборы данных для тренировки нейросетевых моделей, чтобы изучить степень применимости существующих моделей для решения практических задач.

3. ОПИСАНИЕ СИСТЕМЫ

Система обработки НПА состоит из трех модулей (рис. 1):



Рис. 1. Состав системы

1. Модуль векторного представления базируется на предобученной модели с архитектурой BERT [6]. Предполагается дальнейшее обучение на большом корпусе текстов НПА.
2. Модуль ранжирования документов принимает на вход текст документа и текстовый запрос пользователя. В результате выходит список с документами, распределенный по релевантности запросу. При этом из списка исключаются документы, релевантность которых была оценена как низкая.
3. Модуль определения именованных сущностей используется для получения разметки документа по тегам, которые являются полезными для пользователя, интересующегося нормативным правовым актом.

Архитектура BERT в настоящее время является крайне популярна ввиду своих высоких показателей при решении задач обработки естественного языка. Данная архитектура учитывает вхождение токенов наряду с окружающим их контекстом, что позволяет модели отличать омонимы и синонимичные слова, а также определять семантические отношения между словами.

4. АЛГОРИТМ РАБОТЫ МОДУЛЕЙ

4.1. МОДУЛЬ ПОЛУЧЕНИЯ ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ

Процесс получения векторного представления состоит из нескольких стадий. Первой стадией является токенизация текста, то есть разделение текста на токены – условно неделимые единицы текста, над которыми совершаются операции в нейросети. Затем токенизированный текст делится на части, которые можно подать на вход модели (максимальное количество токенов – 512). Токены подают на вход модели BERT и на выходе получают векторное представление текста.

4.2. МОДУЛЬ РАНЖИРОВАНИЯ ДОКУМЕНТОВ

Ранжирование документов [7–9] – задача, подразумевающая определение наиболее релевантных документов для входного запроса. На данный момент задача является нетривиальной, и в этом направлении активно ведутся исследовательские работы.

На вход предобученной модели [10] (рис. 2) приходит конкатенация двух последовательностей – запроса и части документа. В качестве параметра, оценивающего релевантность части документа запросу, используется представление токена [CLS] из последнего слоя модели. Данное представление направляется в полносвязную нейронную сеть, которая решает бинарную задачу классификации, рассчитывая вероятность релевантности части текста запросу. По рассчитанной вероятности производится ранжирование имеющихся документов.

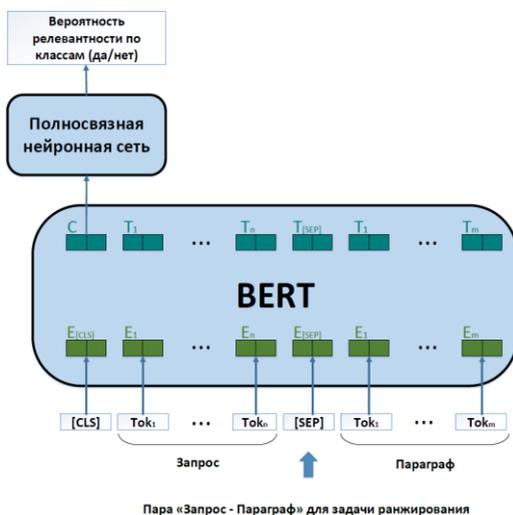


Рис. 2. Использование модели BERT для задачи ранжирования

Примером запроса может быть, например, строка “Концепция правовой информатизации России”. Параграфами будут являться абзацы документов или их части, если они большие по объему текста.

4.3. МОДУЛЬ ОПРЕДЕЛЕНИЯ ИМЕНОВАННЫХ СУЩНОСТЕЙ

Распознавание именованных сущностей [11, 12] – направление в обработке естественного языка с использованием искусственного интеллекта, суть которого заключается в нахождении различных сущностей в тексте и их классификации по заранее заданным катего-

риям, например: персоны, денежные единицы, наименования законов и др. Обычно эта задача сводится к классификации последовательности токенов.

Предсказание производится посредством передачи векторного представления первого токена каждого слова в полносвязную нейронную сеть, решающую задачу многоклассовой классификации.

В модели [13] (рис. 3) используется система разметки BIO. Токен помечается определенным тегом, предшествующим буквам «B» или «I», либо самостоятельной буквой-тегом «O». Обозначение букв в разметке следующее: «B» отвечает за токен, который является началом именованной сущности, «I» - за продолжение сущности, а «O» обозначается, что токен не принадлежит ни одной из определенных заранее именованных сущностей.

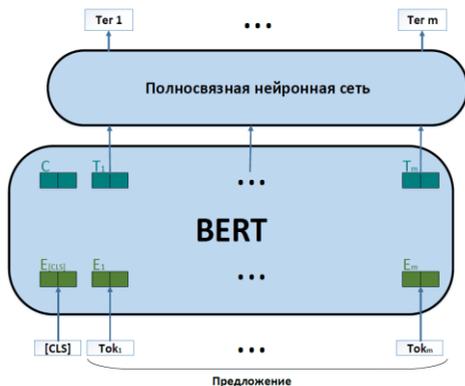


Рис. 3. Использование модели BERT для определения именованных сущностей

5. ДЕМОНСТРАЦИЯ РЕЗУЛЬТАТОВ

Приведем пример работы модуля получения векторного представления на простом предложении: «Я шёл домой по незнакомой улице».

В этом предложении были выделены следующие токены:

'[CLS]', 'Я', 'ш', '##ёл', 'домой', 'по', 'не', '##зна', '##ком', '##ой', 'улице', '.', '[SEP]'.

Пример векторного представления первого токена ([CLS]) (размерность вектора равна 768):

[0.074526, -0.3817189, 0.34543386, ..., -0.50326484, -0.22023882, -0.2767053]

Пример работы модуля определения именованных сущностей на примере абзаца из статьи 18 Федерального закона от 06.04.2011 N 63-ФЗ:

Текст абзаца:

2) с использованием единой системы межведомственного электронного взаимодействия, информационных систем органов государственной власти, Пенсионного фонда Российской Федерации, Федерального фонда обязательного медицинского страхования, единой информационной системы нотариата;

Выходной текст после определения в нем именованных сущностей:

2 (**B-CARDINAL**)) с использованием единой системы межведомственного электронного взаимодействия , информационных систем органов государственной власти , Пенсионного (**B-ORG**) фонда (**I-ORG**) Российской (**I-ORG**) Федерации (**I-ORG**) , Федерального (**B-ORG**) фонда (**I-ORG**) обязательного (**I-ORG**) медицинского (**I-ORG**) страхования (**I-ORG**) , единой информационной системы нотариата ;

Используемая в модуле модель может распознавать 18 именованных сущностей. В приведенном примере были распознаны следующие сущности: **CARDINAL** – число, которое не относится ни к одному типу чисел, определяемых моделью, **ORG** – компании, агентства, институты и т.д. Важно отметить, что модель способна определять страны как отдельную именованную сущность, но в примере слова «Российской Федерации» были правильно отнесены к государственному учреждению, а именно пенсионному фонду Российской Федерации.

Модуль ранжирования на данный момент не дал результатов, превосходящих современные методы ранжирования, использующие индексы в базах данных. Работы в данном направлении будут вестись дальше. Планируется испытание различных вариантов представления текста (как всего текста в целом с использованием токена [CLS], так и отдельных его абзацев) и дообучение на корпусе текстов НПА, что должно дать более хорошие результаты.

Модули системы в перспективе могут применяться для широкого ряда задач: выделения структурных элементов документа, поиска ответов на запрос пользователя по тексту, обнаружения интересующих именованных сущностей, а также для ранжирования документов по запросу.

6. ЗАКЛЮЧЕНИЕ

Разрабатываемые модули системы позволяют создать более гибкую систему поиска на основе автоматического распознавания необходимых сущностей. Для использования ранжирования документов

по произвольно составленным запросам пользователя будут проводиться эксперименты. Обучение используемых моделей на корпусе текстов НПА должно улучшить результаты работы модулей.

Литература

1. Указ Президента РФ от 28 июня 1993 г. N 966 "О Концепции правовой информатизации России" // Информационный правовой портал «Гарант». – URL: <http://pravo.gov.ru/proxy/ips/?docbody=&nd=102024537> (дата обращения: 28.04.2022).
2. Большакова Е.И., Клышинский Э С., Ландэ Д.В., Носков А.А., Пескова О.В., Ягунова Е.В. Автоматическая обработка текстов на естественном языке и компьютерная лингвистика: учеб. пособие – М.: МИЭМ, 2011. – 272 с.
3. Большакова Е.И., Воронцов К.В., Ефремова Н.Э., Клышинский Э.С., Лукашевич Н.В., Сапин А.С. Автоматическая обработка текстов на естественном языке и анализ данных: учеб. пособие – М.: Изд-во НИУ ВШЭ, 2017. – 269 с.
4. DeepPavlov: an open-source framework for text analysis – URL: <https://deeppavlov.ai/> (дата обращения: 05.04.2022).
5. Официальный интернет-портал правовой информации Российской Федерации. – URL: <http://pravo.gov.ru/> (дата обращения: 28.04.2022).
6. Devlin J., Chang M.-W., Lee K, Toutanova K. BERT Pre-training of Deep Bidirectional Transformers for Language Understanding // arXiv.org: an open-access archive. – 2019. – arXiv preprint arXiv:1810.04805v2 (дата обращения: 28.04.2022).
7. Qiao Y., Xiong C., Liu Z., Liu Z. Understanding the Behaviors of BERT in Ranking // arXiv.org: an open-access archive. – 2019. – arXiv preprint arXiv:1904.07531v4 (дата обращения: 28.04.2022).
8. Ma X., Guo J., Zhang R., Fan Y., Ji X., Cheng X. PROP Pre-training with Representative Words Prediction for Ad-hoc Retrieval // arXiv.org: an open-access archive. – 2020. – arXiv preprint arXiv:2010.10137v3 (дата обращения: 28.04.2022).
9. Nogueira R., Cho K. Passage Re-Ranking With BERT // arXiv.org: an open-access archive. – 2020. – arXiv preprint arXiv:1901.04085v5 (дата обращения: 28.04.2022).
10. Конфигурация BERT модели получения векторного представления предложения из проекта DeepPavlov. – https://github.com/deepmipt/DeepPavlov/blob/0.17.3/deeppavlov/configs/embedder/bert_embedder.json (дата обращения: 28.04.2022).
11. Liu Z., Jiang F., Hu Y., Shi C., Fung P. NER-BERT: A Pre-trained Model for Low-Resource Entity Tagging // arXiv.org: an open-access archive. – 2021. – arXiv preprint arXiv:2112.00405v1 (дата обращения: 28.04.2022).
12. Exploring Cross-sentence Contexts for Named Entity Recognition with BERT // arXiv.org: an open-access archive. – 2020. – arXiv preprint arXiv:2006.01563v2 (дата обращения: 28.04.2022).
13. Конфигурация NER-модели из проекта DeepPavlov. https://github.com/deepmipt/DeepPavlov/blob/0.17.3/deeppavlov/configs/ner/ner_ontonotes_bert_mult.json (дата обращения: 28.04.2022).

НЕЙРОННАЯ СЕТЬ ДЛЯ ОПРЕДЕЛЕНИЯ ВЫПОЛНИМОСТИ ФОРМУЛ БУЛЕВОЙ ЛОГИКИ*

*Апарнев А.Н., магистр, аспирант
ФГБОУ «НИУ «МЭИ»
e-mail: apich238@gmail.com*

1. ВВЕДЕНИЕ

Задача определения выполнимости формул булевой логики (также известная как *задача SAT*) состоит в определении, что у формулы есть модель. *Модель* – интерпретация формулы, на которой формула принимает значение 1 (ИСТИНА). *Интерпретация* – отображение, сопоставляющее каждой переменной либо значение 0 (ЛОЖЬ), либо 1.

Эта задача возникает в разных практических задачах, например при проектировании логических схем. К этой задаче сводятся задача планирования, другие NP-полные задачи, задачи логического вывода.

Задача SAT является NP-полной, существующие на данный момент точные алгоритмы её решения имеют экспоненциальную сложность и плохо ускоряются за счёт параллельных вычислений. Поэтому исследуется возможность решения этой задачи методами глубокого обучения, в частности – искусственными нейронными сетями (ИНС).

Например, в [1] разработана рекуррентная графовая нейронная сеть NeuroSAT, решающая задачу CNF-SAT как задачу бинарной классификации: для выполнимых формул нейронная сеть даёт на выходе значение близкое к 1, для невыполнимых – близкое к 0. Предлагается способ извлечения возможной найденной модели, но за ним нет либо обоснования и его результативность не гарантирована.

В другой работе [2] также была предложена рекуррентная графовая нейронная сеть для формул без ограничения на форму. Эта нейронная сеть состоит из 2 частей: первая генерирует нечёткую интерпретацию – сопоставление каждой переменной числа из отрезка $[0,1]$, а вторая – вычисляет значение этой формулы. Логические связки заменяются дифференцируемыми функциями. Такой подход позволяет обучить нейронную сеть находить модель выполнимой формулы без предъявления ей примеров моделей формул.

* Работа выполнена при материально-технической поддержке ЗАО «НОРСИ-ТРАНС».

Замена логических связок функциями вещественных чисел является примером нейросимвольной интеграции – направления искусственного интеллекта, стремящегося в рамках единой системы элементарно объединить методы коннективизма и формальных методов. Этот же подход лежит в основе нечёткой логики Л. Заде.

В нейросимвольных системах для замены чётких конъюнкции и дизъюнкции часто используются t-нормы и t-конормы одной из нечётких логик – минимаксной (логика Заде), вероятностной логики. Также возможно применить логику Лукасевича. В данной работе исследуется, как выбор t-нормы влияет на результат обучения глубокой рекуррентной сети, решающей задачу SAT, а также предлагается новая архитектура такой нейронной сети.

2. АРХИТЕКТУРА НЕЙРОННОЙ СЕТИ

Входными данными нейронной сети является формула, представленная в виде матрицы смежности **M** её дерева разбора, и двух векторов **C** и **N**, кодирующих логические связки. Например, для формулы $\neg(x_1 \wedge (x_1 \vee x_2)) \vee (\neg x_2 \vee \neg x_3)$ матрица **M** будет иметь вид табл. 1:

Таблица 1. Пример матрицы **M**

	\vee	\neg	\wedge	\vee	\vee	\neg	\neg	x_1	x_2	x_3
\vee	0	1	0	0	1	0	0	0	0	0
\neg	0	0	1	0	0	0	0	0	0	0
\wedge	0	0	0	1	0	0	0	1	0	0
\vee	0	0	0	0	0	0	0	1	1	0
\vee	0	0	0	0	0	1	1	0	0	0
\neg	0	0	0	0	0	0	0	0	1	0
\neg	0	0	0	0	0	0	0	0	0	1
x_1	0	0	0	0	0	0	0	0	0	0
x_2	0	0	0	0	0	0	0	0	0	0
x_3	0	0	0	0	0	0	0	0	0	0

Векторы **C** и **N** составляются в соответствии с табл. 2 и в этом примере примут значения **C**=(0,0,1,0,0,0,0) и **N**=(0,1,0,0,0,1,1). Порядок операций в них тот же, что и в строках и столбцах **M**: первой идёт корневая вершина, последними – переменные. Операции импликации предварительно заменяются по формуле $A \rightarrow B \Leftrightarrow \sim A \vee B$.

Таблица 2. Кодирование логических связок

C_i	N_i	Кодируемая операция
0	0	Дизъюнкция
0	1	Отрицание
1	0	Конъюнкция

Нейронная сеть определяется *параметрами* и *гиперпараметрами*. Под *параметрами* понимаются переменные, меняющиеся при обучении нейронной сети, другое их название – весовые коэффициенты. *Гиперпараметры* задаются 1 раз при создании и обучении глубокой нейронной сети и далее не изменяются.

Под *персептроном* понимается глубокая нейронная сеть прямого распространения с полносвязными слоями. Для описания архитектуры персептрона будет использоваться строка вида « $n_1 f_1 n_2 f_2 \dots n_d f_d$ », где n_i – количество нейронов на i -ом слое, f_i – функция активации i -ого слоя. Количество нейронов входного слоя определяется из контекста. f_i принимает значения sigmoid ($\sigma(x)=1/(1+\exp(-x))$), relu ($\text{ReLU}(x)=\max(0,x)$), identity ($\text{id}(x)=x$).

Каждой вершине графа сопоставляется вектор состояния \mathbf{Vs}_i^t размерности d , где i – индекс вершины, t – номер шага рекуррентной сети, а также вектор типа \mathbf{Vt}_i . Начальное значение \mathbf{Vs}_i^0 – нулевой вектор. Вектор \mathbf{Vt}_i принимает значение, зависящее от типа вершины: $\langle 1,0,0,0 \rangle$ для конъюнкции, $\langle 0,1,0,0 \rangle$ для дизъюнкции, $\langle 0,0,1,0 \rangle$ для отрицания, $\langle 0,0,0,1 \rangle$ для переменных. Значение \mathbf{Vt}_i не изменяется.

Векторы состояний вершин многократно изменяются в зависимости от состояния смежных вершин. А именно:

1. На основе векторов состояний вычисляются векторы – т.н. «сообщения» от каждой i -ой вершины к каждой смежной с ней j -ой вершине;
2. Новое состояние каждой вершины вычисляется на основе прежнего вектора состояния и суммы входящих «сообщений».

Этот подход известен, как фреймворк отправки сообщений. Рассмотрим более подробно его этапы.

Сообщение от i -ой вершине к j -ой вычисляется персептроном Msgup (формула « $d+4 \text{ relu } d \text{ ident}$ »), если вершина i в дереве ниже, чем j , и персептроном Msgdown (с такой же формулой) в противном случае. Кроме вектора состояния \mathbf{Vs}_i^t , им на входе подаётся вектор типа \mathbf{Vt}_i , поэтому на входе они принимают вектор размерности $d+4$.

Обозначим сумму входящих сообщений i -ой вершины на t -ом шаге рекуррентной нейронной сети как \mathbf{Inbox}_i^t . Новое состояние i -ой вершины вычисляется с помощью ячейки рекуррентной нейронной сети LSTM [3]:

$$(\mathbf{Vs}^{t+1}_i, \mathbf{Vcell}^{t+1}_i) = \text{LSTM}(\mathbf{Inbox}_i^t, (\mathbf{Vs}_i^t, \mathbf{Vcell}_i^t)),$$

где (...) – кортеж. Описанные действия выполняются для всех вершин графа и для всех шагов $t=0..R-1$. R – гиперпараметр.

После этого, среди полученных \mathbf{Vs}_i^R выбираются состояния тех вершин, которые представляют в формуле переменные. Выбранные векторы состояний обрабатываются персептроном Clv с формулой

«d relu 1 sigmoid». В результате для каждой переменной с помощью нейронной сети получено число из отрезка $[0,1]$. Объединим эти числа в вектор \mathbf{T} , i -й элемент которого соответствует переменной x_{i+1} .

Вектор \mathbf{T} есть некоторая интерпретация формулы в нечёткой логике. Степень истинности формулы вычисляется по правилам нечёткой логики с заданной t -нормой по алгоритму FuzzyEvaluation (рис. 1). Идея алгоритма состоит в том, что значения всех логических связок хранятся в векторе **OpsState** размерности O , и значение каждой операции обновляется в зависимости от значений дочерних вершин в дереве разбора формулы. Нулевой элемент вектора соответствует корневой вершине.

```

FuzzyEvaluation(M,C,N,T,D):
OpsState := нулевой вектор размерности  $O$ 
 $\mathbf{M}' := \mathbf{M}[:O]$ 
for  $i=1..D$ :
     $\mathbf{State} := [\mathbf{OpsState}|\mathbf{T}]$ 
     $\mathbf{G} := \mathbf{M}' \circ \mathbf{State}$ 
     $\mathbf{PreCon} := \text{maximums}(\text{OnesLike}(\mathbf{M}') - \mathbf{M}', \mathbf{G})$ 
     $\mathbf{Conjs} := \text{Conj}(\mathbf{PreCon})$ 
     $\mathbf{Disjs} := \text{Disj}(\mathbf{G})$ 
     $\mathbf{CDValues} := \mathbf{Conjs} \cdot \mathbf{C} + \mathbf{Disjs} \cdot (\text{OnesLike}(\mathbf{C}) - \mathbf{C})$ 
     $\mathbf{NValues} := (\text{OnesLike}(\mathbf{CDValues}) - \mathbf{CDValues}) \cdot \mathbf{N} + \mathbf{CDValues} \cdot (\text{OnesLike}(\mathbf{N}) - \mathbf{N})$ 
     $\mathbf{OpsState} := \mathbf{NValues}$ 
return  $\mathbf{OpsState}[0]$ 

```

Рис. 1. Алгоритм вычисления степени истинности формулы

Здесь, функции Conj и Disj для каждой строки матрицы вычисляют N -арные t -норму и t -конорму соответственно, используя одно из пар определений из табл. 3. Запись $\mathbf{X} \circ \mathbf{y}$ обозначает умножение каждого столбца матрицы \mathbf{X} на соответствующий элемент вектора \mathbf{y} , запись $\mathbf{x} \cdot \mathbf{y}$ обозначает произведение Адамара. Функция maximums поэлементно вычисляет максимальные элементы 2 матриц. D – число, большее или равное высоте дерева – максимального числа рёбер от корня до листа.

Таблица 3. Определения операций нечётких логик

Логика	t -норма	t -конорма
Минимаксная	$\min\{v_1, v_2, \dots, v_n\}$	$\max\{v_1, v_2, \dots, v_n\}$
Вероятностная	$v_1 \cdot v_2 \cdot \dots \cdot v_n$	$1 - (1 - v_1) \cdot (1 - v_2) \cdot \dots \cdot (1 - v_n)$
Лукаевича	$\max\{0, v_1 + v_2 + \dots + v_n - n + 1\}$	$\min\{1, v_1 + v_2 + \dots + v_n\}$

3. ДАННЫЕ

Для обучения и тестирования нейронных сетей были использованы сгенерированные данные – формулы с метками об их выполнимости.

Первым этапом генерации формулы является генерация дерева. Для этого, псевдослучайным образом генерируются состоящие из «0» и «1» строки b_i (i – номер строки). Каждая строка b_i задаёт путь от корневой до листовой вершины как цепочку переходов от текущей вершины к её левому (0) или правому (1) потомку. Если соответствующего потомка нет – он создаётся. Строки добавляются в дерево, пока в нём не будет $N=15$ листовых вершин. Пример генерации дерева с $N=9$ листьями приведён на рис. 2.

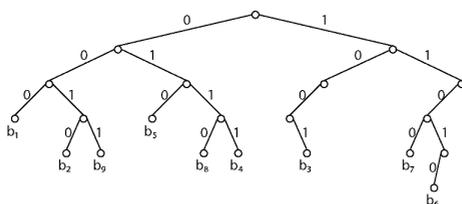


Рис. 2. Дерево, сгенерированное по ветвям $b_1=\langle 000 \rangle$, $b_2=\langle 0010 \rangle$, $b_3=\langle 1001 \rangle$, $b_4=\langle 0111 \rangle$, $b_5=\langle 010 \rangle$, $b_6=\langle 11010 \rangle$, $b_7=\langle 1100 \rangle$, $b_8=\langle 0110 \rangle$, $b_9=\langle 0011 \rangle$.

После того, как дерево сгенерировано, каждой его листовой вершине имя переменной вида « x_i », где i принимает значения от 1 до $n_{\text{vars}}=10$. Вершинам с одним потомком присписывается обозначающая отрицание строка «N». Остальным вершинам произвольным образом присписывается одна из 3 строк: «C» (обозначает конъюнкцию), «D» (обозначает дизъюнкцию), «I» (обозначает импликацию). Полученная формула записывается в префиксной форме, двойные отрицания удаляются. Приведём примеры полученных таким образом формул:

- $IDCDx1Cx2x3IDx1x4x5DNINx6x7x8DIINx5Nx4Nx9DNIx6Nx9x10$,
- $IDIIx1Nx2Nx3CCDNx4x5x6INx7Nx8CNINx1Nx9ICCx9x10x3NIx7Nx2$,
- $CDCIx1x2Nx3CNDNx4NCNx2Nx5CNDNx6x7x2DNCx8x9DINx1Nx10Ix9x3$,
- $CDCNx1DINx2x3Nx4DNx5IDNx3Nx6x7DDDx8Dx9x9x5Dx10Cx8Nx3$.

Большинство получаемых таким образом формул – выполнимые. В общем случае для обучения искусственной нейронной сети в задаче классификации, в данных должны быть представлены все классы, желательно – в относительно равном количестве.

Поэтому, на основе выполнимых формул генерируются невыполнимые. Для этого, выполнимая формула f приводится в дизъюнктивную нормальную форму (ДНФ) f' . Это делается с помощью алгоритма аналитических таблиц. На основе f' генерируется замыкание – формула h такая, что формула $g=f' \wedge h$ – невыполнима, и в сгенерированные данные добавляются формулы f и $f \wedge h$ с метками о выполнимости y_i 1 и 0 соответственно.

Замыкание h генерируется следующим образом. ДНФ f' представляется как массив m массивов литералов. Литерал – либо переменная, либо отрицание переменной. Этот массив сортируется по возрастанию длины вложенных массивов. Далее, из каждого вложенного массива (конъюнкта) k выбирается и добавляется во множество литералов r литерал l такой, что контрарный ему литерал в r ещё не входит. Если же в k таких нет – выбирается произвольный и цикл досрочно прекращается. Если r уже содержит какой-либо из литералов k – переходим к следующему конъюнкту. После этого формула $h = \sim(r_0 \vee r_1 \vee \dots \vee r_m)$ – замыкание, так как либо противоречит каждому конъюнкту из f' , либо невыполнима сама по себе.

Рассмотрим примеры генерации замыканий.

Пример 1. Допустим, имеется $f = \sim((x_2 \rightarrow x_3) \wedge x_1 \wedge (x_3 \rightarrow x_2))$. ДНФ этой формулы будет $f' = \sim x_2 \wedge x_3 \vee \sim x_1 \vee x_2 \wedge \sim x_3$, что представляется в виде массива m как $[[\sim x_2, x_3], [\sim x_1], [x_2, \sim x_3]]$. После сортировки, массив примет вид $m = [[\sim x_1], [\sim x_2, x_3], [x_2, \sim x_3]]$. Изначально, множество r пусто.

Рассмотрим первый конъюнкт $k = [\sim x_1]$. r не содержит x_1 , поэтому добавляем в r $\sim x_1$: $r = \{\sim x_1\}$.

Рассмотрим второй конъюнкт $k = [\sim x_2, x_3]$. r не содержит x_2 или $\sim x_3$, поэтому выбираем и добавляем произвольный литерал: $r = \{\sim x_1, \sim x_2\}$.

Рассмотрим третий конъюнкт $k = [x_2, \sim x_3]$. r не содержит литералов из k , но добавление x_2 в r менее желательно, поэтому список кандидатов сокращается до $[\sim x_3]$, из которого единственным образом выбирается и добавляется в r литерал $l = \sim x_3$: $r = \{\sim x_1, \sim x_2, \sim x_3\}$.

Все конъюнкты обработаны. Замыкание $h = \sim(\sim x_1 \vee \sim x_2 \vee \sim x_3)$.

Пример 2. Допустим, по некоторой формуле f мы получили ДНФ $f' = x_4 \wedge x_2 \wedge \sim x_3 \vee \sim x_1 \wedge x_3 \vee x_3 \wedge \sim x_4 \wedge x_5 \wedge \sim x_2 \vee \sim x_4 \vee x_1 \wedge \sim x_2 \wedge x_4 \vee \sim x_1 \wedge x_5$. Имеем следующее представление этой формулы в виде массива: $m = [[x_4, x_2, \sim x_3], [\sim x_1, x_3], [x_3, \sim x_4, x_5, \sim x_2], [\sim x_4], [x_1, \sim x_2, x_4], [\sim x_1, x_5]]$. После сортировки получаем $m = [[\sim x_4], [\sim x_1, x_3], [\sim x_1, x_5], [x_4, x_2, \sim x_3], [x_1, \sim x_2, x_4], [x_3, \sim x_4, x_5, \sim x_2]]$. Множество r – пусто.

Возьмём первый конъюнкт $k = [\sim x_4]$, выберем из него $l = \sim x_4$ единственным образом и добавляем в r : $r = \{\sim x_4\}$.

Из следующего конъюнкта $k = [\sim x_1, x_3]$ мы можем выбрать любой литерал. Выберем $l = \sim x_1$ и добавим в r : $r = \{\sim x_1, \sim x_4\}$.

В следующем конъюнкте $k = [\sim x_1, x_5]$ уже содержится литерал из r , поэтому сразу переходим к следующему.

Конъюнкт $k=[x_4, x_2, \sim x_3]$ содержит литеру x_4 , контрарную к литере из r , поэтому выбирать будем из оставшихся двух: $x_2, \sim x_3$. Выбираем $l=x_2$ и добавляем в r : $r=\{\sim x_1, x_2, \sim x_4\}$.

В следующем конъюнкте $k=[x_1, \sim x_2, x_4]$ все литеры из k имеют контрарных себе в r . Выбираем из них произвольную $l=x_1$ и выходим из цикла. $r=\{x_1, \sim x_1, x_2, x_4\}$.

Имеем замыкание $h=\sim(x_1 \vee \sim x_1 \vee x_2 \vee x_4)$.

Описанным образом было сгенерировано 70000 формул. Эта выборка была произвольным образом назделена на 50000 обучающих примеров, 10000 валидационных и 10000 тестовых. При этом, в каждом подмножестве поровну выполнимых и невыполнимых примеров.

4. ОБУЧЕНИЕ

Процесс обучения глубокой нейронной сети – это процесс минимизации её ошибок. Для этого, на обучающих данных вычисляются ответы нейронной сети y_p . На основе этих и истинных ответов y_t вычисляется значение функции потерь – в данном случае бинарной перекрёстной энтропии $VSE(y_t, y_p) = -y_t \ln(y_p) - (1-y_t) \ln(1-y_p)$, а по нему – среднее значение ошибки $L(\mathbf{w})$, где \mathbf{w} – вектор параметров нейронной сети. Методом обратного распространения ошибки вычисляется градиент $\nabla L = \langle \partial L / \partial \mathbf{w}_0, \partial L / \partial \mathbf{w}_1, \dots, \partial L / \partial \mathbf{w}_p \rangle$, на основании которого алгоритм градиентного спуска Adam выполняет шаг оптимизации, меняя параметры \mathbf{w} . Во избежание т.н. взрыва градиента, компоненты ∇L , по модулю большие 0.2, отсекаются до этого значения.

Обучение каждой нейронной сети длится 30 эпох. В течении одной эпохи каждый обучающий пример предъявляется нейронной сети ровно один раз. Обучающие примеры группируются в пакеты по 250 штук и предъявляются одновременно.

Список постоянных значений гиперпараметров приведён в таб. 4.

Таблица 4. Значения гиперпараметров.
Для гиперпараметров, не перечисленных здесь, либо используются различные значения, либо значения по умолчанию

Гиперпараметр	Значение
Алгоритм оптимизации	Adam
Порог отсечки градиента	0.2
Скорость обучения	0.0004
Скорость уменьшения весов	0.0001
Длительность обучения	30 эпох
Примеров в одном пакете	250
Размерность векторов \mathbf{d}	16
Количество итераций рекуррентной НС	10

Каждая обученная ИНС проверяется на валидационном наборе данных. Качество работы обученной с заданными гиперпараметрами ИНС проверяется по следующим метрикам:

- ACC – точность, доля правильно классифицированных примеров. Пример классифицируется как выполнимый, если ответ ИНС $y_p > 0.5$.
- FPR – доля невыполнимых формул, классифицированных как выполнимые.
- FNR – доля выполнимых формул, классифицированных как невыполнимые.

Обученные ИНС отличаются по одному гиперпараметру – определению t-нормы (и двойственной ей t-конормы).

5. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Изменение точности обучаемых ИНС можно видеть на графике (рис.3). Первый вывод, который можно сделать из этого графика – даже необученные ИНС с описанной архитектурой дают точность более 80%. 50% точности объясняются тем, что из-за вычисления значения формул в ИНС, невыполнимые формулы сложно принять за выполнимые, что подтверждает график ложно-положительных ошибок (рис. 4).

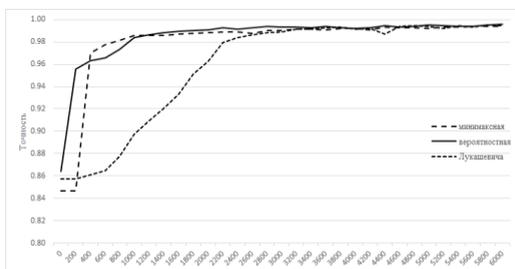


Рис. 3. Точность ИНС на валидационной выборке в процессе обучения

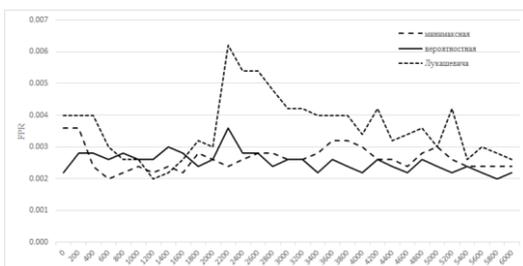


Рис. 4. Значения FPR обучаемых ИНС на валидационной выборке

Оставшиеся >30% можно объяснить низким порогом принятия решения о выполнимости (0.5) и способностью нечётких логик оценивать формулы как скорее выполнимые, на произвольных интерпретациях. В качестве иллюстрации приведём процесс изменения ответа у необученных нейронных сетей для выполнимой формулы (рис. 5,6,7). На таких диаграммах для каждого шага $t=0,1,\dots,R$ указаны даваемые ИНС значения переменных, и значение ур формулы от этих переменных. Сама формула указана сверху, в префиксной форме. SAT обозначает выполнимую формулу, UNSAT – невыполнимую.

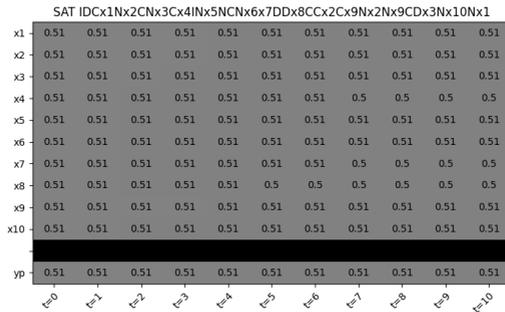


Рис. 5. Вычисление ответа необученной ИНС с использованием минимаксной логики. Вверху – обрабатываемая формула. ур – ответ ИНС

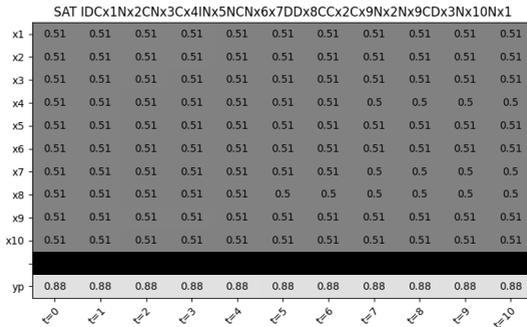


Рис. 6. Вычисление ответа необученной ИНС с использованием вероятностной логики

	SAT	IDC	x1N	x2CN	x3Cx	x4IN	x5NC	x6N	x7DD	x8CC	x2Cx	x9N	x2Nx	x9CD	x3Nx	x10Nx
x1	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51
x2	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51
x3	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51
x4	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
x5	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51
x6	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51
x7	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
x8	0.51	0.51	0.51	0.51	0.51	0.51	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
x9	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51
x10	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51
ур	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	t=0	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	t=9	t=10					

Рис. 7. Вычисление ответа необученной ИНС с использованием логики Лукасевича

Как видим, одна и та же формула с одними и теми же значениями переменных в разных логиках имеет значение от 0.51 до 1.

Второй вывод, который можно сделать из графика на рис. 3 – процесс обучения при использовании t-нормы логики Лукасевича существенно медленнее, чем для остальных логик. Возможной причиной этого может быть то, что t-конорма (для t-нормы это верно в силу двойственности) Лукасевича имеют ненулевую производную только в малой части пространства, а именно – под стандартным симплексом размерности n, объём которого вычисляется по формуле $1/(n+1)!$. Так, для 2 переменных объём с единичной (кроме 0 это единственное принимаемое производной значение) производной составляет 1/2. В результате, для отдельной формулы может быть получен нулевой градиент, и многие такие примеры на процесс обучения не влияют, либо начинают влиять не сразу.

Другие две t-нормы лишены такого недостатка: при минимаксной логике, значение формулы равно либо значению одной из переменных x_i , либо $1-x_i$; при вероятностной – t-норма принимает значение 0 только если один из аргументов принял значение 1, что в данном случае практически невозможно из-за свойств функции σ .

Третий вывод, который можно сделать из графика на рис. 3 – в результате обучения получены ИНС, дающие правильные ответы в большинстве случаев. В частности, на валидационном наборе данных получены показатели точности 99,36% при выборе минимаксной логики, 99,55% при выборе вероятностной и 99,47% – при выборе логики Лукасевича.

Рассмотрим пример, как в процессе вычислений меняется ответ обученной ИНС. Обычно, правильный ответ находится уже за 5 шагов. В некоторых случаях, ИНС после приближения к правильному ответу может перейти к неправильному. На рис. 8 показан пример такого поведения.

	SAT ICDINx1x2Dx3x4DNINx5Cx3Nx1DNx1Nx6IDNx5x7Nx5NCIx8x9x10										
x1	0.48	0.89	0.61	0.02	0.0	0.0	0.0	0.0	0.01	0.23	0.3
x2	0.48	0.58	0.97	1.0	1.0	1.0	1.0	0.98	0.33	0.02	0.01
x3	0.48	0.97	1.0	1.0	1.0	1.0	0.99	0.92	0.11	0.05	0.32
x4	0.48	0.58	0.96	0.99	1.0	1.0	1.0	0.99	0.43	0.03	0.01
x5	0.48	0.89	0.9	0.18	0.01	0.0	0.09	0.91	1.0	1.0	1.0
x6	0.48	0.89	0.98	0.97	0.35	0.01	0.0	0.0	0.01	0.49	0.98
x7	0.48	0.89	0.98	0.97	0.43	0.02	0.24	0.88	0.99	1.0	1.0
x8	0.48	0.89	0.98	0.85	0.23	0.02	0.03	0.58	0.98	1.0	1.0
x9	0.48	0.58	0.99	1.0	1.0	1.0	0.83	0.07	0.01	0.0	0.0
x10	0.48	0.95	0.99	1.0	0.99	0.95	0.39	0.02	0.0	0.0	0.0
ур	0.52	0.89	0.9	0.18	0.01	0.05	0.61	0.98	1.0	1.0	1.0
	t=0	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	t=9	t=10

**Рис. 8. Пример работы ИНС с временным провалом.
t-норма – минимум**

6. ЗАКЛЮЧЕНИЕ

Одно из перспективных направлений исследований в области искусственного интеллекта – интеграция машинного обучения и формальных рассуждений. Формальной компонентой при этом часто выступает нечёткая логика с t-нормой. Показано, что для интеграции не все нечёткие логики одинаково хороши – в частности, возникают ощутимые проблемы при использовании t-нормы Лукасевича в гибридной системе.

Исходный код нейронной сети для решения задачи SAT доступен публично по адресу <https://github.com/Apich238/mySAT>.

Литература

1. Selsam D., Lamm M., Bünz B., Liang P., de Moura L., Dill D.L. Learning a sat solver from single-bit super-vision // arxiv.org. 2018. – URL: <https://arxiv.org/abs/1802.03685> (дата обращения: 08.04.2022). – P. 11.
2. Amizadeh S., Matushevych S., Weimer M. Learning to solve circuit-sat: an unsupervised differentiable approach. – 2018. – URL: <https://openreview.net/pdf?id=BJxgz2R9t7> (дата обращения: 08.04.2022). – P. 14.
3. Hochreiter S., Schmidhuber J. Long short-term memory // Neural Computation. – 1997. – Vol. 9, No. 8. – P. 1735-1780.

ПРЕДСКАЗАНИЕ НАМЕРЕНИЙ КОНКУРЕНТА В СРЕДЕ С ПРОТИВОДЕЙСТВИЕМ

*Городецкий В.И., д.т.н. профессор
АО «Эврика»
e-mail: vladim.gorodetsky@gmail.com*

1. ВВЕДЕНИЕ

Эта проблема имеет долгую историю – около 80 лет назад она породила теорию игр, исследование операций, корпорацию RAND, а в 1968 г. была опубликована ныне классическая книга Т. Саати [1], в которой математически предсказана современная ситуация в треугольнике США–Китай–Россия. Новый всплеск интереса к этой проблеме появился в 1990-х, когда DARPA финансировала программу исследований в области группового управления, которые велись в ведущих университетах США, Европы и Азии. Тогда на смену математическому подходу, характерному для предыдущего этапа, пришли компьютерные модели искусственного интеллекта (ИИ).

В последнее десятилетие проблема управления в среде с противодействием вновь вышла на первый план по ряду причин. С одной стороны, появилось убеждение, что современные компьютерные системы и технологии ИИ вполне могут справиться с задачами противоборства сложных систем. С другой стороны, эта проблема стала более острой для многих критических приложений. Примерами таких приложений являются конкурентная борьба компаний и даже государств на глобальных рынках, где эта борьба часто выходит за пределы рыночных отношений; операции в киберпространстве, когда одна из сторон пытается нарушить безопасность критически важной компьютерной системы или инфраструктуры, безопасность её данных, а другая сторона стремится предотвратить это в реальном времени. Подобные операции составляют основу гибридных войн с дезинформацией и психологическим воздействием на какую-то группу населения одной или ряда стран. Ярким примером таких войн, проводимых параллельно в физическом и виртуальном пространствах, являются «цветные революции». Этот список можно продолжить, и он пополняться новыми приложениями.

Несмотря на различие приведённых приложений, все они могут быть представлены одной и той же абстрактной моделью, причём не только концептуально, но и на уровне решаемых ими задач и архитектуры программного обеспечения.

В работе рассматривается одна из принципиальных задач проблемы конкуренции двух систем в общей среде – задача предсказания намерений конкурента. В ней описывается разработанный подход к решению этой задачи, а также обобщённая функциональная архитектура системы, реализующая этот подход. Новизна предложенного подхода состоит в том, что он использует концепцию цифрового двойника *поведения* конкурента, который задействуется как на этапе машинного обучения системы предсказания его намерений, так и на этапе принятия решений. Методологическая новизна работы состоит в применении сценарной модели поведения конкурента как средства структуризации группового поведения распределённых объектов, которые формируют соперничающую систему.

2. ОСОБЕННОСТИ ПРИЛОЖЕНИЙ

Далее для описания моделей группового поведения объектов будет использоваться нижеследующая терминология:

Намерение системы – преимущество или качество другого рода, которое стремится получить система в результате исполнения миссии.

Состояние системы, объекта – время, список значений её атрибутов.

Цель системы – множество состояний которое она стремится достигнуть для реализации её намерения; цели объектов, составляющих систему, являются компонентами дерева целей системы в целом.

Сценарий (миссии) – структурированная последовательность переходов состояний объектов системы из исходного состояния в целевое.

Индивидуальное поведение объекта – последовательность задач, которые он решает в составе группы при исполнении сценария миссии.

Траектория объекта в сценарии – многомерная функция значений его атрибутов в зависимости от времени процесса выполнения сценария.

Паттерн поведения – последовательность действий объекта/системы.

Групповое поведение – структурированное в пространстве и во времени множество паттернов поведения объектов группы.

Рассмотрим отличительные черты систем, в которых в общей среде функционируют две стороны с антагонистическими интересами.

1. Такие системы имеют иерархическую *структуру* – они состоят из большого числа *автономных*, обычно, *мобильных* физических, виртуальных и/или социальных объектов, обладающих встроенными вычислительными и коммуникационными возможностями.

2. Они обладают групповой динамикой, но конкуренту доступна информация только о поведении отдельных его объектов.

3. Поведение обеих сторон является целенаправленным – оно управляется *намерением*, которое сторона стремится реализовать, например, получить конкурентное преимущество, предсказать намерение конкурента, нанести ему некоторый вред, дискредитировать и т.п.

4. Каждая сторона реализует своё намерение с помощью *сценария группового поведения*, переводящего её объекты в некоторое целевое состояние из заданного множества.

5. Намерение может быть реализовано разными сценариями при неизменном множестве целевых состояний. Сценарий может трансформироваться в процессе исполнения в зависимости от реакции конкурента и случайных факторов. Например, очередной шаг группы хакеров зависит от информации, добытой об атакуемой сети. Очередной шаг «цветной революции» будет зависеть от противодействия оппонента. Новое осложнение у больного COVID-19 зависит от того, какие «слабые места» больного «обнаружил» вирус и от ранее предпринятого лечения.

6. Все такие системы функционируют в реальном времени.

7. Объекты системы распределены в пространстве, поэтому для координации поведения им нужен интенсивный обмен сообщениями.

Перечисленные черты конкурирующих систем оказывают решающее влияние на поведение каждой стороны и на процессы управления.

3. ПОСТАНОВКА ЗАДАЧИ ПРЕДСКАЗАНИЯ НАМЕРЕНИЙ

Обозначим конкурирующие стороны символами **A** и **B** и будем их называть системами. Эти системы имеют несовместимые интересы, и каждая из них стремится реализовать свои намерения, которые могут представлять ту или иную угрозу для противоположной стороны. По этой причине каждая система должна параллельно решать две задачи:

(1) управлять реализацией своих намерений, и

(2) прогнозировать намерения и поведение соперника с целью оценки угрозы, исходящей от него, и принимать контрмеры.

Далее будет рассматриваться только задач (2) и, для определённости, будем полагать, что её решает система **B**, для которой система **A** является конкурентом с вредоносными намерениями. Рассмотрим компоненты и содержательную постановку этой задачи.

Пусть $\hat{A} = \{A_1, \dots, A_n\}$ – множество объектов, составляющих систему **A**. Полагаем, что эти объекты распределены в пространстве, физическом или виртуальном, являются автономными, и некоторые из

них могут быть мобильными. Полагаем также, что они способны выполнять определённые действия и, при необходимости, координировать своё пространственно-временное поведение в интересах реализации намерений системы A . Объекты множества $\{A_1, \dots, A_n\}$ формируют нижний уровень по отношению к системе A , а система A в этой иерархии выполняет целеполагание для объектов A_1, \dots, A_n и управляет, при необходимости, их групповым поведением при реализации своего намерения. Объекты A_1, \dots, A_n могут находиться в различных отношениях, иметь набор свойств и способностей и пространственные координаты, например, широту и долготу, зависящие от времени.

Введём обозначения, необходимые для описания формальной постановки задачи предсказания намерений системы A .

$I = \{I_1, \dots, I_m\}$ – множество возможных намерений системы A ;

$\widehat{X} = \{\widehat{X}(I_1), \dots, \widehat{X}(I_m)\}$ – множество классов целевых состояний $\widehat{X}(I_s)$ системы A , где $\widehat{X}(I_s) \in \widehat{X}$ – множество *целевых состояний* этой системы для намерения $I_s \in I$, при этом достижение состояния $X \in \widehat{X}(I_s)$ отождествляется системой A с реализацией её намерения $I_s \in I$.

$G = \langle X, V \rangle$ – направленный граф переходов состояний системы A , где X – множество классов её состояний, $V = \{V_{p,s}\}$ – множество дуг.

Полагаем, что переходы состояний $V_{p,s}$ выполняются объектами из множества $\{A_1, \dots, A_n\}$ с помощью сценария $S_{p,q}$ группового поведения этих объектов, $S_{p,q} \in \widehat{S}(I_p, I_q)$, где $\widehat{S}(I_p, I_q)$ – класс сценариев. Если для каждого класса состояний $\widehat{X}(I_1), \dots, \widehat{X}(I_m)$ и класса сценариев группового поведения $\widehat{S}(I_p, I_q) \in \widehat{S}$ имеются множества примеров, то задачи определения целевых состояний и сценариев группового поведения системы A сводятся к задачам машинного обучения классификации.

Разобьём множество намерений $I = \{I_1, \dots, I_m\}$ на два подмножества I_0 и I_d , $I_0 \cup I_d = I$, и назовём I_0 множеством намерений, не несущих угрозы системе B , а множество I_d – несущих угрозу. Множество намерений системы A можно разбить и на большее число классов с введением градаций «степени угрозы», но для простоты обозначений здесь описывается постановка задачи с двумя классами угроз. Поскольку каждому $I_s \in I$ поставлен в соответствие класс целевых состояний $\widehat{X}(I_s)$, то множеству намерений I_0 системы A , не представляющих угрозы для системы B , можно поставить в соответствие множество его целевых состояний \widehat{X}_0 , а множеству намерений повышенной угрозы I_d – множество целевых состояний \widehat{X}_d . Аналогично разо-

бъём множество классов сценариев. Тогда сценарии $S_{p,q} \in \widehat{S}_0$ будут отвечать безопасным сценариям системы A , а $S_{p,q} \in \widehat{S}_d$ – тем, которые представляют угрозу для B .

Рис. 1 иллюстрирует введённые понятия, а также классификацию намерений, целевых классов состояний системы A и классов сценариев её группового поведения. Поскольку в описанной модели классы целевых состояний и сценариев группового поведения однозначно связаны с намерениями системы A , то задачу предсказания её намерений и оценки угрозы, исходящих от них для системы B , можно сформулировать в двух вариантах:

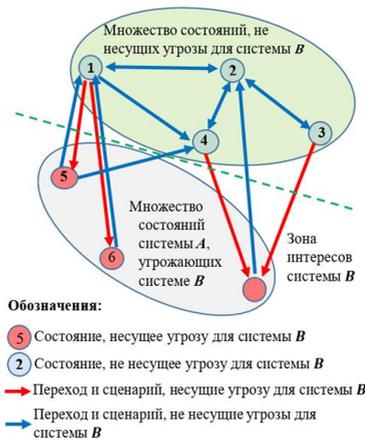


Рис. 1. Иллюстрация понятий классов состояний, классов сценариев переходов состояний и их классификация

Задача 1. Предсказать класс целевого состояния конкурента.

Задача 2. Определить класс сценария его группового поведения.

Продемонстрируем введённые понятия и поясним существо задач 1 и 2 на примере. На рис. 1 система A имеет семь классов состояний. Четыре из них $\widehat{X}_0 = \{\widehat{X}(I_1), \dots, \widehat{X}(I_4)\}$ не опасны для B (они окрашены в синий цвет), а три других, формирующих $\widehat{X}_d = \{\widehat{X}(I_5), \widehat{X}(I_6), \widehat{X}(I_7)\}$ (красного цвета) представляют для него угрозу. Этим подмножествам классов целевых состояний отвечают подмножества намерений $I_0 = \{I_1, I_2, I_3, I_4\}$ и $I_d = \{I_5, I_6, I_7\}$ с такими же значениями индексов. На этом же рисунке представлен граф допустимых переходов из одного класса состояний в другое. Все дуги этого графа, которые входят в вершины множества I_0 (это дуги синего цве-

та) отвечают классам сценариев группового поведения, не несущим угрозы для системы **B**. Все остальные дуги (они имеют узлом назначения состояния из множества \widehat{X}_d) отвечают сценариям, несущим угрозу системе **B**. К первой группе относятся сценарии из множества $\widehat{S}_0 = \{\widehat{S}(I_1, I_2), \widehat{S}(I_1, I_4), \widehat{S}(I_2, I_1), \widehat{S}(I_2, I_3), \widehat{S}(I_2, I_4), \widehat{S}(I_3, I_2), \widehat{S}(I_4, I_2), \widehat{S}(I_5, I_1), \widehat{S}(I_5, I_4), \widehat{S}(I_6, I_1), \widehat{S}(I_7, I_2)\}$, а ко второй группе – сценарии группового поведения из множества $S_d = \{\widehat{S}(I_1, I_5), \widehat{S}(I_1, I_6), \widehat{S}(I_3, I_7), \widehat{S}(I_4, I_7)\}$.

4. МОДЕЛЬ БОЙДА

В соответствии с моделью враждебного взаимодействия двух систем, предложенной Дж. Бойдом (J. Boyd) [2], системы **A** и **B** в процессе выполнения своих функций повторяют стандартный цикл действий из четырёх фаз: *восприятие, оценка и прогноз ситуации, принятие решения, реализация решения* (рис. 2). Эта модель, предложенная первоначально для моделирования и управления военными операциями, оказалась работоспособной и в широком классе конкурентного поведения систем типа модели конкуренции на рынке, где она активно используется.

В модели Бойда каждая из конкурирующих сторон на метуровне должна структурировать своё поведение в соответствии с этим циклом, и для успеха в соперничестве каждая сторона должна отслеживать смену фаз цикла конкурента в реальном времени. Исследования этой модели показали, что в ней «побеждает» та сторона, которая действует рационально и опережает соперника в каждом цикле.

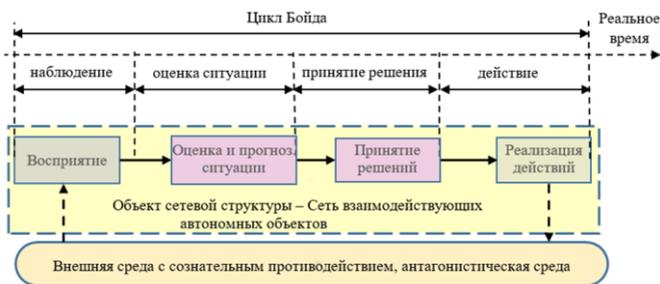


Рис. 2. Цикл Бойда системах реального времени

В задаче оценки и предсказания намерений системы **A** цикл Бойда определяет для **B** временные рамки процессов обновления информации о конкуренте, её обработки и принятия решений.

5. ИСХОДНАЯ ИНФОРМАЦИЯ И ИСТОЧНИКИ ЗНАНИЙ

Для корректности постановки задачи проанализируем источники информации о состоянии и поведении системы **A**, которую система **B** может использовать в алгоритме предсказания, а также источники знаний, доступные ей в задаче предсказания намерений соперника.

5.1. ИСТОЧНИКИ И ФОРМА ПРЕДСТАВЛЕНИЯ ИСХОДНОЙ ИНФОРМАЦИИ

Система **B** в процессе «наблюдения» за объектами **A** может получать о них информацию в разной форме и из разных источников, например, в виде текстов, собираемых из социальных медиа, средств массовой информации и специальных источников (это характерно для задач противодействия «цветным революциям»), или с помощью сенсоров, например, средств и методов диагностики (в задаче предсказания развития болезни и осложнений у больных, инфицированным вирусом), или такая информация формируется с помощью обмена сообщениями между объектами, составляющими систему компьютерной безопасности **B** (в задаче противодействия групповой атаке хакеров на компьютерную сеть) и т.п. Обычно эта информация является весьма фрагментарной, неполной, непредсказуемой по составу атрибутов и содержать возмущения. Она может оказаться слишком бедной и недостоверной для того, чтобы с её помощью можно было предсказывать намерения оппонента и целевые состояния его поведения с нужной точностью.

5.2. ИСТОЧНИКИ ЗНАНИЙ

Проанализируем источники знаний о системе **A**, которые могут быть использованы для предсказания её намерения и оценки их угрозы.

а) *Онтология*, представляющая знания системы **B** (1) об организационной структуре системы **A**, о составе, структурах и атрибутах её компонент A_1, \dots, A_n ; (2) о типовых классах состояний этих объектов и их атрибутах; (3) о множестве вариантов намерений подсистемы **A** и об уровне угрозы, которую каждое намерение может представлять для системы **B**; (4) о множестве целевых классов состояний системы **A** в зависимости от реализуемых намерений и, возможно, (5) о классах сценариев группового поведения объектов A_1, \dots, A_n , которые используются системой **A** для достижения целевых состояний, определяемых её намерениями. Онтология является важной компонентой знаний, которые могут включать знания экспертов и знания из семантического веба.

б) *Множество примеров наблюдений*, которые могут быть семантически размеченными (для каждого пример указаны намерения, которые при этом стремилась реализовать или реализовала система A , или классы целевых состояний примера) или неразмеченными. Примеры являются источником знаний, которые можно извлечь из них с помощью машинного обучения, рассматривая множество примеров как обучающую выборку. Но эта выборка в данной задаче имеет специфику:

- Она обычно имеет малый объём и огромную размерность пространства признаков; т.е. она является выборкой «малых данных». Это исключает применение нейросетевых моделей глубокого обучения, которые способны извлекать из данных поведенческие знания.
- Прогнозирование намерений требует знаний о групповом поведении объектов системы A , однако, методы машинного обучения, способные извлекать сценарные знания из малой выборки, отсутствуют.
- Для извлечения поведенческих знаний нужно иметь данные, содержащие такие знания. В [3] утверждается, что извлечение знаний о поведении из данных – это особая задача, поскольку поведенческие аспекты не всегда в данных представлены.
- В доступной выборке пропущенных значений может быть больше, чем означенных, и состав таких атрибутов примера случаен.

Важный вывод, который следует из анализа особенностей доступного множества примеров состоит в том, что нужно искать дополнительные источники данных. Ими могут быть математические модели группового поведения объектов A_1, \dots, A_n , составляющих систему A .

в) *Экспертные знания*. С задачами описываемого класса обычно работают эксперты. Они имеют богатый опыт в предметной области, и знают особенности поведения соперника, типовые сценарии поведения его объектов при различных намерениях и состояниях внешней среды. Их знания включают индикаторы (комбинации признаков, паттерны поведения соперника, тенденции в их динамике) и экспертные правила принятия решений на основе этих индикаторов. Обычно на основе поступающей информации они выдвигают несколько альтернатив для намерений соперника и его целевых состояний и пытаются найти для них аргументы за и против, отвергая альтернативы, несовместимые с получаемой информацией. В целом, этот подход экспертов, работает, и если его перевести на язык алгоритмов и программ, то он должен работать и в автоматическом варианте. Важно, что эксперты знают структуры признаков и паттерны поведения, которые могут существенно облегчить задачу *инжиниринга признаков* для машинного обучения.

г) *Математические модели группового поведения объектов* A_1, \dots, A_n в зависимости от намерений системы A . Отметим, что эти модели строят на основе экспертных знаний, но знаний экспертов более широкого круга, чем тот, который решает задачу предсказания намерений в конкретном приложении. Они включают знания математиков, социологов, наряду с предметными знаниями. Эти модели играют большую роль в проблемах управления и ИИ. Особо важную роль эти модели играют в задачах генерации обучающей выборки. Они способны интегрировать различные источники знаний. В ИИ такие модели получили специальное название – *цифровой двойник* системы.

6. ПРЕДСКАЗАНИЕ НАМЕРЕНИЙ

Рассмотрим задачу предсказания намерений конкурирующей системы A в двух её постановках, сформулированных в разд. 4, и покажем важную роль её цифрового двойника в обоих вариантах её постановки.

6.1. ЗАДАЧА 1: ПРЕДСКАЗАНИЕ ЦЕЛЕВОГО СОСТОЯНИЯ КОНКУРЕНТА

В разд. 5 проведён анализ источников экспертных знаний. Из него следует, что эксперты могут предложить варианты структур признаков и паттернов поведения, по значениям которых в текущем цикле Бойда система B сможет предсказать классы целевых состояний системы A . Используя эти решения, система B сможет предсказывать её намерения, их потенциальные угрозы и на этой основе формировать решающие правила по выбору мер противодействия. Эту схему можно использовать для алгоритма предсказания намерений соперника. Опишем содержательно его общую идею и выясним роль математической модели системы A в этой задаче, названной ранее задачей 1.

В этой задаче (разд. 4) полагается, что объекты A_1, \dots, A_n действуют в соответствии со сценарием, который неизвестен системе B . Эксперты тоже не знают этого сценария. На рис. 3 он представлен графом (слева), задающим частичный порядок на множестве действий, выполняемых объектами A_1, \dots, A_n .

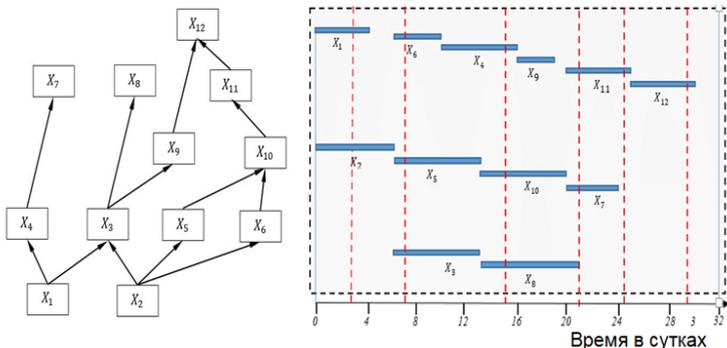


Рис. 3. Сценарий реализации намерения системы A , в форме графа (слева) и диаграммы Ганта (справа). Красными линиями отмечены моменты времени, к которым относятся сообщения о состоянии и поведении системы A , получаемые системой B

На временной шкале тот же сценарий представлен диаграммой Ганта (справа). Пусть сообщения, получаемые системой B , относятся к моментам времени, отмеченным на диаграмме Ганта вертикальными красными линиями. В эти моменты входная информация может содержать различные сведения об объектах A_1, \dots, A_n и о выполняемых ими действиях, в том числе, выполняемых объектами параллельно. Например, из рис. 3 можно получить, что группы объектов $\{A_1 A_2\}$, $\{A_3, A_5, A_6\}$, $\{A_4, A_8, A_{10}\}$, $\{A_7 A_{11}\}$ выполняют параллельно свои действия X_1 и X_2 , X_3, X_5 и X_6 , X_4, X_8 и X_{10} , X_7 и X_{11} , соответственно. Эксперты на основе опыта могут также установить и последовательности действий, которые повторяются в разных реализациях для некоторого намерения. Например, ими могут наблюдаться паттерны действий $\langle X_1 X_3 \rangle$, $\langle X_1 X_3 X_8 \rangle$, $\langle X_5 X_{10} X_{11} \rangle$, $\langle X_6 X_{10} \rangle$. На графе сценария они представлены явно, и хотя этот граф системе B неизвестен, эти паттерны могут быть найдены по получаемым данным.

Однако, для их обнаружения нужно иметь представительную выборку данных, в которой отражено поведение объектов системы A . Трудно надеяться, что имеющаяся выборка примеров удовлетворяет этим требованиям, т.к. она имеет обычно малый объём и в ней поведение объектов системы A может быть не представлено нужным образом.

Ситуацию можно кардинально *изменить к лучшему*, если использовать *модель группового (сценарного) поведения* объектов системы A , зависящую от её намерений. Её можно будет использовать как генератор размеченной выборки данных для каждого намерения системы A . Каждый пример такой выборки будет представлять собой

множество *траекторий совместного поведения* объектов A_1, \dots, A_n с их привязкой к общему времени. Эта выборка будет содержать информацию о групповом поведении объектов системы A и о временных закономерностях их поведения, что необходимо для решения задач предсказания. Поэтому её можно будет использовать как выборку для машинного обучения классификатора предсказываемых целевых состояний системы A с распределением вероятностей на множестве их альтернатив. Этот классификатор позволит для каждого входа (множества наблюдаемых траекторий поведения объектов A_1, \dots, A_n) построить распределение вероятностей на множестве альтернатив намерений и оценить, представляет ли наблюдаемое поведение системы A угрозу для системы B .

Заметим, что программный прототип инструмента, который решает описанную задачу генерации выборки, находится в разработке.

6.2. ЗАДАЧА 2: РАСПОЗНАВАНИЕ КЛАССА СЦЕНАРИЕВ ПОВЕДЕНИЯ

В постановке задачи 1 (разд. 6.1) сценарий группового поведения объектов A_1, \dots, A_n в зависимости от намерений системы A не является предметом классификации для системы B , и она «не знает» что-либо об этих сценариях. В отличие от этого, в постановке задачи 2 (разд. 4) полагается, что система B «знает» множество классов сценариев группового поведения объектов A_1, \dots, A_n в зависимости от намерения и использует эти знания при решении задачи классификации сценариев.

Идея алгоритма распознавания класса сценариев поведения системы A в реальном времени состоит в том, что задачи 1 и 2 решаются параллельно, а решение формируется на основе слияния их решений.

Предварительно с помощью математической модели группового поведения объектов системы A генерируются обучающие выборки траекторий совместного поведения её объектов A_1, \dots, A_n , примеры которых размечаются именами классов целевых состояний (для задачи 1) и именами классов сценариев (для задачи 2). После обучения соответствующих классификаторов система B будет готова к работе.

Общая структура системы, включающая процедуры машинного обучения обоих классификаторов и предсказание намерений системы A , использует модель Бойда (рис. 2), в которую добавлен контур с цифровым двойником группового поведения системы A (рис. 4). Этот контур решает две задачи. Первая – генерация обучающих выборок, пополняющих имеющееся множество примеров, если таковое имеется. Машинное обучение проводится офлайн на этапе подготовки системы

предсказания намерений к работе, но может использоваться и для модификации классификаторов по мере накопления новых примеров. Результаты работы контура машинного обучения используются в блоке «Оценка и прогнозирование» цикла Бойда.

Вторая задача – это прогнозирование траекторий поведения объектов A_1, \dots, A_n в задаче предсказания намерений. Соответствующий итеративный алгоритм кратко поясняется ниже.

Данные для s -го цикла Бойда – результаты $(s - 1)$ -го цикла:

1. $\tilde{X}(s - 1)$ – множество альтернатив классов целевых состояний;
2. $\tilde{I}(s - 1) = \{I_{i_1}, \dots, I_{i_k}\}$ – множества альтернатив намерений системы A ;
3. $P(\tilde{I}(s - 1)) = \{p(I_{i_1}), \dots, p(I_{i_k})\}$ – распределение вероятностей альтернатив на множестве намерений;
4. $\tilde{S}(s - 1) = \{\tilde{S}(I_p, I_q)\}_{p,q}$ – текущее множество альтернатив сценария;
5. $\tilde{X}^{(i)}(s) = \{\tilde{X}_1^{(i)}(s), \dots, \tilde{X}_n^{(i)}(s)\}, i \in \{1, \dots, i_k\}$ – состояния объектов A_1, \dots, A_n в s -ом цикле Бойда, предсказанные в $(s - 1)$ -ом цикле с помощью цифрового двойника для всех $S \in \tilde{S}(s - 1)$.

Алгоритм на текущем шаге s (см. рис. 4):

1. Получает факты из входных сообщений, Источник – блок «Восприятие», получатель – блок «Оценка и прогноз» (рис. 4).

2. Обновляет прогноз $(s - 1)$ -ого цикла для состояний объектов $\tilde{X}^{(i)}(s)$, используя входные факты, результат – $\tilde{X}^{(i)}(s)$.

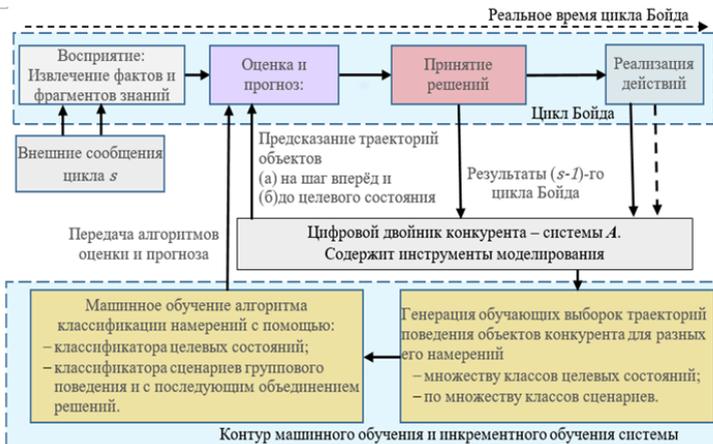


Рис. 4. Цикл Бойда с контуром машинного обучения

3. Прогнозирует целевые состояния для текущего множества альтернатив сценариев, стартуя из состояния $\tilde{X}^{(i)}(s)$.

4. Выполняет классификацию целевых состояний и сценариев с помощью алгоритмов, полученных методами машинного обучения, и строит новое распределение $P(\tilde{I}(s)) = \{p(I_{i_1}), \dots, p(I_{i_k})\}$ для s -го цикла Бойда.

5. Обновляет s -ое множество альтернатив для намерений, применяя пороговый алгоритм для вероятностей $P(\tilde{I}(s)) = \{p(I_{i_1}), \dots, p(I_{i_k})\}$, и обновляет множества $\tilde{X}(s-1)$ и $\tilde{S}(s-1)$.

6. Выполняет с помощью цифрового двойника прогноз начального условия $\tilde{X}^{(i)}(s+1) = \{\tilde{X}_1^{(i)}(s+1), \dots, \tilde{X}_n^{(i)}(s+1)\}$, $i \in \{i_1, \dots, i_k\}$, стартуя из состояний $\tilde{X}^{(i)}(s) = \{\tilde{X}_1^{(i)}(s), \dots, \tilde{X}_n^{(i)}(s)\}$, $i \in \{i_1, \dots, i_k\}$, для всех текущих альтернатив классов сценариев.

8. $s = s + 1$ Переход к п. 1, т.е. к выполнению следующего цикла Бойда.

7. ЗАКЛЮЧЕНИЕ

Основные проблемы задачи предсказания намерений конкурента обусловлены дефицитом онлайн-информации о его текущем поведении, а также дефицитом знаний о его поведении. Другие факторы сложности – это слабая зрелость методов машинного обучения в поведенческих моделях ИИ и отсутствие представительных обучающих данных.

Использование цифрового двойника группового поведения конкурента для генерации и обогащения выборки, а также для прогнозирования траекторий объектов конкурента позволяют свести задачу предсказания его намерений к решению двух задач машинного обучения и классификации с последующим объединением их решений.

Литература

1. Саати Т.Л. Математические модели конфликтных ситуаций. – М.: Сов. радио, 1977. – 300 с. (Saaty T.L. Mathematical models of arms control and disarmament: application of mathematical structures in politics, New York: Wiley, 190 pp.).
2. OODA. – https://en.wikipedia.org/wiki/OODA_loop (дата обращения: 30.03.2022).
2. Longbing Cao. Data Science Thinking. – Springer, 2018. – 360 p.

ПОСТРОЕНИЕ СЛУЧАЙНЫХ ЛЕСОВ ДЕРЕВЬЕВ РЕШЕНИЙ С ПРИМЕНЕНИЕМ ЯДЕРНЫХ РАЗДЕЛИТЕЛЕЙ

*Девяткин Д.А., научный сотрудник.
ФИЦ ИУ РАН
e-mail: devyatkin@isa.ru*

1. ВВЕДЕНИЕ

Деревья решений [1], их композиции [2] и случайные леса [3] применяются во многих системах интеллектуальной обработки данных. Деревья решений состоят из *узлов*, каждый из которых делит данные на два подмножества в соответствии с *критерием построения разделителя*, таким как неоднородность Джини (Gini impurity) или информационная энтропия. Для поиска параметров, минимизирующих этот критерий решается задача дискретной оптимизации. В деревьях решений с одномерными разделителями оптимизация тривиальна, так как имеется возможность протестировать все возможные пороговые значения для каждого признака обрабатываемых данных. Однако, подобный тип деревьев решений имеет ограниченную применимость при анализе многомерных разреженных данных, которые часто используются во многих задачах обработки текста и изображений.

Один из подходов к решению этой проблемы является обучение деревьев с более сложными разделителями, например, применение гиперплоскостей для разделения данных в узлах дерева (линейных разделяющих критериев). К сожалению, подобные деревья решений имеют низкую обобщающую способность, поэтому они применимы только в составе рандомизированных композиций, построенных методами бэггинга, со-бэггинга или случайного леса. Кроме того, большинство подходов к построению таких деревьев имеют низкую вычислительную эффективность и большое количество гиперпараметров. Тем не менее, при использовании таких композиций также могут наблюдаться эффекты, связанные с переобучением, поэтому необходимо также использовать методы регуляризации, позволяющие найти баланс между сложностью получаемых моделей и их обобщающей способностью. В работе предложен алгоритм для построения случайных лесов деревьев решений с ядерными разделителями (линейными, по-

линомиальными, Гауссовскими). Построение этих разделителей осуществляется путем оптимизации функции потерь, подобной применяемой в методе опорных векторов.

Основные результаты работы состоят в следующем.

1. Предложен метод обучения ядерных разделителей путем оптимизации SVM-подобной функции потерь; следовательно, при их построении могут быть использованы вычислительно-эффективные методы оптимизации.
2. Эмпирически показано, что на многих наборах данных применение нелинейных ядер повышает точность и полноту классификации.
3. Эмпирически показано, что регуляризация построения лесов деревьев решений с ядерными разделителями может значительно улучшить обобщающую способность лесов деревьев решений.

2. МЕТОДЫ ПОСТРОЕНИЯ ДЕРЕВЬЕВ РЕШЕНИЙ С ЛИНЕЙНЫМИ РАЗДЕЛИТЕЛЯМИ

Деревья решений и случайные леса успешно применяются при решении задач классификации и регрессии благодаря вычислительной эффективности алгоритмов их обучения. Стандартным подходом к их построению остается жадный рекурсивный алгоритм. В этом алгоритме в каждом узле дерева выполняется оптимизация, которая делит обучающие данные на два подмножества в соответствии с критерием построения разделителя, таким как критерий Джини в CART [4] или информационная энтропия в C4.5 [5]. Далее, этот шаг рекурсивно повторяется для полученных помножеств обучающих данных. При использовании одномерного разделителя значение некоторого признака сравнивается с константой, что эквивалентно разбиению обучающего набора данных гиперплоскостью, параллельной одной из осей. Параметры этой гиперплоскости могут быть найдены с помощью простой комбинации алгоритмов линейного поиска и биекции. В случае применения линейных разделителей необходимо задать функцию, явно связывающую параметры разделяющей гиперплоскости и значения критерия. Оптимизация такой функции зачастую является нетривиальной задачей.

В статье [6] для поиска разделяющих гиперплоскостей используется линейная регрессия или линейный дискриминантный анализ. Однако этот метод позволяет строить деревья только для задач бинарной классификации. В работе [7] предложены CO2-деревья с линейными разделителями, при построении которых необходимо оптимизировать выпукло-вогнутую функцию потерь, а также предложен подход

регуляризации при обучении таких деревьев. Задача выпукло-вогнутой оптимизации может быть довольно эффективно решена с помощью градиентного метода, предложенного в [8]. Хотя эта функция является верхней границей эмпирических потерь, «гладкость» этой функции и точность решения сильно зависят от гиперпараметров, что ограничивает применимость метода. В статье [9] представлены составные деревья решений (Composite Trees), листовые классификаторы в которых выбираются из набора, состоящего из подсемейств различной сложности. Предложен алгоритм обучения случайных составных деревьев (Random Composite Forest, RCF), основанный на генерации набора случайных составных деревьев. Однако, в промежуточных узлах таких деревьев по-прежнему используются одномерные разделители.

Перспективным направлением исследований также является сквозное обучение лесов деревьев решений с линейными разделителями. В статье [10] представлен подход к обучению деревьев с применением EM-алгоритма. Деревья представляют собой вероятностные модели во время обучения, но полностью детерминированы во время классификации. Экспериментальные исследования подхода на наборах размеченных изображений показывают, что он позволяет получать более сложные разбиения, чем обычные линейные разделители. Однако, EM-алгоритм максимизации ожидания медленно сходится на больших наборах объектов, описываемых многими признаками. Аналогично, в работе [12] предлагается использовать метод обратного распространения ошибки, который обычно используется для обучения многослойных нейронных сетей, для обучения деревьев решений с линейными разделителями и фиксированной структурой. Однако использование такой структуры может привести к обучению излишне сложных классификаторов с большим количеством параметров и, как следствие, к ухудшению обобщающей способности. В работе [13] предложены взвешенные деревья решений (WODT). В этих деревьях всем обучающим примерам во всех узлах назначаются веса, а затем проводится обучение линейных разделителей путем оптимизации функции взвешенной информационной энтропии. Стоит отметить, что функция потерь в WODT невыпуклая; следовательно, нахождение глобального оптимума при построении разделителей не гарантируется. В то же время экспериментальные результаты показывают высокие оценки точности и полноты метода на многих наборах данных.

В настоящей работе предложен метод обучения и регуляризации случайного леса деревьев решений с ядерными разделителями (с линейными, полиномиальными, гауссовскими и другими ядрами). В предложенном методе аналогично WODT [13] примерам из обучаю-

щей выборки назначаются различные веса. Однако в предложенном методе обучение разделителей сводится к решению SVM-подобной задачи выпуклой оптимизации с ограничениями-неравенствами, для которой ранее было предложено большое количество вычислительно-эффективных методов. При этом вместо поиска глобального оптимума критерия построения разделителей производится поиск квазиоптимального решения, что позволяет сохранить высокую скорость обучения.

3. МЕТОД ПОСТРОЕНИЯ ДЕРЕВЬЕВ РЕШЕНИЙ С ЯДЕРНЫМИ РАЗДЕЛИТЕЛЯМИ

Для построения деревьев решений используется стандартный жадный рекурсивный алгоритм. На каждом шаге строится узел дерева, который разделяет обучающие данные, затем эта процедура рекурсивно повторяется для «левого» и «правого» подмножеств обучающих данных, пока не будет достигнута заданная высота дерева. Для вычислительно-эффективного построения разделителей задается непрерывная гладкая функция потерь, которая отражает зависимость между параметрами разделителя и значением критерия его построения.

В основе предложенного метода построения ядерных разделителей лежит предположение о том, что ошибка классификации по поддеревьям, приводящая к значительному изменению критерия построения разделителя, должна приводить к большему приросту функции потерь, чем ошибка, повлекшая небольшое изменение этого критерия. Один из методов, позволяющих реализовать такой подход при обучении метода опорных векторов, называется "slack re-scaling" [14]. Преимущество этого подхода в том, что получаемые результаты не зависят от масштабирования признаков объектов.

Рассмотрим построение разделителя как задачу бинарной классификации, в которой при обучении примеры с метками определенных классов распределяются по поддеревьям и $S = \{-1, +1\}$ – метки этих поддеревьев. Пусть установлено целевое распределение классов в поддеревьях $c_S : Y \rightarrow S$, которое минимизирует критерий построения разделителя. Некоторые критерии могут разделять примеры одного класса по разным поддеревьям, но в предложенном методе эта особенность игнорируется в угоду скорости обучения. Пусть минимальное значение критерия построения разделителя соответствует эмпирическим оценкам вероятностей отнесения выборки данных к поддеревьям P_L и P_R и вероятностей классов поддеревьев этой выборки p_L и p_R . Зададим:

$$s_{best} = \arg \min P_L g(p_L) + P_R g(p_R), \quad (1)$$

где s_{best} – теоретически минимальное значение критерия построения разделителя (в случае, если данные могут быть разделены без ошибок согласно распределению c_s), $g(p)$ – неоднородность Джини или информационная энтропия.

Пусть w_{best} – параметры (возможно, недостижимые) разделяющей гиперплоскости, которая позволяет получить значение критерия s_{best} . Обучение разделителя на наборе данных из m обучающих примеров производится путем решения следующей задачи оптимизации:

$$w^*, \xi^* = \arg \min_{w, \xi} \frac{1}{2} \|w\|^2 + \frac{C}{m} \sum_{i=1}^m \xi_i. \quad (2)$$

При ограничениях:

$$\forall i, w^T x_{si} \geq 1 - \frac{\xi_i}{L(s_i, s)},$$

где w^* – параметры разделяющей гиперплоскости, ξ^* – дополнительные переменные, C – параметр регуляризации, а $L(s_i, s)$ отражает прирост критерия построения разделителей в случае отнесения примера i к некорректному поддереву s вместо s_i .

Если применить к этой задаче условия Каруша-Куна-Таккера, то получится двойственная задача оптимизации:

$$a^* = \arg \max a - \frac{1}{2} \sum_{i, s \neq s_i} \sum_{j, \tilde{s} \neq s_i} a_{is} a_{j\tilde{s}} K(x_i, x_j) + \sum_{i, s \neq s_i} a_{is}. \quad (3)$$

При ограничениях:

$$\sum_{s \neq s_i} \frac{a_{is}}{L(s_i, s)} \leq \frac{C}{m},$$

где a_{ij} – вес примера i из обучающей выборки (отличный от нуля для опорных векторов), а $K(x_i, x_j) : X^2 \rightarrow \mathbb{R}$ – положительно определенное ядро. В отличие от классификации структур [14], эта задача эф-

фактивно решается в явном виде, поскольку классов всего два (два поддерева). Гиперпараметр регуляризации C должен быть подобран эмпирически для каждого набора данных.

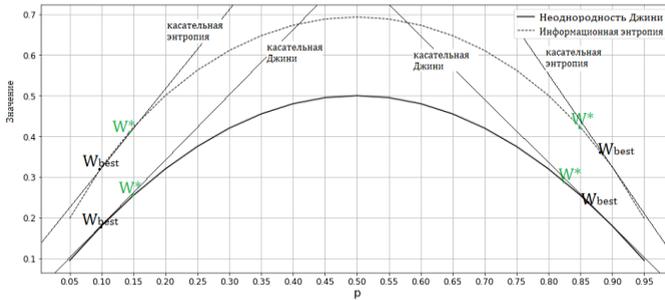


Рис. 1. Критерии построения разделителей и эмпирический риск $\hat{R}(w)$. $\hat{R}(w)$ задает касательную к графику критерия построения разделителя в точке наилучшего разбиения w_{best} . Точке w^* соответствуют параметры разбиения, достижимые на практике на анализируемых данных

Предположим, что полученный разделитель относит некоторый элемент к поддереву s^* вместо s . Изменение оценок вероятностей (по сравнению с оценками для лучшего разбиения s_{best}) можно обозначить: $P_L^*, P_R^*, p_L^*, p_R^*$ и прирост критерия равен:

$$L(s^*, s) = P_L^* g(p_L^*) + P_R^* g(p_R^*) - P_L g(p_L) - P_R g(p_R). \quad (4)$$

Зададим $\hat{R}(w) = \frac{1}{m} \sum_{i=1}^m L(s_i, x_i^T w)$ — эмпирический риск на обучающей выборке. В работах [14, 15] приводится доказательство того, что решение задач (2) или (3) приводит к оптимизации эмпирического риска $\hat{R}(w)$. Стоит отметить, что эмпирический риск $\hat{R}(w)$ не является тождественным критерию построения разделителя. Эмпирический риск может быть переформулирован как $\hat{R}(p_{err}) = \sum_{y \in Y} p_{err}(y) L(c_s(y), -c_s(y))$, где $p_{err}(y)$ возвращает долю некорректно распределенных примеров класса y (тех, что по итогам

классификации распределяются по поддеревьям не в соответствии с S_{best}). Все рассматриваемые критерии построения разделителей являются выпуклыми функциями от вероятностей классов в поддеревьях. Можно однозначно преобразовать функцию потерь от классов $L(c_S(y), -c_S(y))$ в функцию потерь от изменений вероятностей классов по сравнению с лучшим разбиением S_{best} (см. выражение (4)). Следовательно, функция $L(s_i, x_i^T w)$ возвращает значение частной производной критерия построения разделителей в точке S_{best} , т.е. график \hat{R} является касательной к графику критерия построения разделителя в точке S_{best} , которой соответствуют параметры с параметрами разделяющей гиперплоскости w_{best} . При оптимизации (2) производится спуск по касательной целевой функции настолько близко к w_{best} , насколько это позволяют распределение классифицируемых объектов в признаковом пространстве и выбранная функция ядра (см. рис. 1).

Для представленного метода гарантируется, что для заданного распределения классов по поддеревьям разделяющая гиперплоскость будет построена так, чтобы минимизировать $\hat{R}(w)$. Однако, для других распределений $Y \rightarrow S$ значение $\hat{R}(w^*)$ может оказаться меньше, поскольку полученные параметры w^* будут ближе к w_{best} . Теоретически для оптимизации критерия необходимо рассмотреть все пары $Y \rightarrow S$. Однако, было решено отказаться от достижения глобального максимума, ввиду того, что сам алгоритм построения деревьев является жадным. В процессе построения разделителей можно выбирать теоретически лучшее распределение классов по поддеревьям, либо случайным образом выбрать несколько хороших распределений, решить задачу (2) или (3) для всех из них и выбрать лучшее. Такой подход позволяет дополнительно рандомизировать структуру деревьев, что положительно сказывается на обобщающей способности случайных лесов, формируемых из них. Представленный подход был протестирован на нескольких наборах данных (MNIST, CIFAR, USPS), при этом не обнаружено различий в точности классификации с помощью полученных композиций деревьев решений по сравнению с проверкой всех возможных распределений $Y \rightarrow S$.

4. РЕГУЛЯРИЗАЦИЯ СЛУЧАЙНЫХ ЛЕСОВ ДЕРЕВЬЕВ РЕШЕНИЙ

Большая часть подходов к регуляризации случайных лесов деревьев решений состоит в искусственном внесении случайных различий между деревьями композиции. В этих подходах предлагается, например, случайным образом выбирать критерий построения разделителей или задавать долю признаков, которые нужно протестировать, чтобы построить узел дерева. Вместе с тем, подобный подход может отрицательно повлиять на смещение (bias) композиции. Несколько иной подход представлен в работе [16], в которой предлагается отбросить эмпирические оценки вероятностей классов, хранящиеся в листьях деревьев предварительно обученного случайного леса, и переопределить их. Для получения новых значений оптимизируется глобальная функция потерь, сходная с применяемой в методе опорных векторов. В ходе экспериментальных исследований этот подход применялся для регуляризации случайных лесов деревьев решений с ядерными разделителями. Оптимизация глобальной функции потерь может привести к переобучению композиции. Для предотвращения этого эффекта используется метод прунинга, который состоит в соединении двух соседних листьев деревьев, в случае, когда норма векторов этих листьев близка к нулю.

5. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНОГО ИССЛЕДОВАНИЯ МЕТОДОВ ПОСТРОЕНИЯ ЛЕСОВ ДЕРЕВЬЕВ РЕШЕНИЙ

Экспериментальные исследования разработанного метода проводились на четырех размеченных наборах данных из коллекции UCI; а именно, SatImage, USPS, Letter, MNIST [17]. Эти наборы данных применяются для обучения и оценки методов распознавания изображений: наборы MNIST и USPS содержат рукописные изображения цифр, а набор Letter содержит латинские буквы. Еще один набор данных, на котором проводились исследования – CIFAR-10, также связан с распознаванием изображений [18]. В этом наборе содержатся изображения разрешением 32 на 32 пикселя с восемью уровнями серого. Набор включает 10 классов (самолет, лошадь, птица и т. д.). Исследовались следующие методы: случайный лес деревьев решений (Random Forest), случайный лес деревьев, построенных методом CO2 (CO2 Forest), случайный лес деревьев WODT, случайный лес деревьев решений с ядерными разделителями (Kernel Forest).

Для подбора гиперпараметров исследуемых методов использовалась статистическая процедура перекрестного скользящего контроля. Для оценки качества классификации (точность, ассигасу) использовалась отложенная выборка. Для лесов деревьев решений с ядерными разделителями выполнялся подбор следующих гиперпараметров: количество деревьев $T=\{30,100,300\}$, параметр регуляризации при построении разделителей $C=\{100, 1000, 3000, 5000\}$, максимальная глубина дерева $n=\{3,4,5,6,7\}$, доля признаков, которые необходимо учитывать в каждом узле $f=\{0.08,0.1,0.2,0.3,0.4,0.5\}$, коэффициенты регуляризации (до 0,9), а также параметры ядра ($\gamma=\{10,100\}$ для Гауссовского ядра, $\text{degree}=3$ для полиномиального ядра).

В табл. 1 представлены полученные результаты (точность классификации на отложенной выборке) экспериментального исследования методов построения случайных лесов деревьев решений различных видов. Наилучшие результаты для использованных наборов данных распознавания были получены с применением разделителей с Гауссовским ядром.

Таблица 1. Результаты экспериментальных исследований (точность) методов построения случайных лесов с деревьями различных видов

Набор данных/ Метод	MNIST	USPS	Letter	SatImage	Cifar-10
Random Forest	0.972	0.936	0.963	0.911	0.501
CO2 Forest	0.981	0.945	0.982	0.911	-
WODT	0.943	0.905	0.879	0.876	-
Kernel Forest	0.991	0.946	0.975	0.918	0.581
Kernel Forest + регуляризация	0.992	0.952	0.981	0.920	0.590
Kernel Forest + регуляризация + прунинг	0.992	0.954	0.982	0.921	0.591

В табл. 1 также показаны результаты экспериментальных исследований регуляризации случайных лесов деревьев решений с ядерными разделителями. Следует отметить, что регуляризация по методу [16] позволяет значительно улучшить результаты классификации на этих наборах данных (до 0,6%), повышая применимость деревьев со сложными ядрами. В то же время применение прунинга в дополнение к регуляризации практически не привело к улучшению результатов классификации.

6. ЗАКЛЮЧЕНИЕ

Применение рандомизированных композиций деревьев решений с нелинейными ядерными разделителями позволяет достигать большей точности при решении многих задач классификации, особенно связанных с распознаванием изображений. Однако предложенный подход к построению таких композиций требует дальнейшего улучшения, направленного на повышение скорости обучения. Наилучшие по точности результаты классификации обычно достигаются при гиперпараметре регуляризации $C > 1000$. В таком случае функция потерь становится негладкой, что усложняет оптимизацию. Поэтому необходимы дальнейшие исследования, направленные на создание быстрых алгоритмов построения опорных векторов при обучении узлов деревьев решений.

В результате экспериментов также показано, что метод регуляризации ансамблей деревьев решений [16] позволяет повысить обобщающую способность случайных лесов деревьев решений с ядерными разделителями.

В ходе анализа текстов, изображений, структурированных данных зачастую необходимо решать задачу моделирования дискретных процессов. Наилучшей точности решения этой задачи теоретически можно было бы достигнуть при применении дискретных моделей. Дальнейшие исследования будут направлены на использовании случайных лесов деревьев решений с ядерными разделителями в качестве компонентов более сложных моделей, которые позволяют выявлять сложные неявные зависимости между признаками и классифицируемыми объектами.

Литература

1. Breiman L. Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J.. Classification and regression trees. Belmont, CA: Wadsworth // International Group. – 1984. – Vol. 432. – P. 151-166.
2. Chen T., Guestrin C. Xgboost: A scalable tree boosting system //Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. – 2016. – P. 785-794.
3. Breiman L. Random forests //Machine learning. – 2001. – Vol. 45, No. 1. – P. 5-32.
4. Breiman L. Some properties of splitting criteria // Machine Learning. – 1996. – Vol. 24, No. 1. – P. 41-47.
5. Quinlan J. R. Induction of decision trees // Machine learning. – 1986. – Vol. 1. – No. 1. – P. 81-106.
6. Menze, B. H., Kelm, B. M., Splitthoff, D. N., Koethe, U., & Hamprecht, F.A. On oblique random forests //Joint European Conference on Machine Learning and Knowledge Discovery in Databases. – Springer, Berlin, Heidelberg, 2011. – P. 453-469.

7. Norouzi, M., Collins, M. D., Fleet, D. J., Kohli, P. Co2 forest: Improved random forest by continuous optimization of oblique splits // arXiv preprint arXiv:1506.06155. – 2015.
8. Yuille A.L., Rangarajan A. The concave-convex procedure // Neural computation. – 2003. – Vol. 15, No. 4. – P. 915-936.
9. DeSalvo G., Mohri M. Random composite forests // Proceedings of the AAAI Conference on Artificial Intelligence. – 2016. – Vol. 30, No. 1.
10. Bartlett P.L., Mendelson S. Rademacher and Gaussian complexities: Risk bounds and structural results // Journal of Machine Learning Research. – 2002. – Vol. 3, No. Nov. – P. 463-482.
11. Hehn T.M., Kooij J.F.P., Hamprecht F.A. End-to-end learning of decision trees and forests //International Journal of Computer Vision. – 2020. – Vol. 128, No. 4. – P. 997-1011.
12. Irsoy O., Alpaydin E. Autoencoder trees // Asian conference on machine learning. – PMLR, 2016. – P. 378-390.
13. Yang B. B., Shen S. Q., Gao W. Weighted oblique decision trees // Proceedings of the AAAI Conference on Artificial Intelligence. – 2019. – Vol. 33, No. 01. – P. 5621-5627.
14. Tschantaridis, I., Joachims, T., Hofmann, T., Altun, Y., Singer, Y. Large margin methods for structured and interdependent output variables //Journal of machine learning research. – 2005. – Vol. 6. – No. 9.
15. Taskar B., Guestrin C., Koller D. Max-margin Markov networks //Advances in neural information processing systems. – 2003. – Vol. 16.
16. Ren S. Ren, S., Cao, X., Wei, Y., Sun, J. Global refinement of random forest //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. – 2015. – P. 723-730.
17. Murphy P. M., Aha D. W. UCI Repository of machine learning databases, Irvine, CA: University of California, Department of Information and Computer Science. – 1994. – URL: <https://www.ics.uci.edu/mllearn/MLRepository.html>.
18. Krizhevsky A. Learning multiple layers of features from tiny images. Technical report, University of Toronto. – 2009.

СРАВНЕНИЕ ПОДХОДОВ К ГЕНЕРАЦИИ ТЕКСТОВЫХ ОПИСАНИЙ

Ипатов М.А., студент

НИУ «МЭИ»

e-mail: mipatov0pls@ya.ru

Мохов А.С., к.т.н.

НИУ «МЭИ»

e-mail: ASMokhov@mail.ru

1. ВВЕДЕНИЕ

Задача генерации текстов находит применение во многих областях, например – системы машинного перевода, перефразирование текста, исправление опечаток, автоматическое написание рекламных объявлений, автоматическое создание описаний товаров для интернет-магазинов, построение диалоговых систем.

В данной работе будет рассматриваться задача генерации текстовых описаний для товаров интернет-магазина. Обычно, для создания описаний товаров прибегают к помощи специалистов-копирайтеров, которые на основе характеристик конкретной модели составляют такие описания, однако с увеличением количества товаров данная задача становится все более трудозатратой и дорогостоящей.

Применение специальных предиктивных моделей позволит генерировать подобные описания в автоматическом режиме на основе только лишь характеристик каждого товара и позволит ускорить и удешевить данную задачу.

2. ЯЗЫКОВОЕ МОДЕЛИРОВАНИЕ

В основе обработки естественного языка (Natural Language Processing, NLP) лежит понятие языковой модели.

Языковая модель (language model) решает задачу моделирования вероятностного распределения последовательности токенов – букв, слогов или целых слов естественного языка, то есть рассчитывает вероятность $P(w_1, w_2, \dots, w_k)$, где $\{w_1, w_2, \dots, w_k\}$ – последовательность токенов. Смежной задачей является продолжение последовательности – предсказание следующего токена на основе извест-

ных предыдущих: подсчет $P(w_k | w_1, w_2, \dots, w_{k-1})$. Формула (1) отражает связь между этими задачами через определение условной вероятности.

$$P(w_1, w_2, \dots, w_k) = P(w_1) * P(w_2 | w_1) * \dots * P(w_N | w_1, w_2, \dots, w_{k-1}). \quad (1)$$

Для построения языковых моделей существует два подхода: частотный и нейросетевой.

Простым способом вычисления условной вероятности $P(w_k | w_1 \dots, w_{k-1})$ является формула (2).

$$P(w_k | w_1 \dots, w_{k-1}) = \frac{N(w_1 \dots w_{k-1} w_k)}{N(w_1 \dots w_{k-1})}. \quad (2)$$

Здесь $N(w_1 \dots, w_k)$ – это количество раз, когда последовательность $\{w_1 \dots, w_k\}$ встретилась в тексте. Но на практике этот способ неработоспособен, потому что многие сочетания слов $\{w_1 \dots, w_k\}$ никогда не встречались, следовательно обнуляется вероятность (2) такой последовательности.

3. НЕЙРОСЕТЕВОЙ ПОДХОД К ГЕНЕРАЦИИ ТЕКСТОВ

Задача нейронной сети – получить векторное представление для предыдущего контекста и, используя это представление, предсказать распределение вероятностей для следующего токена, т.е. $P(w_k | w_{k-N+1} \dots, w_{k-1})$. Одной из наиболее современных архитектур нейронных сетей является архитектура «Трансформер» [1].

Нейронная сеть «Трансформер» состоит из двух частей – кодировщика (encoder) и декодера (decoder). Кодировщик осуществляет кодирование входной последовательности в векторное представление, а декодер с учетом этих данных авторегрессионно предсказывает токены результирующей последовательности. В оригинальной архитектуре описанной в [1] обе части сети состоят из 6 последовательно соединенных блоков.

Существует ряд архитектур, основанных на «Трансформере», которые могут отличаться количеством и составом слоев в кодировщике и декодере, количеством последовательных блоков в этих частях. Существуют также сети, состоящие лишь из одной части: кодировщика или декодера.

Для решения задачи генерации текста подходят архитектуры GPT (Generative Pre-trained Transformer) и T5 (Text-to-Text Transfer Transformer). Рассмотрим их подробнее.

3.1. АРХИТЕКТУРА GPT-3

В 2018 году компания OpenAI представила модель Generative Pre-trained Transformer (GPT) [2]. Данная модель представляет собой модифицированный декодер оригинальной архитектуры «Трансформер». Представленная модель была обучена на крупном корпусе текстовых данных что позволило ей показать выдающиеся результаты при обработке длинных последовательностей, выявлению в них долгосрочных зависимостей, генерации текста и пониманию естественного языка. За моделью GPT последовали GPT-2 [3] и GPT-3 [4]кратно отличающиеся количеством параметров и некоторыми особенностями архитектуры.

Модель GPT-3 была представлена в 2020 году и существует в различных вариациях: количество параметров варьируется от 125 миллионов до 175 миллиардов. Она обучалась на 570 Гб текстовых данных, в который большую часть составляют англоязычные документы.

Позднее в 2020 году Sberbank.AI совместно со SberDevices основываясь на описании оригинальной GPT-3 разработали, предобучили на корпусе русскоязычных текстов (600 Гб) и выложили в открытый доступ ряд моделей ruGPT3 [5].

Таблица 1. Семейство моделей ruGPT3

Наименование модели	Параметров, млн.
ruGPT3Small	125
ruGPT3Medium	356
ruGPT3Large	760
ruGPT3XL	1320

3.2. АРХИТЕКТУРА T5

Text-To-Text Transfer Transformer (T5) – это представленная исследователями из Google [6] seq2seq модель, основанная на архитектуре «Трансформер». Она содержит и кодировщик, и декодер и, таким образом, достаточно близко повторяет оригинальную архитектуру хотя и отличается в деталях некоторых слоев.

Оригинальная модель T5 является мультизадачной и обучена ряду задач, например перевод, ответы на вопросы и классификация.

Данная модель также имеет российский аналог, также представленный лабораторией Sberbank.AI [7]. Модель ruT5 обучена на корпусе текстов, в которых преобладает русский язык (300 Гб). Она решает задачи, аналогичные оригинальным.

Таблица 2. Семейство моделей ruT5

Наименование модели	Параметров, млн.
ruT5-base	222
ruT5-large	737

Стоит отметить, что обучение моделей с миллионами параметров на таких купных наборах данных требует длительного времени (недели) и серьезных вычислительных мощностей. Но уже будучи предобученными, эти модели могут быть дообучены на выполнение смежной задачи и это не потребует таких временных затрат и вычислительных ресурсов. Именно эта возможность настройки (fine-tuning) предобученных моделей используется в данной работе.

4. ИСПОЛЬЗОВАНИЕ ТРАНСФОРМЕРОВ ДЛЯ ГЕНЕРАЦИИ ТЕКСТОВЫХ ОПИСАНИЙ ТОВАРОВ

4.1. ПОСТАНОВКА ЗАДАЧИ

В статье рассматривается задача генерации текстовых описаний ноутбуков по их текстовым характеристикам. Для проведения исследования был собран набор текстовых документов, содержащих наименование, характеристики и описания ноутбуков с сайта www.citilink.ru.

После некоторой обработки, фильтрации и исключения нехарактерных элементов была сформирована выборка.

Таблица 3. Параметры выборки

	средн.	макс.	мин.
Кол-во характеристик у одного товара	30.26	35	22
Длина документа в токенах	511.83	615	368
Длина описания в токенах	194.40	261	115
Всего записей		689	
Всего характеристик		37	

Таким образом, для каждого товара из выборки существуют две части: характеристики товара и его описание.

Таблица 4. Пример из выборки

Характеристики	Описание
Название : DELL Vostro 3401 Диагональ экрана : 14 Разрешение экрана : 1920x1080 Подсветка экрана : есть Поверхность экрана : матовая Процессор : Intel Core i3 1005G1 2 ядра 1.2 ГГц (3.4 ГГц, в режиме Turbo)	Описание : Двухъядерный ноутбук DELL Vostro 3401 адресован представителям бизнеса, может использоваться в дороге, на деловых встречах, в домашних условиях, в офисе. Функционирует на базе процессора Intel Core i3 1005G1 с видеокарткой

<p>Оперативная память : 8 Гб DDR4, 2666 МГц</p> <p>Тип видеокарты : интегрированный</p> <p>Видеокарта : Intel UHD Graphics</p> <p>HDD : 1000 Гб</p> <p>Wi-Fi : ДА, 802.11 a/b/g/n/ac</p> <p>Bluetooth : ДА</p> <p>Кабельная сеть : 10/100/1000 (Gigabit Ethernet) Мбит/с</p> <p>USB 2.0 : 1</p> <p>USB 3.0 : 2</p> <p>HDMI : 1</p> <p>Операционная система : Linux</p> <p>Веб-камера : встроенная</p> <p>Микрофон : есть</p> <p>Разъем наушники/микрофон : комбинированный разъем</p> <p>Акустическая система : стереодинамики</p> <p>Цвет клавиатуры : черный</p> <p>Подсветка клавиш клавиатуры : есть</p> <p>Количество ячеек батареи : 3 cell</p> <p>Энергоемкость батареи : 42 Wh</p> <p>Цвет : черный</p> <p>Размеры : 328.7 x 239.5 x 19.9 мм</p> <p>Вес : 1.64 кг</p> <p>Гарантия : 12 мес.</p> <p>Кард-ридер : есть, поддержка Micro SD</p>	<p>Intel UHD Graphics и оперативной памятью объемом 8 Гб, определяющими высокую скорость выполнения операций и быстрый отклик.</p> <p>Лэптоп прекрасно работает в режиме многозадачности. Ноутбук DELL Vostro 3401 оснащен экраном диагональю 14" с разрешением 1920x1080 пикселей, что дает возможность получать четкую, яркую картинку, независимо от содержания контента. Для долгосрочного хранения файлов на жестком диске предусмотрено 1000 Гб памяти.</p> <p>Встроенным кард-ридером поддерживаются карты памяти microSD.</p> <p>Батарея энергоемкостью 42 Wh обеспечивает многочасовой режим автономной работы. Имеются встроенная веб-камера, микрофон, стереосистема, комбинированный разъем для микрофона и наушников. Есть необходимый набор портов и разъемов для подключения периферии.</p>
--	--

Для проведения дальнейшего исследования выборка разбивается на тренировочную и тестовую составляющие в соотношении 9 к 1.

Сравним модели архитектур GPT-3 и T5 на примере решения поставленной задачи. Были выбраны две модели: **ruGPT3Medium** и **ruT5-large**. Выбор именно этих моделей обусловлен техническими возможностями вычислительных средств.

Для оценки качества обученных моделей будем использовать перплексию – обратную вероятность тестового набора, нормализованная по количеству слов:

$$PP(W) = 2^{H(W)} = 2^{-\frac{1}{N} \log_2 P(w_1, w_2, \dots, w_N)}.$$

Далее рассмотрим особенности обучения моделей и сравним результаты.

4.2. ПРОВЕДЕННЫЕ ИССЛЕДОВАНИЯ

В ходе исследования использовались две предобученные модели типа «трансформер» – **ruGPT3Medium** и **ruT5-large**. Для формирования с помощью данных моделей текстов заданного формата требуется произвести их дообучение на сформированной обучающей выборке.

Исследования проводились в среде Google Colab Pro, которая предоставляет GPU NVIDIA Tesla P100 16 Гб. Для уменьшения объема памяти GPU, требуемого для обучения моделей, использовался оптимизатор NovoGrad [8]. Согласно исследования авторов [8] данный оптимизатор работает не хуже Adam (AdamW) но при этом снижает расход памяти. Обучение каждой модели происходило на протяжении 3-х эпох.

Для дообучения **ruGPT3Medium** обучающая выборка была сформирована в виде набора длинных текстовых предложений состоящих из двух частей: характеристики и описание. Так как архитектура GPT представляет из себя один декодер, модель обучается предсказывать каждый последующий токен только на основе предыдущей последовательности. Таким образом в ходе обучения модель последовательно проходит по каждому предложению из выборки и авторегрессионно предсказывает его продолжение.

Для генерации описания с помощью этой модели необходимо подать ей на вход начало последовательности, т.е. часть с характеристиками. После этого модель продолжит текст так, как она этому обучилась.

Чтобы обучить модель **ruT5-large** элементы выборки необходимо представить в виде двух отдельных строк: характеристики (*source*) и описание (*target*). Строка *source* подается на вход кодировщику, который преобразует ее в некоторое векторное представление, которое, в свою очередь, передается в декодер и влияет на генерацию результирующей последовательности. На вход декодера подается специальный токен начала последовательности и декодер последовательно генерирует текст основываясь на уже сгенерированной части (авторегрессионно) и на информации из кодировщика.

Таким образом, в ходе обучения эта модель учится предсказывать токены лишь в части с описанием, в отличии от моделей GPT.

По итогам обучения получили модели, показывающие следующие значения перплексии:

Таблица 5. Перплексия обученных моделей

Наименование модели	Перплексия
ruGPT3Medium	7.4303
ruT5-large	8.7495

В табл. 6 представлены примеры сгенерированных описаний. Приведенные верно значения характеристик выделены **жирным шрифтом**, неверно – *курсивом*, повторы подчеркнуты.

Таблица 6. Результаты генерации

Реальный пример	
Характеристики	Сгенерированное описание
<p>Название : DELL Vostro 3401 Диагональ экрана : 14 Разрешение экрана : 1920x1080 Подсветка экрана : есть Поверхность экрана : матовая Процессор : Intel Core i3 1005G1 2 ядра 1.2 ГГц (3.4 ГГц, в режиме Turbo) Оперативная память : 8 Гб DDR4, 2666 МГц Тип видеокарты : интегрированный Видеокарта : Intel UHD Graphics HDD : 1000 Гб Wi-Fi : ДА, 802.11 a/b/g/n/ac Bluetooth : ДА Кабельная сеть : 10/100/1000 (Giga- bit Ethernet) Мбит/с USB 2.0 : 1 USB 3.0 : 2 HDMI : 1 Операционная система : Linux Веб-камера : встроенная Микрофон : есть Разъем наушники/микрофон : ком- бинированный разъем Акустическая система : стереоди- намики Цвет клавиатуры : черный Подсветка клавиш клавиатуры : есть Количество ячеек батареи : 3 cell Энергоемкость батареи : 42 Wh Цвет : черный Размеры : 328.7 x 239.5 x 19.9 мм Вес : 1.64 кг Гарантия : 12 мес. Кард-ридер : есть, поддержка Micro SD</p>	<p style="text-align: center;">ruGPT3Medium</p> <p>Ноутбук DELL Vostro 3401 выполнен на базе процессора Intel Core i3 1005G1 с тактовой частотой 1.2 ГГц и оперативной памяти объемом 8 Гб. Он оснащен экраном диагональю 14 дюймов с разрешением 1920x1080 пикселей и поддерживает <i>технологию multi-touch</i> для быстрого доступа к необходимой информации. Для хранения данных предусмотрен накопитель на 1000 Гб. <u>Объем твердотельного накопителя составляет 1000 Гб</u>. Имеется кард-ридер, поддерживающий карты памяти формата micro SD. <u>Предусмотрен слот для карт памяти формата micro SD</u>. <u>Встроенный кард-ридер поддерживает карты памяти формата micro SD емкостью до 32 Гб</u>. Клавиатура оснащена <i>сенсорным дисплеем диагональю 14 дюймов с поддержкой технологии multi-touch</i>. Есть модуль Wi-Fi для подключения внешних устройств.</p> <p style="text-align: center;">ruT5-large</p> <p>Ноутбук DELL Vostro 3401 оснащен экраном диагональю 14 дюйма с разрешением 1920x1080 пикселей, что позволяет ему работать в режиме многозадачности. Встроенный микрофон и стереодинамики способствуют быстрому отклику пользователя на любую информацию. Для подключения периферийных устройств предусмотрены <i>разъемы Wi-Fi и Bluetooth</i>. Ноутбук DELL Vostro 3401 обладает компактными размерами. Его вес составляет 1.64 кг. Он выполнен в прочном корпусе черного цвета. За счет двухъядерного процессора Intel Core i3 1005G1 обеспечивается стабильная работа системы Linux</p>

Искусственный пример	
Характеристики	Сгенерированное описание
<p>Название : Super_NB 2001 Gaming Диагональ экрана : 21 " Разрешение экрана : 1337x228 Поверхность экрана : матовая Тип матрицы : nfc Процессор : CMD processor 7 ядер 16.16 ГГц (46.0 ГГц, в режиме Turbo) Оперативная память : 28 Гб DDR5 Тип видеокарты: интегрированный Видеокарта : qwerty-graphics SSD : 720 Гб Wi-Fi : ДА, 802.11 a/b/g/n/ac Bluetooth : ДА, v5.0 Кабельная сеть : 10/100/1000 (Gigabit Ethernet) Мбит/с USB 2.0 : 13 USB 3.0 : 22 HDMI : 11 Операционная система : CMD-shell Веб-камера : встроенная Микрофон : есть Разъем наушники/микрофон : комбинированный разъем Акустическая система : стереодинамики Цвет клавиатуры : черный Цифровой блок клавиатуры : есть Подсветка клавиатуры : есть Тип батареи : Al-Ion Количество ячеек батареи:36 cell Энергоемкость батареи : 176 Wh Цвет : черный Размеры : 1.23 x 456 x 78.9 мм Вес : 19 кг Гарантия : 322 мес. Материал корпуса : пластик Время работы от батареи : 82ч Кард-ридер : есть WA SD</p>	<p style="text-align: center;">ruGPT3Medium</p> <hr/> <p>Super NB 2001 Gaming - игровой ноутбук с экраном диагональю 21 дюйм и разрешением 1337x228 пикселей. Он оснащен процессором CMD processor 7 ядер 16.16 ГГц, оперативной памятью объемом 28 Гб и видеокартой QWERTY-GRAFICS. Ноутбук имеет твердотельный накопитель на 720 Гб, что позволяет использовать его для хранения информации. Кроме того, имеется кард-ридер, который поддерживает карты <i>памяти формата SD</i>. Для подключения периферии предусмотрены порты USB 2.0 и USB 3.0. Клавиатура ноутбука выполнена из пластика черного цвета. Ноутбук оборудован <i>литий-полимерной</i> батареей, которая обеспечивает длительное время работы без подзарядки. Его вес равен 19 кг. ;</p> <hr/> <p style="text-align: center;">ruT5-large</p> <hr/> <p>Ноутбук Super_NB 2001 Gaming оснащен экраном диагональю 21 дюйма с разрешением 1337x228 пикселей, что обеспечивает высокое качество изображения. Для работы в режиме многозадачности предусмотрен процессор CMD processor 7 ядер (16.16 ГГц), оперативная память объемом 28 Гб и видеокарта qwerty-graphics. <i>Встроенная система Wi-Fi</i> позволяет подключиться к интернету через порты USB 2.0 и USB 3.0. <u>Ноутбук Super_NB 2001 Gaming</u> <u>оснащен экраном диагональю 21 дюйма с разрешением 1337x228 пикселей.</u></p>

5. ВЫВОДЫ

По приведенным выше примерам можно заключить, что модель **ruGPT3Medium** обладает большей обобщающей способностью, может генерировать более развернутые и вариативные тексты. Это под-

тверждается более низким значением перплексии. Так, например, стоит отметить, что, если в названии присутствует слово «Gaming», модель **ruGPT3Medium** упоминает в описании, что данный ноутбук – игровой, а **ruT5-large** игнорирует это его свойство.

Также отметим, что сгенерированные описания часто могут содержать некоторые характеристики, не указанные во входных данных. Ошибки такого рода наиболее критичны с точки зрения поставленной задачи, т.к. могут ввести в заблуждение покупателя, прочитавшего такое описание, и вместе с тем, такие ошибки сложнее выявлять в автоматическом режиме. Кроме того, описания могут содержать повторы или ошибочные значения характеристик.

Несмотря на некоторые фактические и логические ошибки можно сказать, что все полученные описания в той или иной мере соответствуют характеристикам исходного товара. Данный подход к решению задачи генерации текстовых описаний оказался оправданным. Достаточно лишь сгенерировать некоторое количество описаний товара, выбрать из них наиболее удачный пример и вручную отредактировать его неточности. Таким образом, нейросетевая генерация текстовых описаний может значительно облегчить и удешевить наполнение контентом сайта интернет-магазина или решение другой схожей задачи. Тем не менее, такая система пока не может использоваться как самостоятельная и требует редакции со стороны человека.

Для улучшения качества генерации текстовых описаний можно предложить два подхода:

Во-первых, последовательная генерация предложений описания – по отдельности на каждую характеристику. Это поможет настраивать и контролировать набор характеристик, упоминающихся в описании.

Во-вторых, форматирование выборки таким образом, чтобы заменить значения характеристик в первой части и в тексте описания на специальные токены. Так нейронной сети будет проще находить места расположения значений в тексте. При этом необходима дополнительная постобработка сгенерированных текстов – замена специальных токенов на значения соответствующих им характеристик.

Упомянутые подходы имеют свои сложности в реализации, но потенциально могут повысить качество сгенерированных описаний, что позволит совсем отказаться от человеческой редакции полученных текстов. Они будут рассмотрены и применены в ходе дальнейших исследований.

Литература

1. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need // arXiv.org. – 2017. – URL: <https://arxiv.org/abs/1706.03762> (дата обращения: 05.12.2021).
2. Radford A., Narasimhan K., Salimans T., and Sutskever I. Improving language understanding by generative pre-training // paperswithcode.com. – 2018. – URL: <https://paperswithcode.com/paper/improving-language-understanding-by> (дата обращения: 03.02.2022).
3. Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners // paperswithcode.com. – 2019. – URL: <https://paperswithcode.com/paper/language-models-are-unsupervised-multitask> (дата обращения: 03.02.2022).
4. Brown T.B., Mann B., Ryder N., Subbiah M., Kaplan J., Dhariwal P., Neelakantan A., Shyam P., Sastry G., Askell A., et al. Language models are few-shot learners // arXiv.org. – 2020. – URL: <https://arxiv.org/abs/2005.14165> (дата обращения: 24.01.2022).
5. «Сбер выложил русскоязычную модель GPT-3 Large с 760 миллионами параметров в открытый доступ» // habr.com. – 2019. – URL: <https://habr.com/ru/company/sberbank/blog/524522/> (дата обращения: 10.12.2021).
6. Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer // arXiv.org. – 2019. – URL: <https://arxiv.org/abs/1910.10683> (дата обращения: 14.02.2022).
7. «ruT5, ruRoBERTa, ruBERT: как мы обучили серию моделей для русского языка» // habr.com. – 2019. – URL: <https://habr.com/ru/company/sberbank/blog/567776/> (дата обращения: 08.12.2021).
8. Boris Ginsburg, Patrice Castonguay, Oleksii Hrinchuk, Oleksii Kuchaiev, Vitaly Lavrukhin, Ryan Leary, Jason Li, Huyen Nguyen, Yang Zhang, Jonathan M. Cohen. Stochastic Gradient Methods with Layer-wise Adaptive Moments for Training of Deep Networks // arXiv.org. – 2020. – URL: <https://arxiv.org/abs/1905.11286> (дата обращения: 03.02.2022).

ПРОГНОЗИРОВАНИЕ КООРДИНАТ ДИНАМИЧЕСКОГО ОБЪЕКТА В УСЛОВИЯХ НЕТОЧНОСТИ ИЗМЕРЕНИЙ МЕТОДАМИ НЕЙРОСЕТЕВЫХ ТЕХНОЛОГИЙ

Палюх Б.В., д.т.н., профессор

Тверской государственной технической университет

e-mail: pboris@tstu.tver.ru

Кемайкин В.К., к.т.н., доцент

Тверской государственной технической университет

e-mail: vk-kem@mail.ru

Полыгаев А.С., аспирант

Тверской государственной технической университет

e-mail: a.polygaev@gmail.com

1. ВВЕДЕНИЕ

Задача прогнозирования координат движущейся цели в условиях неточности измерений координат и недостатке априорной информации о характере движения цели крайне актуальна в системах траекторной обработки радиолокационной информации.

Применительно к задаче траекторной обработки задача прогнозирования координат сводится к последовательному решению двух задач: фильтрации измерений и прогнозированию.

Задача фильтрации измерений сводится к задаче прогнозирования временных рядов, которая традиционно решается при помощи применения фильтров Калмановского типа [1, 2]. Как известно, фильтр Калмана успешно применяется в случаях, когда модель системы линейна, а шум измерений некоррелирован и имеет плотность вероятности, соответствующую плотности вероятности нормального (Гауссовского) распределения [3]. Чтобы обойти данные ограничения, известны такие модификации фильтра Калмана, как расширенный фильтр Калмана (Extended Kalman Filter) [4] и ансцентный фильтр Калмана (Unscent Kalman Filter) [5]. Однако, как и классический фильтр Калмана, данные модификации требуют априорных знаний в виде модели процесса, отклонение от которой ведет к некорректным результатам [3]. Задача прогнозирования может быть решена двумя способами: экстраполяцией отфильтрованных координат цели и получением априорной оценки фильтра на следующий момент времени.

В качестве альтернативы традиционному подходу к фильтрации и экстраполяции координат движущейся цели в данной статье предлагается использовать многослойную искусственную нейронную сеть с долговременной краткосрочной памятью (Long Short-Term Memory), обученную на референсной выборке неточных измерений и истинных положений цели. Такой подход не требует построения точной модели наблюдаемого процесса, а также позволяет обеспечить адаптивность подсистемы фильтрации и экстраполяции к изменениям характера движения цели в пределах заданной модели движения.

В отличие от известных работ, описывающих применение рекуррентных нейронных сетей в задаче прогнозирования временных рядов, в данной работе обучение нейронной сети проводится на предварительной выборке координат ограниченной длины, причем в отличие от работ [7, 8] моделируются ошибки измерения координат, а характер движения цели при тестировании сети отличается от модели, используемой при обучении нейронной сети и положенной в основу фильтра Калмана.

2. ФИЛЬТР КАЛМАНА

Алгоритм работы фильтра Калмана заключается в последовательном выполнении двух основных шагов: прогнозирования измерений на основе заложенной модели процесса и обновления состояния фильтра по результатам измерений [6].

Основным достоинством фильтра Калмана применительно к задаче траекторной обработки является тот факт, что фильтр Калмана не требует накопления измерений за несколько шагов и дает оценку согласно модели процесса на каждом шаге измерения.

Также для решаемой задачи важен тот факт, что фильтр позволяет дать априорную оценку состоянию процесса (положению цели) на следующий за текущим измерением момент времени.

Состояние процесса в момент времени n обозначается $x(n)$. Изменение состояния системы описывается уравнением:

$$x(n) = F(n) * x(n - 1) + B(n) * u(n) + W(n), \quad (1)$$

где $F(n)$ – матрица перехода состояний, B – матрица управления, $u(n)$ – вектор управляющих воздействий, $W(n)$ – шум процесса.

Измерения положения цели осуществляются регулярно и связаны с истинными значениями следующим образом:

$$z(n) = H(n) * x(n) + V(n), \quad (2)$$

где $H(n)$ – модель измерений, $V(n)$ – белый шум измерений с нулевым средним значением.

Оценка состояния системы в момент времени $n+1$ равна:

$$\hat{x}(n+1|n) = F(n) * \hat{x}(n|n), \quad (3)$$

где $\hat{x}(n|n)$ – предыдущая оценка состояния.

Оценка измерения на момент времени $n+1$ получается из оценки состояния системы с учетом модели измерений:

$$\tilde{z}(n+1|n) = H(n+1) * \hat{x}(n+1|n). \quad (4)$$

Затем, после получения очередного измерения оно сравнивается с ранее полученной оценкой измерения:

$$\tilde{z}(n+1) = z(n+1) - \tilde{z}(n+1|n). \quad (5)$$

Далее оценка состояния система уточняется на основе полученного измерения и коэффициента усиления фильтра:

$$\hat{x}(n+1|n+1) = \hat{x}(n+1|n) + K(n+1) * \tilde{z}(n+1). \quad (6)$$

где $W(n+1)$ – коэффициент усиления фильтра Калмана.

Усиление фильтра Калмана требует вычислений ковариационной матрицы состояния $P(n+1|n)$ и ковариационной матрицы ошибок $S(n+1)$. Ковариационная матрица состояния вычисляется как:

$$P(n+1|n) = F(n) * P(n|n) * F(n)' + Q(n), \quad (7)$$

где $Q(n)$ – матрица ковариации белого шума процесса.

Ковариационная матрица ошибок рассчитывается следующим образом:

$$S(n+1) = H(n+1) * P(n+1|n) * H(n+1)' + R(n+1), \quad (8)$$

где $R(k+1)$ – ковариация белого шума измерений.

Коэффициент усиления фильтра Калмана рассчитывается как:

$$K(n+1) = P(n+1|k) * H(n+1)' * S(n+1)^{-1}. \quad (9)$$

Затем, ковариационная матрица состояния обновляется с использованием рассчитанного коэффициента усиления фильтра Калмана как:

$$P(n+1|n+1) = P(n+1|n) - K(n+1) * S(n+1) * W(n+1)'. \quad (10)$$

Вычисление выражений 3-10 выполняется в цикле на каждом шаге измерений состояния наблюдаемой системы (рис. 1).

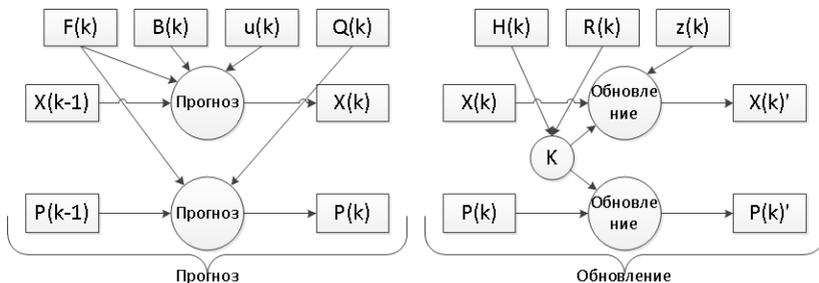


Рис. 1. Принцип работы фильтра Калмана

3. НЕЙРОННАЯ СЕТЬ LSTM

3.1. НЕЙРОННАЯ СЕТЬ LSTM

Сети с долгой краткосрочной памятью или иначе LSTM-сети (от англ. Long Short-Term Memory) – это модификация классической рекуррентной нейронной сети, способные находить долгосрочные и краткосрочные зависимости. Такие сети хорошо приспособлены к обучению на задачах прогнозирования временных рядов в тех случаях, когда измерения достаточно сильно отделены друг от друга по времени.

LSTM сети также как рекуррентные нейронные сети имеют цепную структуру, но устройство клетки сети более сложное: клетка состоит из четырех нейронов, соединенных специальным образом (рис. 2). В работе [10] можно увидеть отличия нейронной сети LSTM от классической рекуррентной нейронной сети.

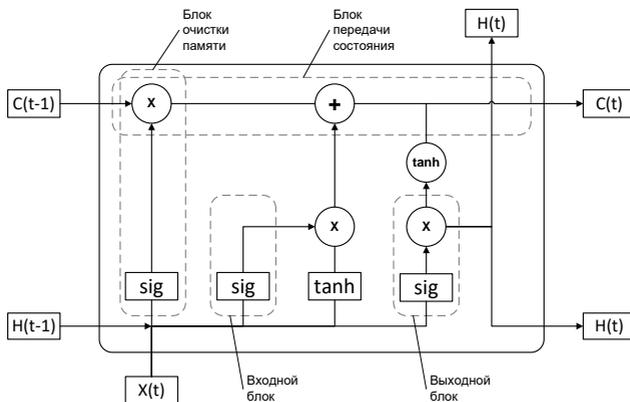


Рис. 2. Структура клетки LSTM-Сети (блоками обозначены нейроны клетки, кругами – операции)

LSTM-сети способны как отслеживать долгосрочные зависимости в данных, так и удалять (забывать) информацию. Этот процесс регулируется структурами, называемыми блоками или вентилями (англ. gates).

Как указано на рис. 2, у каждой клетки имеются две рекуррентные компоненты: выходной вектор H_t и вектор состояния C_t . На вход сети в момент времени t подается вектор измерений X_t , а также выходной вектор H_{t-1} и вектор состояния C_{t-1} предыдущей клетки.

Состояние входного блока рассчитывается как:

$$I_t = \text{sig}(x_t W_i + H_{t-1} W_i + b_i), \quad (11)$$

где x_t - вектор измерений, W_i - вес нейрона входного блока, H_{t-1} - выходной вектор предыдущей клетки, b_i - смещение (англ. bias) функции активации.

Далее работает блок очистки памяти:

$$F_t = \text{sig}(x_t W_f + H_{t-1} W_f + b_f), \quad (12)$$

где x_t - вектор измерений, W_f - вес нейрона блока очистки памяти, H_{t-1} - выходной вектор предыдущей клетки, b_f - смещение (англ. bias) функции активации.

Изменение памяти для блока передачи состояния рассчитывается как:

$$\tilde{C}_t = \text{tanh}(x_t W_c + H_{t-1} W_c + b_c), \quad (13)$$

где W_c - вес нейрона блока передачи состояния, H_{t-1} - выходной вектор предыдущей клетки, b_c - смещение функции активации.

Затем рассчитывается выходной вектор состояния клетки:

$$C_t = F_t C_{t-1} + I_t \tilde{C}_t, \quad (14)$$

где F_t - состояние блока очистки памяти, C_{t-1} - вектор состояния предыдущей клетки, I_t - состояние входного блока, \tilde{C}_t - оценки изменения памяти.

Состояние выходного блока рассчитывается по формуле аналогичной формуле для входного блока:

$$O_t = \text{sig}(x_t W_o + H_{t-1} W_o + b_o), \quad (15)$$

где x_t - вектор измерений, W_o - вес нейрона выходного блока, H_{t-1} - выходной вектор предыдущей клетки, b_o - смещение функции активации.

Затем применяется тангенциальный слой:

$$H_t = O_t * \tanh(C_t), \quad (16)$$

где O_t - состояние выходного блока клетки, C_t – выходной вектор состояния клетки.

3.2. МЕТОД АДАПТИВНОЙ ОЦЕНКИ МОМЕНТОВ ADAM

Для обучений нейронных LSTM и GRU в данной работе используется алгоритм ADAM (англ. Adaptive Moment Estimation) – «метод адаптивной оценки моментов».

В данном методе используются скользящие средние значения, рассчитанные как для градиентов весов, так и для вторых моментов градиентов. Суть алгоритма заключается в делении скорости обучения для каждого веса в отдельности весов на скользящие средние значения градиентов веса.

Значения скользящих среднего первого и второго моментов градиента при обучении рассчитываются следующим образом (m_t от англ. «mean» – значение скользящего среднего для первого момента градиента, v_t от англ. «variance» – для второго момента градиента):

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, \end{aligned} \quad (1)$$

где t – номер итерации, g_t – значение первого момента градиента, g_t^2 – значение второго момента градиента, β_1, β_2 – коэффициенты забывания для первого и второго моментов градиента соответственно.

Первый и второй момент градиента являются частью гиперпараметра алгоритма обучения нейронной сети. По-умолчанию, в начале обучения они имеют значения 0.9 и 0.999 соответственно.

Оценки скользящего среднего для первого и второго моментов градиента на очередном шаге обучения рассчитываются как:

$$\begin{aligned} \hat{m}_t &= \frac{m_t}{1 - \beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1 - \beta_2^t}, \end{aligned} \quad (2)$$

где t – номер итерации, \hat{m}_t – оценка скользящего среднего для первого момента градиента, \hat{v}_t – оценка скользящего среднего для второго момента градиента,

β_1, β_2 – коэффициенты забывания для первого и второго моментов градиента соответственно.

Полученные оценки скользящего среднего для первого и второго моментов градиента затем используются для обновления значений веса нейрона, рассчитанного на основе предыдущего значения:

$$W_{t+1} = W_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} \hat{m}_t, \quad (3)$$

где η – размер шага итерации, \hat{m}_t – оценка скользящего среднего для первого момента градиента, \hat{v}_t – оценка скользящего среднего для второго момента градиента, ϵ – относительно малое значение, используемое, чтобы избежать деления на ноль.

По умолчанию для значения ϵ используется значение 10^{-8} .

3.3. АРХИТЕКТУРА ГЛУБОКОЙ НЕЙРОННОЙ СЕТИ

Предлагаемая архитектура нейронной сети включает в себя два сложносоставных слоя, каждый из которых состоит из рабочего слоя LSTM с заданным количеством нейронов, слоя активации и слоя исключения (англ. dropout) и представлена на рис. 3.

Ожидается, что правильно подобранная функция активации и значение вероятности исключения данных для слоя исключения помогут избежать переобучения сети и повысят адаптивность сети к отклонениям от обучающей выборки и пропускам данных.

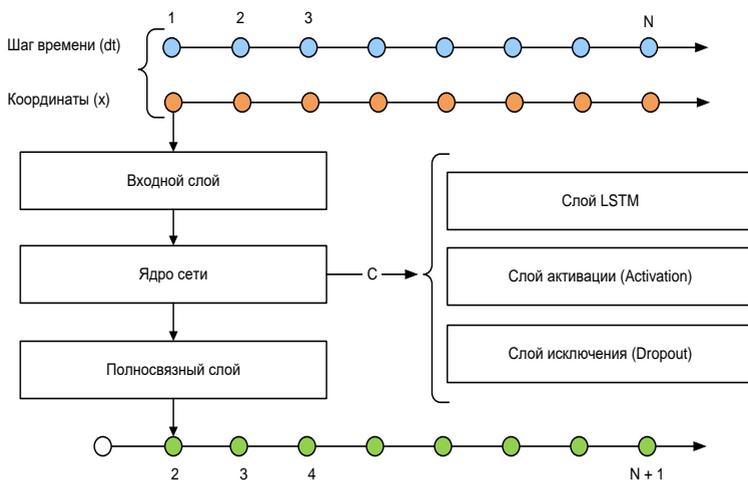


Рис. 3. Схематическая структура глубокой нейронной сети

Таблица 1. Архитектура нейронной сети

Номер слоя	Наименование слоя	Параметры слоя
1	Входной слой	Количество входов: 2
2	LSTM-слой	Функция активации состояния: Tanh Функция активации вентиляей: Sigmoid
3	Слой активации	Тип функции активации: ReLu
4	Dropout-слой	Вероятность исключения: 0,2
5	Полносвязный слой	Количество выходов: 2
6	LSTM-слой	Функция активации состояния: Tanh Функция активации вентиляей: Sigmoid
7	Слой активации	Тип функции активации: ReLu
8	Dropout-слой	Вероятность исключения: 0,2
9	Полносвязный слой	Количество выходов: 1

Итоговые параметры обучения и архитектуры нейронной сети были получены по результатам проведения оптимизации гиперпараметра методами байесовской оптимизации и представлены в табл. 2.

Таблица 2. Результаты оптимизации гиперпараметра нейронной сети

Наименование параметра	Диапазон изменения	Результат оптимизации
Длина обучающей выборки	От 1 до 100	63
Количество составных слоев	От 1 до 3	2
Количество нейронов скрытого слоя	От 1 до 100	33
Тип функции активации	ReLu, Tanh, LeakedReLu	ReLu
Вероятность исключения	От 0 до 0,5	0,27
Пороговое значение градиента	От 0,1 до 5	0,0107
Начальная скорость обучения	От 0,001 до 1	0,0026

4. ОПИСАНИЕ ЭКСПЕРИМЕНТА

В качестве модели движения цели, использовался процесс изменения во времени некоторой величины, заданный уравнением:

$$x = A * \sin(w * t + phi), \quad (16)$$

где $A = 5$, $w = 3 * \pi$, $phi = 0$, время моделирования фиксировано в диапазоне: $t \in [0, 10]$, шаг времени = 0.01.

Данная модель использовалась для обучения нейронной сети, а также в качестве модели, лежащей в основе используемого для сравнения результатов работы нейронной сети фильтра Калмана

Для экспериментальной отработки сформированы несколько наборов измерений на основе выражения 16 с наложением на полученные координаты цели белого гауссового шума с соотношением сигнал-шум (Signal-Noise Rate) заданным значением 5.

Дополнительно, чтобы имитировать несоответствие модели процесса заложенной в фильтр Калмана модели, был сформирован тестовый набор измерений, с соотношением сигнал-шум 10, на основе модифицированной модели измерений, для которой: $A = f(t)$, $w = 3 * \pi$, $phi = 0$.

Используемая нейронная сеть прямого распространения представлена одним слоем нейронов размерностью 4, функция активации – гиперболический тангенс. Для обучения использовалось обучение с учителем, используемый метод обучения – метод обратного распространения ошибки.

Размер обучающей выборки для рассматриваемой сети составляет 2/3 от общей длины набора данных. Данные измерений нормализованы в диапазоне от 0 до 1.

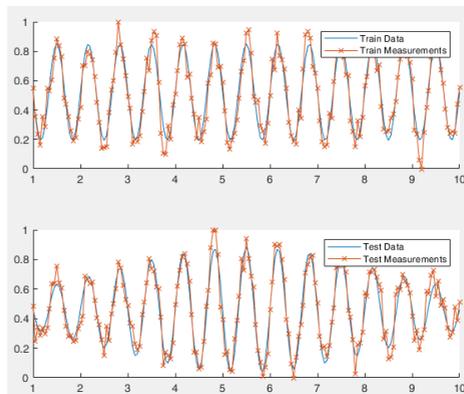


Рис. 4. Обучающий и тестовый наборы данных, SNR = 10

Для сравнения и контроля результатов моделирования дополнительно использовались результаты экстраполяции данных по результатам линейной и сплайновой интерполяции.

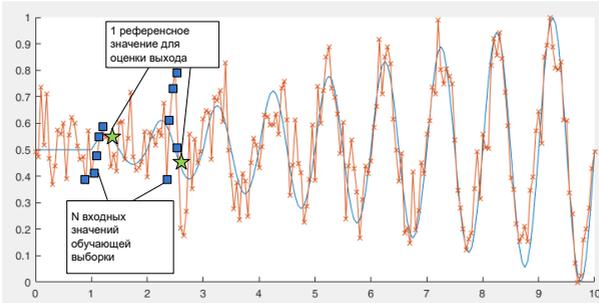


Рис. 5. Иллюстрация процесса обучения нейронной сети

Для анализа результатов для каждого выходного набора данных рассчитывалась средняя абсолютная ошибка, максимальная ошибка, среднеквадратическое отклонение (СКО):

- Средняя абсолютная ошибка (MAE – Mean Absolute Error):

$$MAE = \frac{1}{m} \sum_{i=1}^m |a_i - y_i|$$

- Среднеквадратическое отклонение (MSE – Mean Squared Error):

$$MSE = \frac{1}{m} \sum_{i=1}^m |a_i - y_i|^2$$

- Корень среднеквадратического отклонения (RMSE – Root MSE):

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^m |a_i - y_i|^2}$$

5. РЕЗУЛЬТАТЫ МОДЕЛИРОВАНИЯ

Результаты моделирования для каждого из наборов входных данных представлены в табл. 3, фильтры обозначены следующим образом:

FF NN – нейронная сеть прямого распространения, KF – фильтр Калмана,

linear – экстраполяция по результатам линейной интерполяции,

spline – экстраполяция по результатам сплайновой интерполяции.

Выход нейронной сети в сравнении с фильтром Калмана представлен на рис. 6.

Таблица 3. Результаты моделирования

Фильтр	Средняя абсолютная ошибка	Максимальная ошибка	СКО
Шаг времени модели $t_s = 0.1$			
KF	0.0900	0.2596	0.0111
LSTM	0.0565	0.2111	0.0052
GRU	0.0570	0.2275	0.0048
Deep LSTM	0.0557	0.1947	0.0049
Шаг времени модели $t_s = 0.05$			
KF	0.1169	0.2979	0.0173
LSTM	0.0695	0.2102	0.0073
GRU	0.0748	0.1979	0.0081
Deep LSTM	0.0709	0.1790	0.0077

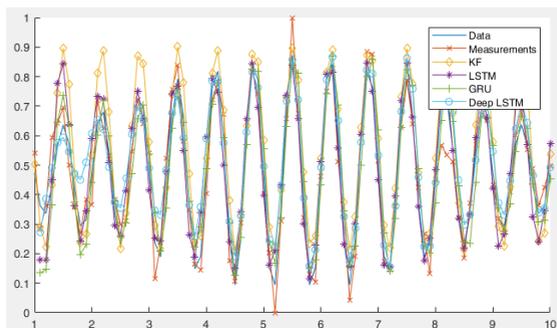


Рис. 6. Данные, измерения, выход нейронной сети и выход фильтра Калмана в сравнении

По результатам моделирования видно, что фильтр Калмана вне конкуренции в ситуации, когда шум измерений относительно высок, а модель процесса в точности соответствует модели, положенной в основу фильтра.

Нейронная сеть прямого распространения, в свою очередь, выдает лучший среди прочих результат в том случае, когда модель процесса подобна, но не повторяет модель, используемую при обучении нейронной сети.

6. ЗАКЛЮЧЕНИЕ

Нейронная сеть прямого распространения, описанная в данной статье, может быть успешно применена в задаче фильтрации измерений (прогнозирования временных рядов) в случае, когда модель протекания процесса не соответствует заранее заданной, а шум измерений по своему характеру не является гауссовым.

Из недостатков данного подхода стоит отметить необходимость обучения нейронной сети на тестовой выборке, что представляет собой определенную вычислительную сложность. Подобная выборка может быть получена на основе модели измерений, лежащей в основе применяемого фильтра Калмана.

Способности нейронной сети к прогнозированию напрямую следуют из её способности к обобщению и выделению скрытых зависимостей между входными и выходными данными. После обучения сеть способна предсказать будущее значение некой последовательности на основе нескольких предыдущих значений и (или) каких-то существующих в настоящий момент факторов. Прогнозирование возможно только тогда, когда предыдущие изменения действительно в какой-то степени предопределяют будущие.

Выбор архитектуры нейронной сети зависит от характера входных данных, их пригодности к классификации и структурированию (для обучения без учителя), от особенностей применения нейронной сети: так рекуррентные нейронные сети имеют более сложную структуру и дольше обучаются, но за счет обратной связи лучше адаптируются к шуму измерений и дают меньшую ошибку.

Вход нейронной сети в задаче для прогнозирования координат представляет собой последовательность координат в предыдущие моменты времени, а выход – координаты в следующий момент времени.

Выбор нелинейной функции активации обусловлен тем, что процесс изменения координат во времени носит нелинейный характер.

Существенное увеличение числа отсчетов для обучения сети может привести к тому, что сеть будет обучаться на устаревших данных, что, в свою очередь, приведет к уменьшению точности прогнозирования.

Литература

1. Arnold M., Milner X.H.R., Witte H., Bauer R. and Braun C. Adaptive AR modeling of nonstationary time series by means of Kalman filtering // in IEEE Transactions on Biomedical Engineering. – May 1998. – Vol. 45, No. 5. – P. 553-562. – DOI: 10.1109/10.668741.

2. Sorensen J.V.T. and Madsen H. Water level prediction skill of an operational marine forecast using a hybrid Kalman filter and time series modeling approach // *Oceans 2003. Celebrating the Past ... Teaming Toward the Future* (IEEE Cat. No.03CH37492), San Diego, CA, USA, 2003. – Vol. 2. – P. 790. – DOI: 10.1109/OCEANS.2003.178415.
3. Brookner E., *Tracking and kalman filtering made easy*, John Wiley and Sons, Inc, New York, USA, 1998.
4. Kalman R.E. A New Approach to Linear Filtering and Prediction Problems» (PDF) // *Journal of Basic Engineering*. – 1960. – 82. – P. 35–45. – DOI: 10.1115/1.3662552.
5. Julier Simon J., Uhlmann Jeffrey K. New extension of the Kalman filter to non-linear systems» (PDF) / In Kadar, Ivan (ed.). *Signal Processing, Sensor Fusion, and Target Recognition VI. Proceedings of SPIE*. – 1997. – 3. – P. 182-193. Bibcode:1997SPIE.3068..182J. CiteSeerX 10.1.1.5.2891. – DOI: 10.1117/12.280797. S2CID 7937456. Retrieved 2008-05-03.
6. Paul Zarchan; Howard Musoff (2000). *Fundamentals of Kalman Filtering: A Practical Approach*. American Institute of Aeronautics and Astronautics, Incorporated. – ISBN 978-1-56347-455-2.
7. Zell Andreas. *Simulation Neuronaler Netze [Simulation of Neural Networks]* (in German) (1st ed.). Addison-Wesley. – 1994. –73 p. – ISBN 3-89319-554-8.
8. Schmidhuber Jürgen (2015-01-01). *Deep learning in neural networks: An overview* // *Neural Networks*. – 61. – P. 85-117. arXiv:1404.7828. doi:10.1016/j.neunet.2014.09.003. – ISSN 0893-6080. PMID 25462637. S2CID 11715509.
9. Goodfellow Ian, Bengio Yoshua, Courville Aaron. *6.5 Back-Propagation and Other Differentiation Algorithms* // *Deep Learning*. – MIT Press, 2016. – P. 200-220. – ISBN 9780262035613.
10. Song X., Liu Y., Xue L., Wang J., Zhang J., Wang J., Jiang L., Cheng Z. Time-series well performance prediction based on Long Short-Term Memory (LSTM) neural network model. *J. Pet. Sci. Eng.* 2020, 186, 106682.
11. Diederik P. Kingma and Jimmy Lei Ba. *Adam: A method for stochastic optimization*. 2014. arxiv:1412.6980v9.

ПРОЕКТИРОВАНИЕ ИНСТРУМЕНТАРИЯ ДЛЯ ВЫБОРА МЕТОДОВ ИНТЕРПРЕТИРУЕМОГО МАШИННОГО ОБУЧЕНИЯ*

Суворова А.В., к.ф.-м.н., доцент

НИУ ВШЭ

e-mail: asuvorova@hse.ru

Смирнова А.В.

НИУ ВШЭ

1. ВВЕДЕНИЕ

Тема интерпретируемого машинного обучения (IML) и объяснимого искусственного интеллекта (XAI) последнее время широко обсуждается как в академических кругах (ведущие конференции (см., например, табл. 1 в [4]), онлайн-курсы, университетские программы), так и в промышленности [8]. Интерпретируемость моделей машинного обучения, необходимость анализа факторов, влияющих на принятие решений, вызывают у общества и научного сообщества все больше вопросов, в том числе касающихся таких тем, как обоснованность решения, алгоритмические предубеждения или дискриминация [2]. Разнообразие существующих методов [3, 21] приводит к дополнительному блоку вопросов: как сделать обоснованный выбор наиболее подходящего метода для конкретного случая, как правильно интерпретировать результаты и как учесть ограничения метода.

Однако, хотя в недавних исследованиях было предложено несколько показателей для оценки интерпретируемости [28], в настоящее время не существует единого метода сравнения алгоритмов интерпретации, позволяющего выбрать лучший алгоритм для каждого случая. Не существует методологии для выбора того, какой алгоритм интерпретации применять в каждой конкретной ситуации, в то время как потребность в таком подходе подчеркивается многими исследователями [17, 28].

Кроме того, несмотря на то, что многие инструменты имеют веб-интерфейс и не требуют навыков программирования, все же для их использования требуется некоторое знание методов XAI. Например, инструмент What-If [27] изначально был ориентирован на широкую аудито-

* Работа выполнена при финансовой поддержке РФФ, проект № 19-71-00064.

рию, а не только на специалистов по машинному обучению, но после апробации стало ясно, что пользователи должны иметь хотя бы некоторый опыт машинного обучения, чтобы конкретизировать проблему и задавать параметры. Более того, недавние исследования показали, что существующие инструменты могут сбивать с толку даже специалистов-практиков в области data science, аналитики склонны чрезмерно доверять этим методам, если результаты представлены в «научном» формате и подкреплены ссылками на публикации [14]; показано, что можно манипулировать доверием пользователей с помощью объяснений [15].

Чтобы упростить процесс выбора метода для объяснения модели машинного обучения и поддержки принятия решений в рабочем процессе построения модели и исследования, мы предлагаем модульную структуру для выбора алгоритма, которую можно легко адаптировать под конкретный сценарий использования, опираясь на результаты оценки пользовательских предпочтений.

2. ОБЗОР И КЛАССИФИКАЦИЯ МЕТОДОВ IML

Более сложные модели чаще используются для решения практических задач, поскольку они обычно более точны [9]. Но такие модели чаще всего являются моделями вида «черный ящик», т.е. не формируют понятного правила предсказания, что вместе с необходимостью объяснения решений, принимаемых моделью, приводит к активному развитию новых методов интерпретации моделей и их более интенсивному использованию в реальных системах [18].

Существует множество понятий, которые исследователи используют для обозначения проблемы преобразования скрытого процесса принятия решений в моделях черного ящика в более понятный, понятный и управляемый человеком процесс: объяснимость, интерпретируемость, понятность, прозрачность, подотчетность, справедливость, непредвзятость, и т. д. [28]. Они различаются с точки зрения желаемых свойств или результатов системы [20] и целевой аудитории [2], среди которых объяснимость и интерпретируемость являются наиболее общими и часто используются взаимозаменяемо [19]. На протяжении всей статьи будут использованы оба термина, означающие «способность объяснить или представить в понятных терминах человеку» [5].

В этом разделе коротко представлены основные свойства методов IML, но подробное описание методов выходит за рамки этой статьи. Гораздо более подробное описание алгоритмов можно найти в книге Кристофа Молнара [21] и в нескольких хороших обзорах [2– 5, 18], в том числе посвященных исследованию человеческого фактора в IML [13], метрикам [28], сравнению доступного программного обеспечения для ХАИ [3].

В работах были предложены различные классификации методов интерпретации в зависимости от масштаба или типа модели, которую метод может объяснить [21], решаемой задачи [17], этапа рабочего процесса, на котором могут быть применены объяснения [28]. Объяснения можно различать следующим образом [3, 28]:

- локальные (объяснение отдельных прогнозов) или глобальные (объяснение модели в целом);
- зависящие от модели (объяснение конкретного типа модели, например, глубокие нейронные сети [1]) или не зависящие от модели (объяснение любой модели, использование только входных и выходных данных, не структуры модели);
- внутренние (объяснение формируется одновременно с построением модели, например дерево решений) или post-hoc (объяснение «поверх» существующей модели);
- статические (разовое объяснение) или интерактивные (обычно объяснение в диалоге с пользователем [3]);
- на основе признаков (feature-based, объяснение на характеристиках данных) или на основе примеров (item-based, объяснение на примерах данных);
- работающие со специфическими типами входных данных (тексты, таблицы, изображения, графики) [18];
- позволяющие ответить на конкретный вопрос (почему, как, что если, почему нет [18]).

К. Молнар предлагает несколько иную классификацию и различает три типа интерпретации моделей: на основе компонентов модели, изучения чувствительности модели и суррогатных моделей [22]. Алгоритмы первого типа строят объяснения на основе внутренних характеристик модели (например, коэффициенты в обобщенных линейных моделях, отдельные правила в деревьях решений или других моделях, основанных на правилах, карты признаков в нейронных сетях). Алгоритмы второго типа (чувствительность модели) позволяют исследовать, как выходные данные модели изменяются в зависимости от входных данных. Этот тип состоит из широкого спектра алгоритмов, включая различные меры важности признаков, динамические меры (например, индивидуальные условные ожидания (ICE [10])). Они могут быть локальными или глобальными (важность признаков). Эти методы обычно не зависят от модели, поскольку они требуют только входных данных и соответствующих прогнозов и не полагаются на внутреннюю структуру модели, но существуют специфичные для модели метрики. Наконец, подход суррогатной модели представляет со-

бой смесь двух других подходов, он приближает модель более простым, интерпретируемым способом (например, линейной моделью или деревом решений), а затем анализирует компоненты интерпретируемой суррогатной модели (один из самых популярных примеров – LIME [24]).

Эти классификации определяют выбор метода на самом первом уровне, выступая в качестве ограничений, и в результате первый модуль в предлагаемом инструментарии выбора модели. Например, если исследователь хочет изучить существующую модель черного ящика на табличных данных, он ограничивает диапазон возможных методов апостериорными подходами, не зависящими от модели. Если, кроме того, главный вопрос, представляющий интерес, заключается в том, почему в конкретном случае получено именно такое предсказание, то методы ограничиваются локальными объяснениями, фокусируясь на влиянии признаков, контрастирующих признаках, репрезентативных примерах и противоречащих (counterfactuals) примерах [17].

3. ОЦЕНИВАНИЕ МЕТОДОВ ИНТЕРПРЕТАЦИИ

Ограничения, описанные в предыдущем разделе, упрощают поиск наилучшего метода интерпретируемости, но обычно все же приводят к списку алгоритмов, а не одному, подходящему под условия. Возможное решение для последующего выбора состоит в том, чтобы оценить оставшиеся алгоритмы и сравнить доступные методы объяснения, чтобы найти предпочтительные объяснения. Проблема заключается в том, что при оценке апостериорных объяснений нет известного правильного объяснения, поскольку реальная внутренняя работа модели неизвестна [28]. Более того, неясно, что такое "хорошее" объяснение, поскольку понятность зависит от того, кто пытается понять объяснение. Существует несколько попыток формализовать эти понятия. Например, в [12] предложен чек-лист, который должны заполнить пользователи, ответив на вопрос, что для них является хорошим объяснением. В [7] авторы разработали рекомендации по проектированию прозрачных интеллектуальных систем, основываясь на ментальных моделях пользователей.

В широко цитируемой классификации способов оценивания объяснений [5] выделяются три категории подходов к оценке:

- оценка итоговых инструментов. Этот вид оценки основан на экспериментах, в которых конечные пользователи решают реальные задачи, используя предложенные инструменты для принятия решений.

- оценка на людях. Этот тип оценки также требует экспериментов с людьми, но, поскольку конечные пользователи (специалисты в предметной области) обычно труднодоступны, а их время дорого, то используется упрощенное приближение к реальной ситуации с непрофессионалами.
- оценка функциональности. Этот тип оценки не требует экспериментов с людьми и использует метрики, основанные на формальном определении интерпретируемости, которые можно оценить математически или с помощью моделирования.

Метрики. Как уже отмечалось, под объяснимостью понимаются различные характеристики. В [28] предоставлен подробный обзор формальных метрик для оценки интерпретируемости с акцентом на свойствах объяснимости (ясность, широта охвата, простота, полнота, обоснованность – подробные примеры см. в таблице 2 в [28]). Обобщая, можно сказать, что метрики для объяснений на основе моделей (например, количество операций во время выполнения, сила взаимодействия, размер модели) обычно относятся только к простоте. Уровень (не)согласия (процент соответствия прогнозов между исходной моделью и объяснениями модели [15]) можно использовать для количественной оценки ясности и обоснованности, но этот показатель можно вычислить только для суррогатных объяснений, где объяснение тоже представляет собой модель, только более простую. Метрики для основанных на примерах объяснений, таких как репрезентативность и разнообразие примеров, относятся к простоте и полноте объяснений. На сегодняшний день существует больше показателей для объяснений на основе признаков (или основанных на признаках), и большинство из них (например, чувствительность, монотонность, избирательность, запоминание важных признаков) ориентированы на надежность, то есть правильность и правдивость.

Оценивание человеком. Как упоминалось выше, общее значение объяснимости – это «способность объяснить или представить в понятных терминах человеку» [5], поэтому «человеческая» часть является существенной при оценивании объяснимости и, как следствие, большая часть исследований по оцениванию объяснений ориентирована на человека. В этих исследованиях описываются конкретные эксперименты, сравнивающие алгоритмы интерпретации [11] или измеряющие удовлетворенность [12], доверие [6] или способность принимать решения на основе объяснений [14].

Другие исследования показывают, что не существует одного наилучшего объяснения, и человеческая оценка качества объяснения зависит от ментальной модели пользователя [20] или, в измеримых

терминах, от роли пользователя в процессе построения и использования модели [13], предметной области [17], уровня знаний [20], ожиданий [17] и предпочтений [25].

Так, если говорить про роли пользователей в процессе построения и использования модели, то авторы [13] предлагают три основные категории ролей заинтересованных сторон: разработчики моделей (model builders: аналитики, специалисты по данным, специалисты по машинному обучению), тестировщики, критики моделей (model breakers: менеджеры по продуктам, эксперты в предметной области, юристы) и потребители моделей (model consumers: конечные пользователи). Большинство алгоритмов и инструментов объяснения ориентированы на разработчиков моделей и процесс отладки моделей или на конечных пользователей и их запросы объяснений, в то время как потребители тестировщиков моделей не имеют надлежащего охвата. Сложность проектирования систем для тестировщиков моделей заключается в том, что они, как правило, не имеют глубоких знаний о методах и их ограничениях и не могут использовать все возможности IML. Предлагаемый нами инструментарий упрощает выбор метода и в первую очередь предназначен для тестирования моделей.

4. ВЫБИРАЕМ ПРАВИЛЬНЫЙ МЕТОД: ИНСТРУМЕНТАРИЙ ВЫБОРА МОДЕЛИ

Предлагаемый нами инструментарий состоит из нескольких модулей:

- Модуль ограничений: формальное описание задачи на основе классификации методов IML на основе свойств (раздел 2 [3]).
- Модуль цели: спецификация на основе вопросов (на какой вопрос пользователь хочет ответить) [17].
- Модуль алгоритмов: доступные алгоритмы и их реализации.
- Модуль метрик: доступные формальные метрики для оценки объяснимости [28].
- (необязательный) Модуль предпочтений: указание пользовательских предпочтений или ролей пользователей для ранжирования методов в соответствии с их полезностью для определенного типа пользователя.
- (необязательный) Модуль вывода: формат объяснения (текстовый, визуальный, числовой).

По данным исследования [17], функции объяснимости должны быть запланированы как часть общего процесса проектирования системы ИИ. В предложенном инструментарии сначала выясняются потребности пользователей в объяснениях, а затем они сопоставляются с

конкретными алгоритмами объяснимости. Инструментарий включает потребности пользователей в качестве одного из входных параметров и упрощает процесс выбора модели, разбивая его на несколько этапов «вопрос-ответ». Конфигурация (пользовательские настройки) хранится в отдельном файле и при необходимости может быть обновлена. Существует также «банк» с моделями интерпретации (модуль алгоритмов), при необходимости могут быть добавлены дополнительные модели.

Сценарий взаимодействия пользователя с системой при условии использования конфигурации по умолчанию, загрузки данных и модели следующий:

- пользователь выбирает вопрос (цель) из доступных вариантов и указывает ограничения (например, локальное, апостериорное, основанное на признаках объяснение);
- сервис на основе конфигурации (заранее определенных предпочтений и ограничений) и выбранного вопроса выбирает подходящую (одну или несколько) модель интерпретации;
- сервис запрашивает необходимые для генерации интерпретации данные и модель;
- сервис передает данные и необходимые параметры выбранной модели интерпретации;
- метрики формального объяснения рассчитываются для выбранных моделей интерпретации;
- модель интерпретации обрабатывает запрос и форматирует вывод в соответствии с заданными параметрами;
- интерпретация возвращается сервису, которая перенаправляет ее пользователю с выводом метрик, если необходимо.

Чтобы формализовать пользовательские предпочтения, было проведено исследование [26] пользователей для выявления типов объяснений, которые пользователи отмечают как полезные, улучшающие пользовательский опыт. Выявление предпочтений проводилось с помощью совместного анализа (conjoint analysis) [23], позволяющего не задавать прямые вопросы о том, какой алгоритм пользователь хочет использовать или какая функция важнее (например, полезнее ли иметь конкретный пример или увидеть общая картина), а сравнивать конечные результаты.

В качестве вариантов интерпретации в исследовании были включены важность признаков, подходы, основанные на примерах, включая контр-примеры, примеры «что, если», аналогичные случаи из исторических данных, а также динамика эффекта (индивидуальные условные ожидания). Эти алгоритмы были выбраны, поскольку они могут отвечать на самые популярные вопросы пользователей [17] и в то же время иллюстрируют разные подходы к ответам на одни и те же вопросы, которые будут сравниваться в исследовании пользователей.

Исследование проводилось среди тех, кто использует результаты модели машинного обучения для принятия решений (менеджеры, владельцы продуктов) и конечных потребителей – тех, на кого влияют решения системы машинного обучения, такие как пользователи продукта. Члены обеих групп не участвуют в построении систем машинного обучения и хотят понимать конкретные решения, а не оценивать показатели производительности. Участники были набраны в крупных российских ИТ-компаниях, а также среди студентов и аспирантов крупного российского вуза. В итоговую выборку вошли 119 участников, однако только 56% из них прошли опрос полностью. Большинство набранных респондентов (33%) составляют ключевую целевую группу – менеджеры и владельцы продуктов, 20% – разработчики программного обеспечения и инженеры (не ориентированные на данные), 10% – специалисты по данным, 20% – аналитики и 18% – бакалавры или магистранты.

В результате исследования было выявлено, что большинство респондентов предпочитали интерпретации в формах динамики эффекта и контр-примеров, наименее предпочтительным подходом оказалась важность признаков, что может быть связано с более высокими уровнями абстракции интерпретации [25].

Общая идея заключается в использовании «конфигурации по умолчанию» для объяснения, когда пользователи формулируют вопрос простым языком (например, «Как я могу изменить результат») и получают объяснение в простой графической или текстовой форме [26]. В результате пользователю не нужно знать даже имя алгоритма. Система сопоставляет вопрос пользователя (выбранный из предопределенного списка) с конкретным алгоритмом IML, подстраиваясь под нужды и предпочтения пользователя в объяснениях. Пример взаимодействия представлен на рис. 1.

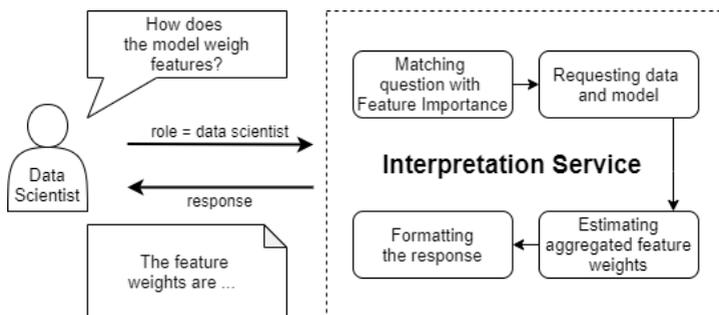


Рис. 1. Пример взаимодействия с системой для одной роли [25]

5. ЗАКЛЮЧЕНИЕ

Растущий интерес к интерпретируемому машинному обучению привел к появлению множества новых тем для исследований, включая изучение различий в потребностях пользователей. Различные варианты использования требуют разных стратегий объяснения, но большинство исследований сосредоточено либо на создателях моделей (аналитиках, специалистах по данным, специалистам по машинному обучению), либо на потребителях моделей. Тем не менее, в нескольких исследованиях подчеркивается необходимость объяснения модели в процессе разработки модели для поддержки связи между разработчиками моделей и экспертами в предметной области.

Мы представляем основу для создания систем на основе вопросов для исследования модели и выбора объяснения. Платформа сочетает в себе ограничения, основанные на желаемых свойствах объяснения или начальных ограничениях модели, сопоставление между простыми вопросами и методами IML, встроенную модель предпочтений, которая позволяет экспертам в предметной области интерактивно исследовать модель, не вдаваясь в детали методов, и вычисление формальных метрик в качестве дополнительных критериев для выбора объяснения. Предлагаемый инструментарий является модульным, его можно расширить дополнительными методами и метриками и настроить под конкретный рабочий процесс.

Литература

1. Angelov P., Soares E. Towards explainable deep neural networks (xdnn) // *Neural Networks*. – 2020. – Vol. 130. – P.185-194.
2. Arrieta A.B., Diaz-Rodriguez N., Del Ser J., et al. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI // *Information Fusion*. – 2020. – Vol. 58. – P. 82-115.
3. Arya V., Bellamy R.K., Chen P.Y., et al. One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques // *arXiv preprint*. – 2019. – URL: [arXiv preprint arXiv:1909.03012](https://arxiv.org/abs/1909.03012).
4. Carvalho D.V., Pereira E.M., Cardoso J.S. Machine learning interpretability: A survey on methods and metrics // *Electronics*. – 2019. – Vol. 8 (8). – P. 832.
5. Doshi-Velez F., Kim B. Towards A Rigorous Science of Interpretable Machine Learning // *arXiv preprint*. – 2017. – URL: <http://arxiv.org/abs/1702.08608>.
6. Drozdal J., Weisz J., Wang D., et al. Trust in automl: Exploring information needs for establishing trust in automated machine learning systems // *Proceedings of the 25th International Conference on Intelligent User Interfaces*. – 2020. – P. 297-307.

7. Eiband M., Schneider H., Bilandzic M., et al. Bringing transparency design into practice // 23rd International Conference on Intelligent User Interfaces. – 2018. – P. 211-223.
8. Gade K., Geyik S.C., Kenthapadi K., et al. Explainable AI in industry: practical challenges and lessons learned: implications tutorial // Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency. – 2020. – P. 699-699.
9. Gaudl S.E., Bryson J.J. The extended ramp model: A biomimetic model of behaviour arbitration for lightweight cognitive architectures // Cognitive Systems Research. – 2018. – Vol. 50. – P. 1-9.
10. Goldstein A., Kapelner A., et al. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation // Journal of Computational and Graphical Statistics. – 2015. – Vol. 24 (1). – P. 44-65
11. Hase P., Bansal M. Evaluating explainable AI: Which algorithmic explanations help users predict model behavior? // arXiv preprint. – 2020. – URL: arXiv preprint arXiv:2005.01831.
12. Hoffman R.R., Mueller S.T., Klein G., Litman J. Metrics for explainable AI: Challenges and prospects // arXiv preprint. – 2018. – URL: arXiv:1812.04608.
13. Hong S.R., Hullman J., Bertini E. Human factors in model interpretability: Industry practices, challenges, and needs // Proceedings of the ACM on Human-Computer Interaction. – 2020. – Vol. 4 (CSCW1). – P. 1-26.
14. Kaur H. et al. Interpreting interpretability: understanding data scientists' use of interpretability tools for machine learning // Proceedings of the 2020 CHI conference on human factors in computing systems. – 2020. – P. 1-14.
15. Lakkaraju H., Bastani O. "How do I fool you?" Manipulating user trust via misleading black box explanations // Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society. – 2020. – P. 79-85.
16. Lakkaraju H., Kamar E., Caruana R., Leskovec J. Interpretable & explorable approximations of black box models // arXiv preprint. – 2017. – URL: arXiv:1707.01154.
17. Liao Q.V., Gruen D., Miller S. Questioning the AI: informing design practices for Explainable AI user experiences // Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. – 2020. – P. 1-15.
18. Linardatos P., Papastefanopoulos V., Kotsiantis S. Explainable AI: A review of machine learning interpretability methods // Entropy. – 2021. – Vol. 23 (1). – P. 18.
19. Miller T. Explanation in artificial intelligence: Insights from the social sciences // Artificial Intelligence. – 2019. – Vol. 267. – P. 1-38.
20. Mohseni S., Zarei N., Ragan E. D. A multidisciplinary survey and framework for design and evaluation of explainable AI systems // ACM Transactions on Interactive Intelligent Systems (TiIS). – 2021. – Vol. 11, No. 3-4. – P. 1-45.
21. Molnar C. Interpretable machine learning. – 2020. – URL: <https://christophm.github.io/interpretable-ml-book/>
22. Molnar C., Casalicchio G., Bischl B. Interpretable machine learning—a brief history, state-of-the-art and challenges // arXiv preprint. – 2020. – URL: arXiv:2010.09337.

23. Rao V.R., et al. Applied conjoint analysis. – Springer, 2014.
24. Ribeiro M.T., Singh S., Guestrin C. "Why Should I Trust You?" // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD'16. – 2016. – P. 1135-1144.
25. Smirnova A. Tools for Analysis of Machine Learning Models Interpretations. Master's thesis, National Research University Higher School of Economics, St.Petersburg, Russia, 2020.
26. Smirnova A., Suvorova A. Machine learning models interpretations: User demands exploration // International Conference on Digital Transformation and Global Society. – 2020. – P. 107-116.
27. Wexler J., Pushkarna M., Bolukbasi T., et al. The What-If tool: Interactive probing of machine learning models // IEEE transactions on visualization and computer graphics. – 2019. – Vol. 26 (1). – P. 56-65.
28. Zhou J., Gandomi A.H., Chen F., Holzinger A. Evaluating the quality of machine learning explanations: A survey on methods and metrics // Electronics. – 2021. – Vol. 10 (5). – P. 593.

СПОСОБЫ ПОВЫШЕНИЯ КАЧЕСТВА ОБУЧЕНИЯ НЕЙРОСЕТЕВЫХ ДИАГНОСТИЧЕСКИХ МОДЕЛЕЙ СЛОЖНЫХ ТЕХНИЧЕСКИХ ОБЪЕКТОВ

*Федотов М.В., к.т.н., зав. лабораторией
АО «ВНИКТИ»*

e-mail: fedotov-mv@vnikti.com

*Шарапов А.Л., инженер-программист 1 кат.
АО «ВНИКТИ»*

e-mail: sharapov-al@vnikti.com

*Грачев В.В., д.т.н., доцент
Петербургский государственный университет путей
сообщения Императора Александра I
e-mail: v_grach@mail.ru*

1. ПОСТАНОВКА ЗАДАЧИ

Сегодня можно наблюдать широкое применение интеллектуальных методов обработки информации для решения разнообразных задач в самых различных областях деятельности, таких как медицина, экономика, интернет, мультимедиа, техника и т.д. Не является исключением и техническая диагностика, где для синтеза интеллектуальных диагностических моделей сложных технических объектов используются методы обучения с учителем. Однако примеров реального применения таких систем известно не так много. Одним из основных ограничивающих факторов является сложность формирования обучающих выборок достаточного объема для всех классов технического состояния сложного технического объекта. И здесь существует несколько проблем.

Во-первых, обучающие данные, идентифицированные для одного конкретного объекта, не всегда могут быть использованы для идентичного другого объекта ввиду наличия технологического разброса характеристик их основных узлов и агрегатов.

Во-вторых, обучающая выборка должна включать образцы, соответствующие каждому классу технического состояния (ТС) объекта диагностики (ОД). Определение класса текущего ТС сложного технического ОД во многих случаях невозможно без вывода его из эксплуатации и частичной разборки с целью инструментального определения

значений параметров, непосредственно характеризующих его состояние и не контролируемых встроенными средствами диагностирования. Для многих видов оборудования инструментальных методов контроля ТС не существует. Примером может служить фильтр масла, для которого регламентирован только предельный перепад давления, но отсутствуют методики инструментальной оценки его фактического состояния.

В третьих, качественное обучение интеллектуальных диагностических моделей требует большого объема данных. Получить необходимые объемы данных, даже используя математические модели объекта, не всегда удастся, т.к. порой для получения нового вектора требуется перестройка всей разработанной модели.

Целью работы является исследование различных способов формирования обучающих выборок для нейросетевых диагностических моделей сложных технических объектов с целью повышения качества их обучения.

2. ОБУЧЕНИЕ НЕЙРОСЕТЕВОЙ ДИАГНОСТИЧЕСКОЙ МОДЕЛИ НА РЕАЛЬНЫХ ДАННЫХ МОНИТОРИНГА ОД В ЭКСПЛУАТАЦИИ

Диагностические модели, построенные на основе искусственных нейронных сетей (ИНС) и обученные на реальных данных, размеченных с использованием результатов периодического инструментального контроля ТС ОД (рис. 1), как правило, обеспечивают высокое качество диагностирования.

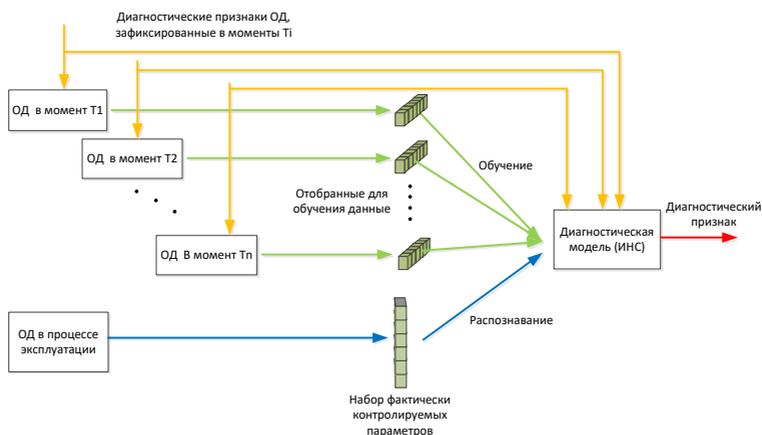


Рис. 1. Формирование обучающей выборки по данным мониторинга ОД

Примером может служить диагностическая модель состояния смазочного масла дизельного двигателя тепловоза. Для построения модели была использована трехслойная искусственная нейронная сеть прямого распространения, реализованная на языке Python с использованием библиотеки глубокого машинного обучения Keras [1]. Структура нейронной сети была выбрана авторами аналогичной использовавшимся ранее для задач диагностики системы смазки [2].

Входными данными нейронной сети являются параметры системы смазки, контролируемые системой управления и диагностики тепловоза и передаваемые с помощью системы удаленного мониторинга на сервер диагностики: давление масла на входе в дизель, давление масла на входе в полнопоточный фильтр, температура масла на выходе из дизеля, частота вращения коленчатого вала дизеля, позиция контроллера машиниста. Выходом нейронной сети является кинематическая вязкость масла, приведенная к стандартным условиям измерения [3].

Обучающая выборка формировалась с использованием результатов лабораторных анализов масла, производимых в соответствии с регламентом на каждом цикле технического обслуживания локомотива (около 30 суток) [4]. С этими данными были сопоставлены значения параметров, зарегистрированные подсистемой бортовой диагностики тепловоза в течение суток, предшествующих дате лабораторного анализа. Все данные, используемые в диагностических моделях, были предварительно отфильтрованы и нормализованы.

Для обучения моделей были сформированы два варианта обучающих выборок. В первый вариант выборки включены данные двух локомотивов, находящихся в работоспособном состоянии, с известными минимальными и максимальными значениями вязкости масла, достигнутыми в процессе эксплуатации. Во второй вариант выборки были добавлены векторы параметров, соответствующих диагностируемому локомотиву. Цикл обучения состоял из 1500 эпох.

Обученные таким образом модели использовались для обработки архивов данных, накопленных на сервере. Полученные результаты сглаживались методом скользящего среднего с суточным окном усреднения. Результаты обработки данных по 20 секциям тепловозов 2ТЭ25КМ за период более 2 лет вместе с результатами периодического лабораторного анализа масла приведены на рис. 2.

Как следует из рисунка, модель, обученная с использованием собственных данных, показывает более высокую точность (ошибка отклика в контрольных точках не превышает 7 %). В ее результатах также отсутствуют ложные аномалии, присущие результатам работы модели, обученной на выборке первого варианта (период июль-август

2021). В целом же модели, обученные по обоим вариантам, показывают приемлемые результаты и надежно фиксируют скачкообразное изменение вязкости масла в моменты его плановой замены.

Более хорошие результаты дает обучение нейросетевой модели на выборке, состоящей только из собственных данных (рис. 3). Ошибка такой модели не превышает 4 %. Минусом такого подхода является возможное ограничение области определения модели вследствие отсутствия в обучающей выборке данных для некоторых режимов работы ОД, встречающихся в эксплуатации, но отсутствовавших в период формирования выборки.

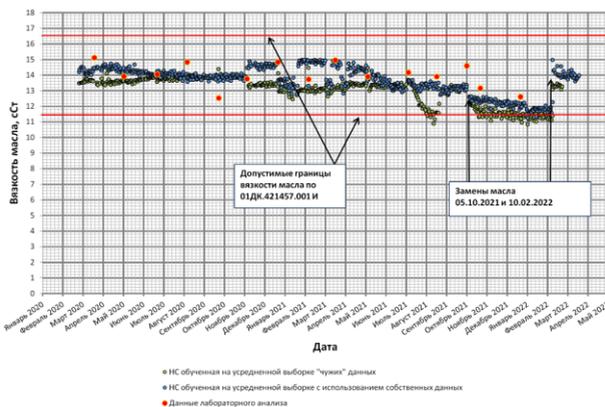


Рис. 2. Результаты определения вязкости масла при помощи нейросетевой модели, обученной на выборках без использования и с использованием собственных данных

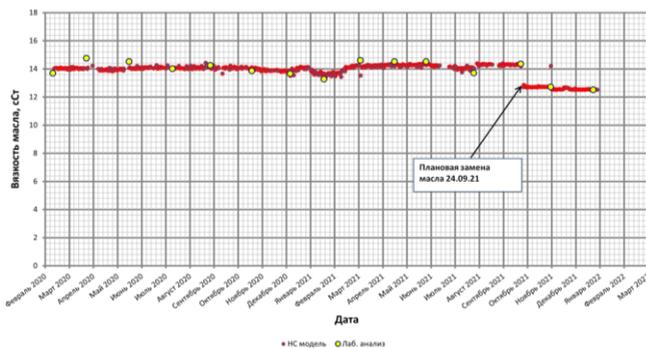


Рис. 3. Результаты определения вязкости масла при помощи нейросетевой модели, обученной на собственных данных

3. ОБУЧЕНИЕ НЕЙРОСЕТОВОЙ ДИАГНОСТИЧЕСКОЙ МОДЕЛИ НА ДАННЫХ, СИНТЕЗИРОВАННЫХ С ПОМОЩЬЮ АНАЛИТИЧЕСКОЙ МОДЕЛИ ОД

Реализовать подход к диагностике сложного ОД, описанный в предыдущем разделе удастся нечасто, т.к. большинство узлов современного сложного оборудования, например локомотива, имеет высокий уровень надежности, не позволяющий сформировать выборку достаточного объема для всех классов состояния ОД. Не всегда существует техническая возможность непосредственного измерения значений информативных диагностических признаков ОД без его разборки.

Одним из вариантов решения этой проблемы является создание специализированного класса аналитических моделей, задачей которых является синтез исходных данных, соответствующих всему диапазону изменения диагностических признаков моделируемого объекта (рис. 4), с последующим использованием их для обучения диагностических моделей. Аналитические модели могут быть выполнены в виде математической, имитационной или другой модели. Размерность получаемых на выходе аналитической модели данных, как правило, выше размерности вектора реально-контролируемых параметров ОД. Поэтому из синтезированных моделью данных предварительно отбираются параметры, соответствующие контролируемым на реальном ОД.

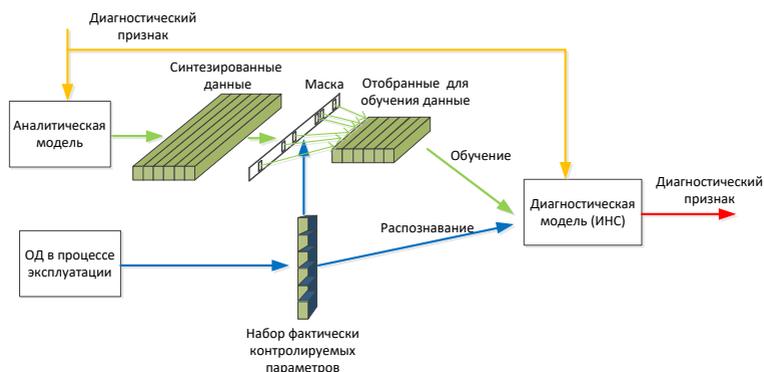


Рис. 4. Формирование обучающей выборки с использованием аналитической модели ОД

Примером такого подхода является создание нейросетевой диагностической модели состояния системы смазки тепловозного дизеля [5]. На первом этапе данной работы была разработана аналитическая модель системы смазки и произведена ее валидация. Далее при помо-

ци аналитической модели был синтезирован массив векторов, соответствующий допустимому в эксплуатации диапазону изменения диагностических признаков, при этом компоненты векторов соответствовали параметрам, контролируемым подсистемой бортовой диагностики тепловоза. Полученный массив использовался в качестве обучающей выборки для нейросетевых диагностических моделей системы смазки. Первая из них оценивает загрязнение фильтра масла, вторая определяет степень износа подшипников коленчатого вала.

С учетом того, что погрешность разработанной аналитической модели составила около 7%, а разработанные нейросетевые модели при тестировании показали среднеквадратичную ошибку отклика не более 3%, можно оценить точность диагностической модели на уровне 8%. На рис. 5 приведены результаты диагностирования состояния фильтра масла тепловоза ТЭП70БС диагностической моделью, обученной с помощью синтезированной выборки исходных данных.

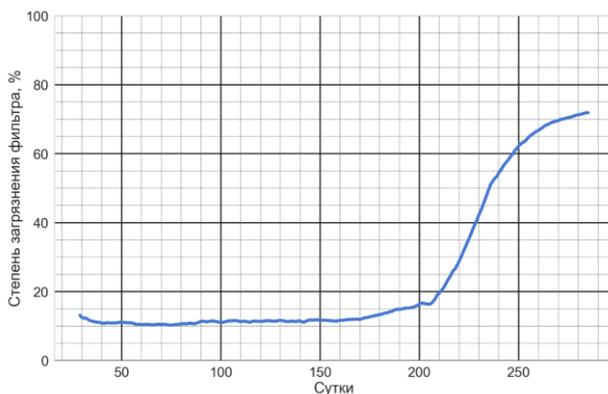


Рис. 5. Определение степени загрязнения фильтра масла тепловоза при помощи нейросетевой модели, обученной на синтезированных данных

4. СИНТЕЗ ОБУЧАЮЩИХ ВЫБОРОК ПРИ ПОМОЩИ ГЕНЕРАТИВНОЙ СОСТЯЗАТЕЛЬНОЙ СЕТИ

Низкая контролепригодность ОД, особенно прежних лет выпуска, отсутствие непрерывного контроля параметров в процессе эксплуатации, в сочетании с высокими абсолютными показателями надежности – все это значительно затрудняет процесс формирования обучающих выборок.

Описанный в предыдущем разделе подход, основанный на использовании развернутой аналитической модели ОД, решает проблему лишь отчасти, поскольку аналитическая модель не позволяет достоверно учесть и воспроизвести случайный разброс характеристик ОД, обусловленный технологическими допусками и фактическим состоянием основных его узлов и агрегатов.

В связи с этим значительный интерес представляет возможность расширения обучающих выборок диагностических моделей методами генеративного моделирования. Генеративные модели строятся, как правило, на базе алгоритмов глубокого машинного обучения и предназначены для формирования наборов данных, статистически идентичных реальным или же синтезированным моделью данным [6]. Сгенерированные таким образом массивы данных могут быть использованы для обучения нейросетевых диагностических моделей (рис. 6).

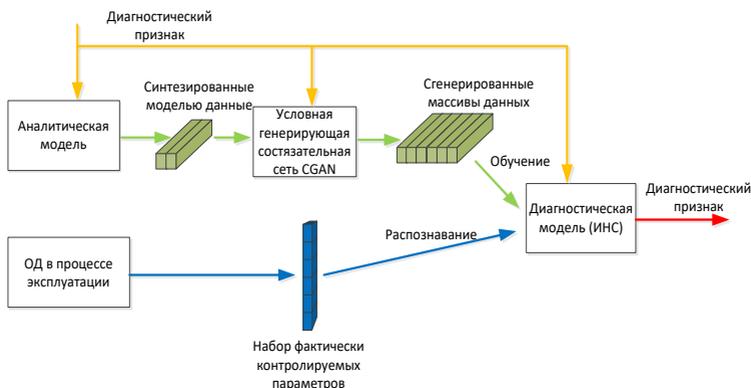


Рис. 6. Формирование обучающей выборки с использованием аналитической и генеративной моделей

Настоящим прорывом в области генеративного моделирования можно считать появление в 2014–2015 гг. генеративных состязательных сетей GAN (Generative Adversarial Networks) [7]. GAN-сеть состоит из двух отдельных подсетей: генератора и дискриминатора (рис. 7). Задачей генератора является преобразование случайного шума в наблюдения, которые выглядят так, как будто были выбраны из исходного набора данных. Дискриминатор должен определить, исходит ли наблюдение из оригинального набора данных или является одной из «подделок» генератора. Главная особенность генеративной состязательной сети – попеременное обучение двух подсетей: генератор постепенно совершенствуется в «об-

мане» дискриминатора, а дискриминатор, адаптируясь, учится правильно определять фальшивые наблюдения, что заставляет генератор искать новые способы обмана дискриминатора, и т.д.



Рис. 7. Структура генеративной состязательной сети

В настоящее время уже существует большое число модификаций GAN-сетей для самых различных применений [7]. И хотя пока основной областью практического применения GAN-сетей является обработка изображений и видео, уже известны примеры использования данного класса сетей для улучшения набора обучающих данных. Например, компания NVIDIA объявила, что использует GAN для увеличения набора данных медицинских снимков компьютерной томографии головного мозга при различных заболеваниях [8]. По информации NVIDIA, эффективность классификации с использованием только классических данных составляет: 78,6%, в то время как при расширении обучающих выборок за счет добавления синтетических данных чувствительность может быть увеличена до 85,7%. Помимо создания других элементов данных внутри класса состояния ОД, GAN-сети могут быть применены для синтеза новых, ранее неразличимых классов.

С целью оценки возможности применения генеративных сетей для формирования массивов обучающих данных на языке Python с применением библиотек Keras и TensorFlow была реализована GAN-сеть, для обучения которой использовались фрагменты 32 индикаторных диаграмм рабочего процесса дизеля 1А-5Д49 теплового двигателя 2ТЭ116, полученные в процессе реостатных испытаний с применением диагностического комплекса «Магистраль-Э» [9]. На рис. 8 и 9 приведены истинные и сгенерированные обученной GAN-сетью информативные фрагменты индикаторных диаграмм, соответствующие двум различным состояниям дизеля: исправному и отказу топливной аппаратуры.

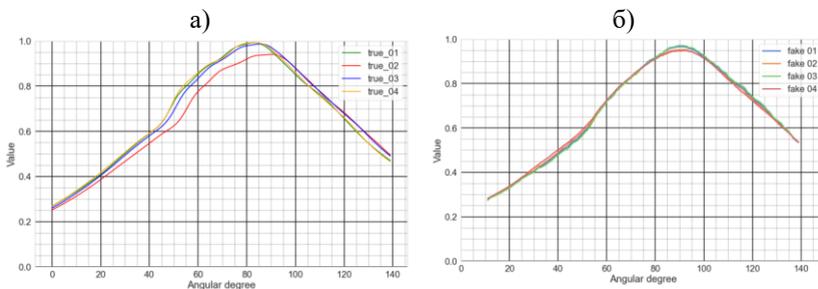


Рис. 8. Истинные (а) и сгенерированные (б) индикаторные диаграммы для исправного состояния дизеля

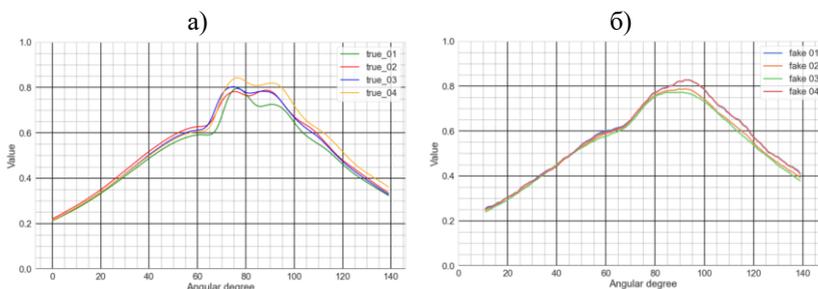


Рис. 9. Истинные (а) и сгенерированные (б) индикаторные диаграммы для отклонения по углу подачи топлива в дизель

Как следует из рисунков, обученная сеть воспроизводит на сгенерированных диаграммах основные особенности формы кривой изменения давления в цилиндре, позволяющие использовать такие диаграммы в качестве объектов обучающей выборки нейросетевой диагностической модели рабочего процесса дизеля.

С целью определения возможности использования сгенерированных GAN-сетью данных для обучения нейросетевых диагностических моделей предлагается две стадии проверки. На первой стадии проверяется качество работы нейросетевой модели, обученной с использованием «поддельных» данных, на тестовой выборке из реальных данных. На второй стадии для обучения модели используются реальные данные, после чего осуществляется проверка качества ее работы на выборке из сгенерированных данных. Если в обоих случаях достигается приемлемая точность модели, сгенерированные векторы данных могут наряду с реальными данными использоваться для обучения диагностической модели с целью повышения точности и достоверности диагноза.

5. ВЫВОДЫ

Приведенные в статье варианты повышения качества обучения нейросетевых диагностических моделей, применяемых для оценки ТС сложных технических объектов, были исследованы на данных эксплуатируемых тепловозов 2ТЭ25КМ, ТЭП70БС и 2ТЭ116У.

Предлагаемые варианты повышения качества обучения нейросетевых диагностических моделей могут значительно расширить круг решаемых задач по диагностике технического состояния сложных технических объектов.

Использование современных методов интеллектуальной обработки данных, таких как генеративное глубокое обучение, открывает новые возможности практического применения технологий машинного обучения.

Литература

1. Джулли А., Пал С. Библиотека Keras – инструмент глубокого обучения. Реализация нейронных сетей с помощью библиотек Theano и TensorFlow. – М.: ДМК Пресс, 2018. – 294 с.
2. Федотов М.В. Интеллектуальное управление техническим состоянием локомотива / М.В. Федотов, Ю.И. Клименко, В.В. Грачев // Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов X-й Международной конференции (ИММВ-2021, Коломна, 17-20 мая 2021 г.). В 2-х томах. Т. 2. – Смоленск: Универсум, 2021. – С. 356-368.
3. ГОСТ 33-2016. Нефть и нефтепродукты. Прозрачные и непрозрачные жидкости. Определение кинематической вязкости. – М.: Стандартиформ, 2017. – 39 с.
4. Локомотивы и моторвагонный подвижной состав. Инструкция по применению смазочных материалов. 01.ДК.421457.001 И. // Открытое акционерное общество «Российские железные дороги». – М., 2005. – 104 с.
5. Федотов М.В., Грачев В.В. Предиктивная система технического диагностирования локомотивов с использованием методов интеллектуального анализа данных // Транспорт Российской Федерации. – 2020. – № 6. – С. 28–34.
6. Николенко С.И. Глубокое обучение. Погружение в мир нейронных сетей / Николенко С.И., Кадурын А.А., Архангельская Е.О. – СПб.: Питер, 2018. – 480 с. – (Библиотека программиста).
7. Фостер Д. Генеративное глубокое обучение. Творческий потенциал нейронных сетей. – СПб.: Питер, 2020. – 336 с. – (Бестселлеры O'Really).
8. Maayan F., Diamant I., Klang E., Amitai M., Goldberger J., Greenspan H. GAN-based Synthetic Medical Image Augmentation for increased CNN Performance in Liver Lesion Classification // arXiv.org. – URL: <https://arxiv.org/abs/1803.01229v1> (дата обращения: 11.04.2022).
9. Перминов В.А., Клименко Ю.И., Васин П.А., Нестеров И.Э. К вопросу обоснования выбора ПЭЛ для определения уровня энергоэффективности тепловозов косвенным способом // Вестник ВНИКТИ. – Коломна, 2015. – № 97. – С. 34-44.

СРАВНИТЕЛЬНОЕ ИССЛЕДОВАНИЕ ДИАГНОСТИКИ БОЛЕЗНИ ПАРКИНСОНА С ПОМОЩЬЮ МАШИННОГО ОБУЧЕНИЯ

Хаддад Некула, аспирант

МГТУ им. Н.Э. Баумана

e-mail: nekoula@inbox.ru

Мышенков К.С., д.т.н., профессор

МГТУ им. Н.Э. Баумана

e-mail: myshenkovks@bmstu.ru

Хаддад Гассан, д.т.н., профессор

Университет Дамаска

e-mail: dr.eng.ghassan.haddad@gmail.com

1. ВВЕДЕНИЕ

Диагноз болезни Паркинсона (БП) традиционно основывается на моторных симптомах. Несмотря на то, что немоторные симптомы (например, когнитивные изменения, такие как проблемы с вниманием и планированием, нарушения сна и др.) присутствуют у многих пациентов до начала БП [1], их сложно оценить, им не хватает специфичности, и они отличаются вариабельностью от пациента к пациенту [2]. Таким образом, некоторые симптомы еще не позволяют поставить независимый диагноз болезни Паркинсона [3], хотя некоторые из них использовались в качестве вспомогательных диагностических критериев [4]. Соответственно, в работе были проанализированы два разных набора данных, которые содержат информацию о моторных признаках пациента.

Первый набор данных содержит показатели походки 93 пациентов с идиопатической болезнью Паркинсона (средний возраст 66.3 года, 63% мужчин) и 73 здоровых лиц (средний возраст 66.3 года, 55% мужчин) [5]. Этот набор данных включает в себя записи вертикальной реакции опоры субъектов, когда они шли в своем обычном, самостоятельно выбранном темпе в течение примерно 2 минут по ровной поверхности. Под каждой ступней было расположено 8 датчиков (рис. 1), измеряющих силу (в ньютонах) как функцию времени. Выход каждого из этих 16 датчиков был оцифрован и записан с частотой дискретизации 100 Гц. Для этого набора данных были

выбраны 4 признака: стандартное отклонение для общей силы, рассчитанной по всем датчикам на левой ноге; стандартное отклонение для общей силы, рассчитанной по всем датчикам на правой ноге; вес и скорость ходьбы испытуемого.

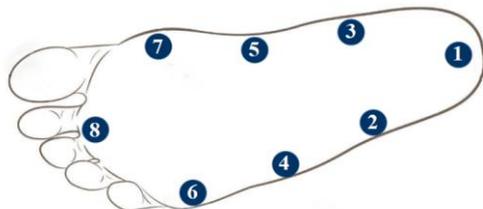


Рис. 1. Расположение датчиков на стопе

Второй набор данных включает информацию о нажатиях клавиш, полученную от 85 испытуемых, из которых 43 здоровых, а у 42 пациентов диагностирована болезнь Паркинсона. Основными признаками данного набора данных являются: скорость печати; результат смены постукивания пальцами и результат нажатия одной клавиши [6].

2. МЕТОД РЕШЕНИЯ

В данной работе решается задача классификации болезни Паркинсона с помощью алгоритма градиентного бустинга [7], поскольку этот алгоритм показывает хорошие результаты для решения этого класса задач [8, 9]. Параметры алгоритма градиентного бустинга приведены в табл. 1.

Таблица 1. Параметры алгоритма градиентного бустинга

Глубина решающего дерева	Скорость обучения	Показатели оценки	Коэффициент разделения
5	0.01	Значение интервала площади под PR-кривой (AUC-PR)	0.7 для обучения, 0.3 для тестирования. (Данные были разделены случайным образом)

После обучения алгоритма были рассчитаны значения вектора Шепли (Shapley) $SHAPValues(i)$, учитывающие влияние каждого i -го признака на выход модели:

$$SHAPVALUES(i) = \sum_{S_i \subseteq S/X_i} \frac{|S_i|!(M-|S_i|-1)!}{|M!} [f(S_i \cup X_i) - f(S_i)], i = 1, \dots, M,$$

где M – количество входных признаков; $S = \{X_i\}$ – множество всех наборов входных данных модели; $X_i = \{x_{ij} \mid j = 1, \dots, J_i\}$ – множество (набор) входных данных i -го признака, состоящее из J_i компонентов; $S_i \subseteq S \setminus X_i$ – подмножество, объединяющее наборы входных данных за вычетом данных i -го признака и объединяющее данные $|S_i|$ признаков; $f(S_i) = [f(x_{ij}) \mid X_i \subseteq S_i]$ – прогноз модели по входным данным x_{ij} , объединяемым подмножеством S_i .

Для оценки качества классификации помимо точности классификации были выбраны такие метрики, как ИП, ЛП, ИО, ЛО, а также значение интервала площади под PR-кривой (AUC-PR). Где:

ИП – истинно положительные решения;

ИО – истинно отрицательные решения;

ЛО – ложноотрицательные решения;

ЛП – ложноположительные решения.

Площадь под PR-кривой (AUC-PR), которая совмещает значения *Precision* и *Recall* в одной визуализации, широко используется для несбалансированных наборов данных, где один класс наблюдается чаще, чем другой класс [10]. Значения *Precision* и *Recall* рассчитывались по формулам:

$$Precision = \frac{ИП}{ИП + ЛП}$$

$$Recall = \frac{ИП}{ИП + ЛО}$$

3. РЕЗУЛЬТАТЫ

Для обоих наборов данных были использованы одинаковые параметры алгоритма и применялись одинаковые методы обработки. После извлечения из набора данных нужных признаков, была рассчитана корреляция между признаками и выходом модели по набору данных о походке (рис. 2) и набору данных о нажатиях клавиш (рис. 3). Поскольку значения стандартной деривации между левой и правой ступнями сильно коррелировали, были использованы значения только по одной правой ступне.

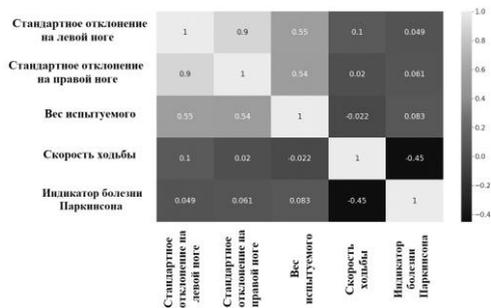


Рис. 2. Корреляция между признаками и выходом модели по набору данных о походке

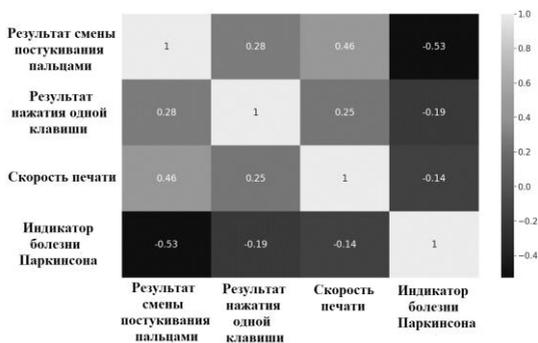


Рис. 3. Корреляция между признаками и выходом модели по набору данных о нажатиях клавиш

Была проведена бинарная классификация признаков болезни Паркинсона с использованием алгоритма градиентного бустинга после разделения набора данных на тестовые и обучающие выборки в соотношении 30% на 70%, соответственно (табл. 2). Для более детального анализа была рассчитана важность признаков для каждого набора данных (табл. 3). Параметры использованного компьютера приведены в табл. 4.

Таблица 2. Результаты классификации

Набор данных	AUC	Длительность обучения, мс	Точность	Матрица путаницы			
				ИП	ЛП	ИО	ЛО
Походка	0.92	2130	0.78	14	12	58	8
Нажатие клавиш	0.64	1650	0.65	8	7	9	2

Таблица 3. Средние абсолютные значения вектора Шепли (Shapley) для признаков

Набор данных	Название признака	Средние абсолютные значения вектора Шепли
Походка	Стандартное отклонение для общей силы, рассчитанной по всем датчикам на правой ноге	0.6
	Вес испытуемого	0.25
	Скорость ходьбы	1.3
Нажатие клавиш	Результат смены постукивания пальцами	1.7
	Результат нажатия одной клавиши	0.7
	Скорость печати	0.6

Таблица 4. Параметры использованного компьютера

Марка процессора	Частота процессора, ГГц	Число ядер	Оперативная память, ГБ
Intel(R) Xeon(R)	2.20	2	12

4. ЗАКЛЮЧЕНИЕ

Болезнь Паркинсона характеризуется множеством двигательных симптомов. У пациентов с БП отмечают двигательный дефицит, в том числе: нарушение походки, нарушение почерка и силы захвата. В большинстве исследований для диагностики БП используется анализ походки или анализ нажатия клавиш. Однако, ни в одном из исследований не представлено сравнительного анализа между этими моделями. В работе проведен сравнительный анализ диагностики болезни Паркинсона с помощью машинного обучения с двумя разными наборами данных, чтобы оценить эффективность каждой модели для диагностирования БП. Классификация признаков болезни Паркинсона по набору данных о походке показала лучшие результаты по сравнению с классификацией по набору данных о нажатиях клавиш. Точность классификации по этим наборам составила 78% и 65%, соответственно.

Растущая распространенность нейродегенеративных состояний, таких как болезнь Паркинсона, усилила острую потребность в более совершенных инструментах для удаленного управления хроническими заболеваниями. Цифровые медицинские технологии в форме удаленного мониторинга с помощью специализированных датчиков открывают широкие возможности для преобразования исследований и революционного

изменения клинического ведения БП для поддержки диагностики и мониторинга. Поэтому, в следующих работах будет рассмотрено использование этих датчиков для более эффективного анализа, что позволит собирать данные сразу за несколько недель, а иногда и месяцев, что положительно влияет на качество анализа. Кроме того, планируется рассмотреть использование методов интеллектуального анализа данных [11], таких как ДСМ-метод [12], методов вербального анализа решений [13], которые широко используются для ранней диагностики болезни Паркинсона [14].

Литература

1. Jankovic J. Parkinson's disease: clinical features and diagnosis // *Journal of neurology, neurosurgery & psychiatry*. – 2008. – № 79 (4). – P. 368-376.
2. Zesiewicz T.A., Sullivan K.L., Hauser R.A. Nonmotor symptoms of Parkinson's disease. Expert review of neurotherapeutics. – 2006. – № 6 (12). – P. 1811-1822.
3. Mei J., Desrosiers C., Frasnelli J. Machine learning for the diagnosis of parkinson's disease: A review of literature // *Frontiers in aging neuroscience*. – 2021. – № 13. – P. 184.
4. Postuma R.B. MDS clinical diagnostic criteria for Parkinson's disease // *Movement disorders*. – 2015. – № 30 (12). – P. 1591-1601.
5. Goldberger A. PhysioBank, PhysioToolkit and PhysioNet: Components of a new research resource for complex physiologic signals // *Circulation [Online]*. – 2000. – № 101 (23). – P. e215–e220.
6. Giancardo L. Computer keyboard interaction as an indicator of early Parkinson's disease // *Scientific Reports* 6. – 2016. – № 34468.
7. Хаддад Н., Мышенков К.С., Черненький С.В. Классификация признаков болезни Паркинсона с помощью методов машинного обучения // *Нейрокомпьютеры и их применение: Тезисы докладов XX Всеросс. научн. конф.* – М.: МГППУ, 2022. – С. 192-193.
8. Sindhanaiselvan K. Forecasting Chronic Disease using Gradient Boosting Algorithm // *International Journal of Engineering and Advanced Technology*. – 2019. – P. 1066-1070.
9. Mathivanan G. An Ensemble Basedheart Disease Predictionusing Gradient Boosting Decision Tree // *Turkish Journal of Computer and Mathematics Education*. – 2021. – № 10 – P. 3648-3660.
10. Sofaer H.R. The Area Under the Precision-recall Curve as a Performance Metric for Rare Binary Events // *Methods in Ecology and Evolution*. – 2019. – № 10 (4). – P. 565-577.
11. Вагин В.Н. Знания и убеждения в интеллектуальном анализе данных. – М.: Физматлит. – 2019. – 536 с.
12. Финн В.К. Интеллект, информационное общество и гуманитарное образование // *Гуманитарные чтения РГГУ – 2011. – М.: РГГУ, 2012. – С. 376-411.*
13. Ларичев О.И. Вербальный анализ решений. – М.: Наука, 2006. – 181 с.
14. Tucker C.A. Data Mining Methodology for Predicting Early Stage Parkinson's Disease using Non-Invasive, High-Dimensional Gait Sensor Data // *IEE Transactions on Healthcare Systems Engineering*. – 2015. – P. 238-254.

**МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ
ДЛЯ ПРОГНОЗИРОВАНИЯ РИСКА
ВНУТРИГОСПИТАЛЬНОЙ ЛЕТАЛЬНОСТИ ПОСЛЕ
КОРОНАРНОГО ШУНТИРОВАНИЯ***

*Шахгельдян К.И., д.т.н., профессор, директор
Институт информационных технологий ВГУЭС
зав. лабораторией
ДВФУ*

e-mail: carinash@vvsu.ru

*Рублев В.Ю., врач сердечно-сосудистый хирург, ассистент
ДВФУ*

e-mail: dr.rublev.v@gmail.com

*Гельцер Б.И., д.м.н., профессор, чл.-корр. РАН,
зам. директор по науке
Школа медицины ДВФУ*

e-mail: boris.geltser@vvsu.ru

*Щеглов Б.О., студент
Школа Медицины ДВФУ
e-mail: b.shcheglov@mail.ru*

*Костерин В.В., студент
Институт информационных технологий ВГУЭС
e-mail: kraiaisherdev@gmail.com*

*Щеглова С.Н., к.пед.н., доцент
СВГУ*

e-mail: schegloff@mail.ru

1. ВВЕДЕНИЕ

Медицина является одной из наиболее приоритетных областей для разработки и применения интеллектуального анализа данных и технологий искусственного интеллекта. С одной стороны эта область, в которой накоплены большие объемы знаний, формализация которых позволяет разрабатывать интеллектуальные системы. С другой – здесь генерируется

* Работа выполнена при финансовой поддержке РФФИ, проект №19-29-01077 и гранта УМНИК по договору № 16640ГУ/2021.

большие объемы клинических данных, позволяющие с помощью методов интеллектуального анализа данных, извлекать и верифицировать новые знания о развитии и течении заболеваний, персонализированной реакции организма и разрабатывать новые прогностические и диагностические модели развития заболеваний.

Традиционно в клинической практике используются прогностические шкалы, разработанные на основе больших массивов данных, используя классическую математическую статистику и многофакторную логистическую регрессию. С развитием современных методов интеллектуального анализа данных, удастся усовершенствовать такие шкалы и с одной стороны выделить и верифицировать новые предикторы развития заболеваний и их осложнений, с другой – разработать более точные прогностические модели и объяснить отдельные аспекты причин развития заболеваний или осложнений.

Ишемическая болезнь сердца (ИБС) является одной из основных патологий недостаточности кровоснабжения сердца, которой страдают более 126 миллионов человек в мире [1]. Одним из методов ее лечения является аорто-коронарное шунтирование [2–4]. Это хирургическое вмешательство подразумевает проведение ремоделирования стенозированных коронарных артерий путем замены на сосуды, сформированные чаще всего из бедренной вены. В настоящее время операция коронарного шунтирования (КШ) относится к рутинным, однако в ближайший послеоперационный период возможны возникновения осложнений, вплоть до летальности, частота которой составляет 3–6% случаев [5–7].

В современной клинической практике для прогнозирования используются шкалы, разработанные с помощью многофакторной логистической регрессии (МЛР): Euroscore и Euroscore II, The Society of Thoracic Surgeons, NATCH [8–11]. В последние годы ученые активно применяют другие модели машинного обучения для прогнозирования ближайших и долгосрочных исходов КШ. Так, была показана эффективность применения алгоритмов стохастического градиентного бустинга (СГБ), случайного леса (СЛ) и деревьев решений (ДР) [12–15].

Исследователи Huang, Y. С и соавт. [16] на основе СГБ разработали модель для прогнозирования риска летальности пациентов в 10-летнем периоде с точностью на уровне 0.71 т площадью под ROC кривой (AUC) 0.77. В исследовании Arman Kilic и соавт. [17] были представлены результаты моделирования на основе СГБ с AUC=0,808. В работе José Castela Forte и соавт. [18] проводят поиск предикторов, влияющих на риск возникновения смерти в течение 5 лет после проведения КШ, посредством использования алгоритма Super Learner (SL) удалось достигнуть специфичности на 0,89 и чувствительности 0,479.

Исследователями Suveen Angraal и соавт. [19] были разработаны модели для оценки риска смертности и госпитализации по поводу сердечной недостаточности (СН) в течение 3 лет на основе следующих методов: МЛР, СЛ, СГБ и SVM. Наилучшие метрики (С-статистика 0,72) были получены при использовании алгоритма СЛ.

В работе Arinze Nkemdirim Okere и соавт. [20] была рассчитана модель на базе ЛР, ДР и СЛ для определения полугодовой летальности пациентов с ИБС в кардиохирургическом отделении. Наилучшая модель была представлена при использовании метода ДР с точностью и AUC на уровне 99%.

Соответственно, анализ данных литературных источников показывает чрезвычайную актуальность данной темы исследования, высокую точность прогнозирования отдаленных результатов КШ, но отсутствие устойчивых клинико-диагностических критериев в разработанных моделях на краткосрочный промежуток времени, что сакцентировало внимание авторов на рассматриваемой проблеме и для более ее углубленного изучения [6, 7, 21, 22]. Целью исследования является создание собственной прогностической шкалы, позволяющей на дооперационном периоде на базе рассчитанных предикторов предсказывать возникновения внутригоспитальной летальности (ВГЛ) у пациентов с ИБС после КШ.

2. МАТЕРИАЛЫ И МЕТОДЫ

Ретроспективное исследование было выполнено на основе данных электронных историй болезней (ЭИБ) 999 больных ИБС, которым было проведено изолированное КШ в период с 2008 по 2019 гг. на базе кардиохирургического отделения ГБУЗ «Приморская краевая клиническая больница №1» г. Владивостока. Среди обследованной когорты было выделено 2 группы лиц. В первую из них вошли 63 (6,3%) больных, умерших в послеоперационном периоде в течение 2 месяцев (63 дней), во вторую – 936 (93,6%) больных, оставшихся живыми в течение первых 2 месяцев после операции.

Результирующей переменной была определена ВГЛ в форме категориального признака ("жив" или "умер"). В качестве входных данных использовалось 100 до операционных признаков, включавших в себя демографические, антропометрические данные, данные лабораторных и инструментальных исследований, анамнеза пациентов. Кроме этого было рассчитано дополнительно порядка 50 факторов, в том числе индекс массы тела (ИМ), клиренс креатинина (КК), индексы относительной толщины (ИОТ) и массы миокарда левого желудочка (ОИММ) и др.

Исследование данных проводилось с помощью методов статистического анализа и машинного обучения. Первые из них включали тесты хи-квадрат, Фишера, Манна-Уитни и однофакторная ЛР с расчетом весовых коэффициентов. Вторые – многофакторная ЛР, дерево решений (ДР), случайный лес (СЛ), стохастический градиентный бустинг (СГБ) и искусственная нейронной сетью (ИНС). Архитектура ИНС, представляющая собой многослойный перцептрон, подбиралась путем максимизации площади под ROC-кривой (AUC) и в окончательном виде состояла из двух скрытых слоев по 90 и 80 нейронов в каждом. В качестве функции активации ИНС использовались «sigmoid» и «relu». В окончательные модели с учетом лучшей точности ИНС вошла только функция «sigmoid». Точность моделей оценивали по 4 метрикам качества: AUC, чувствительность (Sen), специфичность (Spec), точность (Acc) и полнота (f1-score). Процедура балансировки чувствительности и специфичности включала подбор такой оптимальной границы (threshold), при которой эти две метрики близки между собой. Разработка моделей выполнялась с помощью процедуры кросс-валидации методом стратифицированного k-фолдинга. Модели обучались на выборке из 9/10 пациентов и верифицированы на тестовой (1/10). Значения метрик качества моделей, приведенные в работе, получены путем усреднения оценок прогноза на тестовых выборках.

Дизайн исследования состоял из 4 этапов. На первом из них использовали методы статистического анализа, с помощью которого проводили межгрупповые сравнения потенциальных предикторов послеоперационной смертности. Для непрерывных переменных применялся тест Манна-Уитни, так как предварительная оценка близости данных к нормальному распределению методом Шапиро-Уилка показала отрицательный результат. Для сравнения категориальных переменных использовали критерий хи-квадрат, а для оценки отношения шансов (ОШ) и их доверительных интервалов (ДИ) – тест Фишера. Рассматривая гипотезу о том, что в клинической медицине фатальные (и нефатальные) события возникают при выходе значений показателей за некоторые критерии (пороговые значения), на втором этапе проводили категоризацию непрерывных факторов с помощью статистических методов путем детектирования границ значений анализируемых факторов с наилучшим предиктивным потенциалом. Рассматриваемая процедура включала проверку гипотез о равенстве распределений признаков в группах сравнения. Обнаружение прогностически значимых диапазонов значений осуществлялось с шагом тестирования 0,01 усл. ед. от минимального до максимального значения показателей. Критериям отбора соответствовали оптимальные границы значений факто-

ров, p-value которых имели минимальную, а ОШ – максимальную величину. На третьем этапе исследования по нормированным признакам с помощью однофакторной ЛР определяли весовые коэффициенты, соответствующие значимости влияния отдельных признаков на развитие внутригоспитальной смертности. На четвертом этапе были разработаны многофакторные модели на основе ЛР, СЛ, СГБ и ИНС, структура которых пошагово дополнялась потенциальными предикторами внутригоспитальной летальности с оценкой метрик качества. При увеличении значений последних считали, что включенный в модель показатель может рассматриваться в качестве предиктора послеоперационной смертности. Обработка и анализ данных выполнялись на языке Python.

3. РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ

Сравнительный анализ межгрупповых различий 100 дооперационных и 50 рассчитанных показателей показывает, что статистически значимыми различиями являлись: возраст (ВЗ); эхокардиографические параметры сердца - поперечный (LA2) размер левого предсердия (ЛП), продольный (RA1) и поперечный (RA2) размеры правого предсердия (ПП) и соответствующая “площадь” (RA1×RA2); срочность проведения оперативного вмешательства (СО); наличие сочетанной недостаточности трикуспидального и аортального клапанов (НТК и НАК) в анамнезе; недавний (до 4 месяцев) инфаркт миокарда (ИМ); фракция выброса левого желудочка (ФВ); СДЛА (СДЛА); электрокардиографические (ЭКГ) признаки - длительность сегмента QT и интервала RR; признак тяжелой ХСН: III-IV функциональный класс (ФК); наличие экстракардиальной артериопатии (ЭКА) в анамнезе; ТВ (ТВ); индекс относительной толщины (ИОТ) и относительный индекс массы миокарда (ОИММ); клиренс креатинина (КК) (табл. 1). Все факторы, имеющие статистически значимые различия, а также несколько других приведены в табл. 1.

Таблица 1. Клинико-функциональная характеристика больных

Показатели	Объем выборки	1 группа, n = 63	2 группа, n = 936	p-value
ВЗ, лет	999	67 ± 7,6	63 ± 7,5	0,00003
RA1×RA2, см ²	927	20 ± 4,9	16 ± 4,8	< 0,000001
СО	994	6 (9,5%)	57 (6,1%)	0,008

НТК и НАК	999	7 (11,1%)	56 (6%)	0,00007
ИМ	977	28 (44,4%)	35 (3,7%)	< 0,000001
ФВ, %	977	51 ± 11,3	58 ± 9,3	<0,000001
СДЛА	919	30 ± 10,9	26 ± 8,3	0.0119
RR, мс	999	973 ± 238,1	893 ± 271,2	0.0000024
ХСН	990	21 (33,3%)	42 (4,5%)	0,0003
ЭКА	974	31 (49%)	32 (4,8%)	0,005
ТВ, с	807	18 ± 5,5	19 ± 5,3	0.0019
LA2, мм	927	47 ± 7,3	42 ± 7,0	0,026
ИОТ, с	924	0.42 ± 0,1	0.42 ± 0,09	0.42
QT, мс	999	387 ± 35	391 ± 39	0.24
Базофилы, %	907	27 ± 8,3	47 ± 15,4	0.37
КК, мл / мин	811	65 ± 21.5	75.8 ± 22	< 0,000001
ОИММ	833	1.11 ± 0,33	1.05 ± 0,28	0.16

На втором этапе исследования были рассчитаны оптимальные диапазоны всех показателей, обладающие наибольшим предиктивным потенциалом (табл. 2). Полуавтоматизированная функция поиска и подбора оптимальных границ непрерывных переменных позволила рассчитать не только значимые диапазоны предикторов и определить факторы риска, но и определить дополнительные предикторы, которые ранее не были таковыми верифицированы (QT и ИОТ).

Лучший прогностический потенциал по сравнению со своими непрерывными аналогами показали факторы ТВ, при значении которого ниже 13 значительно увеличиваются риски ВГЛ (ОШ=33,3, p-value < 0,000001), LA2 (ОШ=5,9, p-value < 0,000001), ОИММ (ОШ=4,1, p-value=0,013), ИОТ (ОШ = 59, p-value=0,000007), СДЛА (ОШ=5,3, p-value < 0,00001), QT (ОШ=3,3, p-value=0,03). В тоже время, часть показателей в непрерывной форме имела лучшую статистическую значимость, чем их категориальный аналог (Вз, RA1×RA2, ФВ).

Таблица 2. Границы значений факторов риска послеоперационной летальности с наилучшим предиктивным потенциалом

Предиктор и пороги	ОШ, 95% ДИ	p-value
Вз > 67 лет	2,75 [1.64; 4.6]	0,0001
RA1 * RA2 > 15 см ²	5,04 [2.46; 10.35]	0,000002
СО	3,81 [1.5-9.67]	0,008
НТК и НАК	6,03 [2.43-14.95]	0,00007
ИМ	6,66 [3.84-11.11]	< 0,000001
ФВ < 45%	3,85 [2.17-7.14]	0,000003
СДЛА > 39 мм рт.ст.	5,28 [2.8-9.94]	< 0,000001
Длительность интервала RR > 1800 мс	6,07 [2.34-15.05]	0,00006
ХСН	2,81 [1.62-4.89]	0,0003
ЭКА	2,12 [1.27-3.54]	0,005
ТВ < 13 с	33,3 [8.33-100]	< 0,000001
LA2 > 4.1 см	5,88 [2.86-12.06]	0,0000001
ИОТ < 0,21 с	59 [4.55-100]	0,000007
QT < 340 мс	3,3 [1.23-9,09]	0,03
Базофилы > 12%	0.1 [0-0.99]	0.07
ОИММ > 1.71	4,14 [1.46-11.71]	0,013

На третьем этапе исследования для разработки моделей ЛР, СГБ, СЛ и ИНС поочередно подбирались предикторы в непрерывной или в категориальной форме. Также в таблицах были оценены эффективность использования предикторов в категориальной форме по

сравнению с их непрерывными аналогами. Были также рассмотрены случаи смешанного использования данных в категориальном и непрерывном виде.

Таблица 3. Оценка точности прогностических моделей послеоперационной смертности на тестовых выборках для МЛР

N	Предикторы	Метрики точности		
		Sen	Spec	AUC
1	Вз	0,62	0,62	0,65
2	Вз, RA1×RA2	0,67	0,68	0,74
3	Вз, RA1×RA2, СО	0,69	0,69	0,75
4	Вз, RA1×RA2, СО, НТК и НАК	0,69	0,69	0,75
5	Вз, RA1×RA2, СО, НТК и НАК, ИМ	0,75	0,75	0,83
6	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ	0,76	0,76	0,84
7	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ, СДЛА	0,76	0,76	0,83
8	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ, СДЛА, RR мс	0,77	0,77	0,84
9	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ, СДЛА, RR мс, ХСН	0,77	0,78	0,84
10	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ, СДЛА, RR мс, ХСН, ЭКА	0,78	0,79	0,85
11	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ, СДЛА, RR мс, ХСН, ЭКА, ТВ	0,78	0,79	0,84
12	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ, СДЛА, RR мс, ХСН, ЭКА, ТВ, LA2	0,79	0,79	0,85
13	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ, СДЛА, RR мс, ХСН, ЭКА, ТВ, LA2, ОИММ	0,78	0,79	0,85
14	<i>Вз > 67 лет</i>	0,54	0,70	0,62
15	<i>Вз > 67 лет, RA1×RA2 > 15 см2</i>	0,69	0,68	0,73
16	<i>Вз > 67 лет, RA1×RA2 > 15 см2, СО</i>	0,68	0,70	0,73
17	<i>Вз > 67 лет, RA1×RA2 > 15 см2, СО, НТК и НАК</i>	0,68	0,71	0,74
18	<i>Вз > 67 лет, RA1×RA2 > 15 см2, СО, НТК и НАК, ИМ</i>	0,72	0,73	0,81
19	<i>Вз > 67 лет, RA1×RA2 > 15 см2, СО, НТК и НАК, ИМ, ФВ < 45%</i>	0,76	0,75	0,83
20	<i>Вз > 67 лет, RA1×RA2 > 15 см2, СО, НТК и НАК, ИМ, ФВ < 45%, СДЛА > 39 мм рт.ст.</i>	0,75	0,75	0,83

21	Вз> 67 лет, RA1×RA2 > 15 см2, СО, НТК и НАК, ИМ, ФВ<45%, СДЛА> 39 мм рт.ст., RR> 1800 мс	0,75	0,75	0,83
22	Вз> 67 лет, RA1×RA2 > 15 см2, СО, НТК и НАК, ИМ, ФВ<45%, СДЛА> 39 мм рт.ст., RR> 1800 мс, ХСН	0,79	0,78	0,84
23	Вз> 67 лет, RA1×RA2 > 15 см2, СО, НТК и НАК, ИМ, ФВ<45%, СДЛА> 39 мм рт.ст., RR> 1800 мс, ХСН, ЭКА	0,79	0,79	0,85
24	Вз> 67 лет, RA1×RA2 > 15 см2, СО, НТК и НАК, ИМ, ФВ<45%, СДЛА> 39 мм рт.ст., RR> 1800 мс, ХСН, ЭКА, ТВ < 13 с	0,80	0,80	0,86
25	Вз> 67 лет, RA1×RA2 > 15 см2, СО, НТК и НАК, ИМ, ФВ<45%, СДЛА> 39 мм рт.ст., RR> 1800 мс, ХСН, ЭКА, ТВ < 13 с, LA2 > 4.1 см	0,80	0,79	0,86
26	Вз> 67 лет, RA1×RA2 > 15 см2, СО, НТК и НАК, ИМ, ФВ<45%, СДЛА> 39 мм рт.ст., RR> 1800 мс, ХСН, ЭКА, ТВ < 13 с, LA2 > 4.1 см, ОИММ > 1.71	0,80	0,79	0,86
27	Вз> 67, RA1×RA2 > 15, СО, НТК и НАК, ИМ, ФВ < 45, СДЛА > 39, RR мс, ХСН, ЭКА, ТВ < 13, LA2 > 4.1, ОИММ > 1.7	0,80	0,80	0,87
28	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ < 45, СДЛА > 39, RR мс > 1800.0, ХСН, ЭКА, ТВ < 13, LA2 > 4.1, ОИММ > 1.7	0,80	0,81	0,86
29	RA1×RA2, КК, ИМ, ХСН, ЭКА	0,81	0,8	0,84

Повышение метрик качества МЛР фиксировалось в моделях 2-12 (табл. 3). Максимальная точность по метрике Sen (0,79) была достигнута в модели 12 при AUC 0,85. Значение Spes составило 0,79. Последующее добавление ОИММ снижало метрику Sen. Анализ моделей на основе МЛР, использующих категориальные факторы (модели 14-26 в табл. 3), показал, что точность прогноза на основе моделей с категориальными факторами увеличивалась аналогично моделям с аналогичными непрерывными показателями, качество при этом было несколько выше. В частности, сравнение моделей 12 и 24, демонстрирует, что последняя обладает более лучшими параметрами качества по 4 метрикам качества (0,8 vs 0,79, 0,8 vs 0,79, 0,86 vs 0,85 и 0,8 vs 0,79). Последующее включение категориальных факторов не меняло точ-

ность прогноза (модели 25-26). Были рассмотрены и модели со смешанным типами факторов (модели 27-28), которые показывали лучшее качество прогноза. В тоже время, были получены модели с сопоставимыми метриками качества, но с другим набором параметров, где вместо возраста использовался показатель КК (модель 27).

Модели на основе СГБ, СЛ и ИНС были построены аналогичным образом. Наилучшее качество моделей достигалось при схожем наборе предикторов с небольшими вариациями (табл. 4). Лучшее качество моделей получено методом СЛ на смешанном наборе показателей.

Таблица 4. Лучшие модели МЛР, СЛ, СГБ, ИНС

Алгоритм	Предикторы модели	Метрики точности		
		Sen	Spec	AUC
МЛР	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ < 45, СДЛА > 39, RR мс > 1800.0, ХСН, ЭКА, ТВ < 13, LA2 > 4.1, ОИММ > 1.7	0,80	0,80	0,87
СЛ	Вз, RA1×RA2, СО, НТК и НАК, ИМ, ФВ < 45, СДЛА > 39, RR мс > 1800.0, ХСН, ЭКА, ТВ < 13, LA2 > 4.1, ИОТ > 0.21, QT < 340	0,80	0,81	0,87
СГБ	Вз > 67, RA1×RA2 > 15, СО, НТК и НАК, ИМ, ФВ < 45, СДЛА > 39, RR мс > 1800.0, ХСН, базофилы	0,76	0,76	0,85
ИНС	Вз > 67, RA1×RA2 > 15, СО, НТК и НАК, ИМ, ФВ < 45, СДЛА > 39, RR мс > 1800.0, ХСН, ЭКА, ТВ < 13, LA2 > 4.1, ИОТ > 0.21, QT < 340	0,80	0,79	0,86
СГБ	ИМ, RA1×RA2, СДЛА, QRS, ИМТ, КК	0,88	0,77	0,85

4. ЗАКЛЮЧЕНИЕ

В данном исследовании представлены модели ВГЛ после КШ, разработанные на основе методов статистического анализа и машинного обучения: МЛР, СГБ, СЛ и ИНС. Пошаговый отбор предикторов

обеспечил создание разными методами моделей с сопоставимым качеством прогноза. Для интерпретации моделей были определены пороговые значения показателей, которые обеспечивают наилучший предиктивный потенциал. Использование таких категориальных показателей позволяет не только интерпретировать модели на понятном для клинического врача языке, но и повышает качество прогноза. В тоже время, наше исследование показало, что включение в модель показателей в смешанной форме, обеспечивает лучшее качество моделей.

Рассчитанные в ходе работы предикторы могут быть интерпретированы с актуальных позиций о патофизиологии ИБС и риска возникновения послеоперационных кардиогенных морбидных состояний. При этом наличие рубцовых изменений в сердце, нарушенная насосная функция и размеры сердца, изменения электрофизиологических и коагуляционных параметров в предоперационном периоде могут стать триггерами к возникновению жизненно-опасных осложнений и в итоге летального исхода. С уверенностью можно утверждать, что в данном моделировании часть предикторов являются базовыми параметрами состояния сердечно-сосудистой системы пациентов и их влияние на риск возникновения неблагоприятных исходов продемонстрировано в ряде исследований. Более того, были обнаружены и ранее не используемые предикторы, такие как, сочетанная недостаточность трикуспидального и аортального клапанов, размеры ПП, сниженное ТВ, что необходимо верифицировать в последующих исследованиях.

Литература

1. Khan M.A., Hashim M.J., Mustafa H., Baniyas M.Y., Al Suwaidi S., AlKatheeri R., Alblooshi F., Almatrooshi M., Alzaabi M., Al Darmaki R.S., & Lootah S. Global Epidemiology of Ischemic Heart Disease: Results from the Global Burden of Disease Study. *Cureus*. – (2020). – 12 (7), – e9349. – <https://doi.org/10.7759/cureus.9349>.
2. Neumann F.J., Sousa-Uva M., Ahlsson A., Alfonso F., Banning A.P., Benedetto U., Byrne R.A., Collet J.P., Falk V., Head S.J., Juni P., Kastrati A., Koller A., Kristensen S.D., Niebauer J., Richter D.J., Seferovic P.M., Sibbing D., Stefanini G.G., Windecker S., ... ESC Scientific Document Group // 2018 ESC/EACTS Guidelines on myocardial revascularization. *European heart journal*. – 2019. – 40 (2). – P. 87-165. – <https://doi.org/10.1093/eurheartj/ehy394>.
3. Российское кардиологическое общество (РКО). Стабильная ишемическая болезнь сердца. Клинические рекомендации 2020. *Российский кардиологический журнал*. – 2020. – 25 (11):4076. – <https://doi.org/10.15829/29/1560-4071-2020-4076>.
4. SEC Working Group for the 2018 ESC/EACTS Guidelines for Myocardial Revascularization, Expert Reviewers for the 2018 ESC/EACTS Guidelines for Myocardial Revascularization, & SEC Guidelines Committee (2019). Comments on the 2018 ESC/EACTS Guidelines for Myocardial Revascularization. *Revista espanola de cardiologia (English ed.)*, 72(1), 16–20. <https://doi.org/10.1016/j.rec.2018.11.010>.

5. Карпов Ю.А., Барбараш О.Л., Бощенко А.А., Кашгалап В.В., Кухарчук В.В., Миронов В.М., Панченко Е.П., Руда М.М., Самко А.Н., Соболева Г.Н., Ширяев А.А. Евразийские клинические рекомендации по диагностике и лечению стабильной ишемической болезни сердца (2020-2021). Евразийский кардиологический журнал. – 2021. – (3):54-93. – <https://doi.org/10.38109/2225-1685-2021-3-54-93>.
6. Гельцер Б.И., Шахгельдян К.И., Рублев В.Ю. [и др.]. Алгоритм отбора предикторов и прогнозирование фибрилляции предсердий у больных ишемической болезнью сердца после коронарного шунтирования // Российский кардиологический журнал. – 2021. – Т. 26. – № 7. – С. 40-47. – DOI: 10.15829/1560-4071-2021-4522.
7. Geltser B.I., Rublev V.Y., Shakhgeldyan K.I. [et al.]. Machine learning models for atrial fibrillation prediction after coronary artery bypass graft surgery // CEUR Workshop Proceedings: 10, Kolomna, 17–20 мая 2021 года. – Kolomna, 2021. – P. 311-323.
8. Strilchuk L., Besh D. Estimation of cardiosurgical intervention risk according to EuroSCORE in patients with acute coronary syndrome and different gallbladder conditions before coronary artery bypass grafting. *Kardiochir Torakochirurgia Pol.* – 2018. – 15 (4). – P. 238-240. – Doi:10.5114/kitp.2018.80920.
9. Nashef S.A., Roques F., Michel P., Gauducheau E., Lemeshow S., & Salamon R. European system for cardiac operative risk evaluation (EuroSCORE) // *European journal of cardio-thoracic surgery : official journal of the European Association for Cardio-thoracic Surgery.* – 1999. – 16 (1). – P. 9-13. – [https://doi.org/10.1016/s1010-7940\(99\)00134-7](https://doi.org/10.1016/s1010-7940(99)00134-7).
10. Vassileva C.M., Aranki S., Brennan J.M., Kaneko T., He M., Gammie J.S., Suri R.M., Thourani V.H., Hazelrigg S., & McCarthy P. // Evaluation of The Society of Thoracic Surgeons Online Risk Calculator for Assessment of Risk in Patients Presenting for Aortic Valve Replacement After Prior Coronary Artery Bypass Graft: An Analysis Using the STS Adult Cardiac Surgery Database. *The Annals of thoracic surgery.* – 2015. – 100 (6). – P. 2109-2116. – <https://doi.org/10.1016/j.athoracsur.2015.04.149>.
11. Suenari K., Chao T.F., Liu C.J., Kihara Y., Chen T.J., & Chen S.A. Usefulness of HATCH score in the prediction of new-onset atrial fibrillation for Asians // *Medicine.* – 2017. – 96 (1). – e5597. – <https://doi.org/10.1097/MD.0000000000005597>.
12. Mpanya, D., Celik, T., Klug, E., & Ntsinjana, H. Machine learning and statistical methods for predicting mortality in heart failure // *Heart failure reviews.* – 2021. – 26 (3). – P. 545-552. – <https://doi.org/10.1007/s10741-020-10052-y>.
13. Shin S., Austin P.C., Ross H.J., Abdel-Qadir H., Freitas C., Tomlinson G., Chicco D., Mahendiran M., Lawler P.R., Billia F., Gramolini A., Epelman S., Wang B., & Lee D.S. Machine learning vs. conventional statistical models for predicting heart failure readmission and mortality // *ESC heart failure.* – 2021. – 8 (1). – P. 106-115. – <https://doi.org/10.1002/ehf2.13073>.
14. Sajeev S., Champion S., Beleigoli A., Chew D., et al. Predicting Australian Adults at High Risk of Cardiovascular Disease Mortality Using Standard Risk Factors and Machine Learning // *International journal of environmental research and public health.* – 2021. – 18 (6), 3187. – <https://doi.org/10.3390/ijerph18063187>.

15. Karim M.N., Reid C.M., Cochrane A., Tran L., et al. Mortality risk prediction models for coronary artery bypass graft surgery: current scenario and future direction // *The Journal of cardiovascular surgery*. – 2017. – 58 (6). – P. 931-942. – <https://doi.org/10.23736/S0021-9509.17.09965-7>.
16. Huang Y.C., Li S.J., Chen M., Lee T.S., & Chien Y.N. Machine-Learning Techniques for Feature Selection and Prediction of Mortality in Elderly CABG Patients // *Healthcare (Basel, Switzerland)*. – 2021. – 9 (5). – 547. – <https://doi.org/10.3390/healthcare9050547>.
17. Kilic A., Goyal A., Miller J.K., Gjekmarkaj E., Tam W.L., Gleason T.G., Sultan I., & Dubrawski A. Predictive Utility of a Machine Learning Algorithm in Estimating Mortality Risk in Cardiac Surgery // *The Annals of thoracic surgery*. – 2020. – 109 (6). – P. 1811-1819. <https://doi.org/10.1016/j.athoracsur.2019.09.049>.
18. Castela Forte J., Mungroop H.E., de Geus F., van der Grinten M.L., Bouma H.R., Pettilä V., Scheeren T., Nijsten M., Mariani M.A., van der Horst I., Henning R.H., Wiering M.A., & Epema A.H. Ensemble machine learning prediction and variable importance analysis of 5-year mortality after cardiac valve and CABG operations // *Scientific reports*. – 2021. – 11 (1). – 3467. – <https://doi.org/10.1038/s41598-021-82403-0>.
19. Angraal S., Mortazavi B.J., Gupta A., Khera R., et al. Machine Learning Prediction of Mortality and Hospitalization in Heart Failure With Preserved Ejection Fraction // *JACC. Heart failure*. – 2020. – 8 (1). – P. 12-21. – <https://doi.org/10.1016/j.jchf.2019.06.013>.
20. Okere A.N., Sanogo V., Alqhtani H., & Diaby V. Identification of risk factors of 30-day readmission and 180-day in-hospital mortality, and its corresponding relative importance in patients with Ischemic heart disease: a machine learning approach // *Expert review of pharmacoeconomics & outcomes research*. – 2021. – 21 (5). – P. 1043-1048. – <https://doi.org/10.1080/14737167.2021.1842200>.
21. Рублев В.Ю., Сергеев Е.А., Гельцер Б.И. Информативность гемодинамических индикаторов у пациентов с ишемической болезнью сердца в прогнозировании результатов аорто-коронарного шунтирования // *Тихоокеанский медицинский журнал*. – 2020. – № 1 (79). – С. 16-22. – DOI: 10.34215/1609-1175-2020-1-16-22.
22. Свидетельство о государственной регистрации программы для ЭВМ № 2021662443 Российская Федерация. Программа для отбора предикторов и прогнозирования фибрилляции предсердий у больных ишемической болезнью сердца после коронарного шунтирования: № 2021618886: заявл. 07.06.2021; опубл. 29.07.2021 / Б.И. Гельцер, К.И. Шахгельдян, В.Ю. Рублев, Б.О. Щеглов; заявитель Федеральное государственное бюджетное образовательное учреждение высшего образования «Владивостокский государственный университет экономики, и сервиса».

СЕКЦИЯ 3

**БИОИНСПИРИРОВАННЫЕ
ПОДХОДЫ, ЭВОЛЮЦИОННОЕ
МОДЕЛИРОВАНИЕ,
ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ**

УДК 519.712.2

**РАЗРАБОТКА И ИССЛЕДОВАНИЕ ГИБРИДНОГО
АЛГОРИТМА РЕШЕНИЯ ЗАДАЧИ РАЗМЕЩЕНИЯ
ЭЛЕМЕНТОВ ЭВА**

*Веселов Г.Е., д.т.н., доцент, директор
ИКТИБ Южный федеральный университет
e-mail: gev@sfedu.ru*

*Гладков Л.А., к.т.н., профессор
Южный федеральный университет
e-mail: leo_gladkov@mail.ru*

*Ясир М.Д., аспирант
Южный федеральный университет
e-mail: yasir_82@mail.ru*

1. ВВЕДЕНИЕ

Современный процесс проектирования невозможно представить без применения вычислительной техники. Это связано с размерностью решаемых задач. Сложность проектируемых устройств, число содержащихся элементов и связей между ними непрерывно возрастает. В настоящее время речь идет об устройствах содержащих порядка 10^4 элементов. Решать возникающие задачи такого уровня сложности «в лоб» за счет полного перебора всех возможных вариантов существующие вычислительные устройства пока не позволяют. Поэтому по-прежнему актуальной остается задача сокращения перебора и поиск

эффективной архитектуры и эвристических методов оптимизации, позволяющих получать приемлемые по качеству решения за полиномиальное время. На некоторых этапах проектирования отсутствуют эффективные детерминированные вычислительные методы, что не позволяет решить задачу выполнения проектирования в полностью автоматическом режиме.

Совершенствование технологий изготовления привело к пересмотру некоторых принципов, на которых основан процесс физического проектирования. Это в свою очередь привело к пересмотру актуальных критериев оценки качества размещения. Использувавшиеся ранее критерии, построенные на модели оценки временных задержек пропорционально суммарной длине соединений стали неактуальными, изменилась концепция процесса физического проектирования вычислительных устройств.

Одним из основных трендов современной науки является гибридизация, т.е. создание новых систем, моделей и методов, подразумевающих интеграцию научных подходов различной физической природы. Особенно эффективно использование такого рода систем для решения сложных слабо формализованных задач, в том числе задач проектирования и оптимизации [1–3].

2. ПОСТАНОВКА ЗАДАЧИ

Рассматриваемая в данной работе задача размещения относится к задачам конструкторского (или физического) проектирования (*physical design*). После решения задачи компоновки (разбиения) схемы можно выполнить оценочный расчет площади, занимаемой каждым блоком и определить необходимое число контактов (выводов). Чтобы перейти к решению следующей задачи – трассировки соединений, проектировщику нужно расположить блоки на поверхности кристалла. При этом блокам присваивается определенная форма, и они располагаются на поверхности макета таким образом, чтобы любые два блока не имели взаимного перекрытия, и на поверхности коммутационного поля оставалось достаточно места для проведения соединений. Блоки должны быть размещены таким образом, чтобы минимизировать общую площадь макета. Отдельно определяется расположение контактов в каждом блоке. Исходными данными для задачи размещения является набор блоков, число контактов для каждого блока и список соединений.

Задача размещения элементов является одной из важнейших задач конструкторского этапа проектирования и состоит в определении оптимального пространственного расположения элементов на коммутационном поле [4–6]. В процессе решения строится пространственная модель расположения элементов, которая является основой выполнения всех по-

следующих задач проектирования. Другими словами, общее качество проектирования схемы с точки зрения площади и производительности в основном определяется в процессе решения задачи размещения.

Одним из подходов к решению задачи снижения общих затрат вычислительных ресурсов и времени в процессе проектирования является блочно-иерархический подход. Данный подход позволяет разбить процесс проектирования на этапы и произвести декомпозицию объекта проектирования, что дает возможность снизить итоговую сложность и размерность решаемых задач [4–6].

Важным фактором, влияющим на качество размещения, является форма блоков (элементов). Для упрощения задачи предполагается, что все блоки имеют прямоугольную форму. Как правило, при планировании кристалла для определения формы блока используют соотношение сторон, т.е. отношение высоты блока к его ширине. При этом задают верхнюю и нижнюю границы блока. Как правило, это прямоугольная область, границы которой задаются координатами угловых точек $(x_{low}, y_{low}, x_{high}, y_{high})$. В настоящее время используются и другие, например *L*- или *T*-образные формы, однако работа с элементами такой формы требует значительных вычислительных ресурсов.

Задача размещения состоит в задании такого расположения элементов схемы на рабочем поле, которое приводит к минимизации общей длины соединений схемы. Это связано с тем, что длина соединений может быть легко смоделирована и служит хорошим приближением первого порядка к реальным целевым функциям, таким как синхронизация, величина энергопотребления и т.д. Длина соединений также влияет на качество решения задачи трассировки. Также при размещении необходимо проводить оценку свободных зон и следить, чтобы их площадь была достаточной для последующей трассировки. Блоки должны размещаться таким образом, чтобы между ними оставалось достаточно места для проведения соединений. То есть в процессе размещения необходимо также учитывать возможность трассировки.

Пусть B_1, B_2, \dots, B_n – это блоки (элементы), которые должны быть размещены на коммутационном поле. Для каждого элемента B_i , $1 \leq i \leq n$, задана его высота h_i и ширина w_i . Также задан список цепей, отражающих взаимосвязи между элементами $N = \{N_1, N_2, N_3, \dots, N_m\}$ и $Q = \{Q_1, Q_2, \dots, Q_k\}$ – множество прямоугольных пустых областей, выделенных для трассировки. Обозначим через L_i оценочную длину цепи N_i , $1 \leq i \leq m$. Цель размещения состоит в том, чтобы найти для каждого элемента из набора B позицию на рабочем поле с учетом ориентации из множества $R = \{R_1, R_2, \dots, R_n\}$, чтобы при этом соблюдались следующие условия:

1. Каждый элемент можно поместить в назначенную ему прямоугольную позицию на рабочем поле, то есть R_i имеет ширину w_i и высоту h_i не превышающие габариты позиции.

2. Никакие два элемента не должны перекрываться, т.е. $R_i \cap R_j = \emptyset, 1 \leq i, j \leq n$.

3. Соблюдено условие трассируемости, т.е. площадь свободных областей на рабочем поле $Q_j, 1 \leq j \leq k$ является достаточной для трассировки всех цепей.

4. Общая площадь прямоугольника, ограничивающего области R и Q , является минимальной.

5. Общая длина соединений (длина самой длинной цепи) минимизируется, т.е. $\sum_{i=1}^m L_i \rightarrow \min$.

Таким образом, задача размещения может быть интерпретирована как задача оптимизации аддитивной целевой функции, включающей нормированное значение штрафа за перекрытие площадей размещаемых элементов и оценку суммарной длины соединений:

$$F = \min_{z_j \in Z} (k \cdot O(L(z_j)) + P(S_{\text{общ}}(z_j))),$$

где z_j – текущий вариант размещения; k – весовой коэффициент; $S_{\text{общ}}$ – суммарная площадь перекрытия элементов; $O(L(z_j))$ – оценка длины соединений; $P(S_{\text{общ}}(z_j))$ – штраф за перекрытие площадей.

Эффективным инструментом решения NP-полных комбинаторных задач проектирования являются метаэвристические методы и модели. Термин «биоинспирированные алгоритмы» (или алгоритмы вдохновленные природой), включает в себя генетические, эволюционные, роевые алгоритмы воспроизводящие процессы, происходящие в живой и неживой природе [7–10]. Гибридные модели и системы базируются на научных методах, относящихся к вычислительному интеллекту: нечеткие модели и методы, эволюционные и генетические алгоритмы, многоагентные системы и методы. Они позволяют эффективно работать с нечеткой, плохо формализованной информацией и, в то же время, имеют объективный математический фундамент [11–14]. Хороший эффект при создании алгоритмов проектирования и оптимизации дает гибридизация популяционного алгоритма с двумя и более популяционными и/или не популяционными алгоритмами [10].

Одной из гибридных моделей является нечеткий генетический алгоритм. Нечеткое управление выполняется на основе использования следующего правила [14, 15]:

$$u(t) = f(e(t), e(t-1), \dots, e(t-r), u(t-1), \dots, u(t-r)),$$

где e – разница между ожидаемым и фактическим значениями выходного параметра объекта управления y^* ; f – нелинейная функция, отражающая отношение между входом и выходом НЛК; t – время.

3. МОДИФИЦИРОВАННЫЕ ГЕНЕТИЧЕСКИЕ ОПЕРАТОРЫ

Для решения поставленной задачи была предложена модифицированная методика кодирования хромосом (решений) популяции. Пусть длина хромосомы $L = [1; M]$, где N – количество генов в хромосоме. Поскольку разряды хромосомы $n \in [1; 6]$ определяют параметры функций принадлежности переменных \bar{x}_1 и \bar{x}_2 , а разряды $n \in [7; 12]$ – функций принадлежности переменных \bar{x}_3 и \bar{x}_4 , определим диапазон допустимых значений параметров x_n и σ_n . Для $n \in [1; 6]$: $0 \leq x_n \leq 1$; $0 \leq \sigma_n \leq 0,5$. Для $n \in [7; 12]$: $-1 \leq x_n \leq 1$; $0 \leq \sigma_n \leq 1$. Переменная y^{-k} может принимать значения на интервале: $0 \leq y^{-k} \leq 1$. Т.к. значение y^{-k} формируется из четырёх значений z_i^k , интервал значений z_n для всех n : $0 \leq z_n \leq 0,25$.

Перед применением генетических операторов производится селекция методом «фрулетки».

В алгоритме предложен модифицированный оператор кроссинговера (рис. 1). Перед применением оператора кроссинговера в выбранных индивидах случайным образом выбираем по одному участку, затем определяем 2 точки k_1 и k_2 , которые выбираются случайно на интервале: $k_1, k_2 \in [1; M]$, где N – количество генов в хромосоме. Затем, если $k_1 < k_2$ производится обмен генов, расположенных между точками k_1 и k_2 , если $k_1 > k_2$, производится обмен генов, расположенных между точками 1 и k_2 , k_1 и N .

При выполнении модифицированного многоточечного оператора мутации (рис. 2) из всех хромосом случайным выбираются 3 гена g_i из диапазона $n \in [1; M]$ каждой хромосомы. Затем для g_1 изменяется значение x_n , для g_2 – значения σ_n , для g_3 – значения z_n , новые значения выбираются случайно в пределах заданного интервала допустимых значений выбранного гена.

В качестве критерия останова используется истечение заданного количества итераций. После этого выполняется генетический алгоритм, происходит создание новой популяции решений и возможная корректировка управляющих параметров алгоритма с помощью нечеткого логического контроллера.

4. СТРУКТУРА АЛГОРИТМА

Гибридный алгоритм (рис. 3) начинает работу с задания исходных данных, определения критерия оптимизации и ограничений задачи. После этого происходит создание стартовой популяции решений (начальное размещение) и оценка качества начальной популяции (как отдельных индивидов, так и среднего значения по популяции) с точки зрения выбранного критерия оптимизации.

0.35	0.07	0.12	0.8	0.24	0.74	0.34	0.72	-0.34	-0.34	0.34	0.63
0.12	0.48	0.25	0.27	0.33	0.46	0.18	0.34	0.02	0.68	0.73	0.5
0.07	0.18	0.12	0.09	0.08	0.21	0.1	0.22	0.21	0.07	0.08	0.16
0.43	0.37	0.76	0.54	0.35	0.67	-0.56	0.2	0.63	-0.18	0.64	0.34
0.21	0.04	0.07	0.43	0.12	0.1	0.87	0.31	0.42	0.63	0.18	0.85
0.15	0.12	0.19	0.18	0.08	0.23	0.14	0.05	0.25	0.02	0.05	0.24
0.83	0.64	0.34	0.64	0.08	0.06	0.3	-0.48	0.86	0.04	-0.96	0.98
0.1	0.43	0.05	0.26	0.29	0.14	0.54	0.32	0.34	0.34	0.05	0.34
0.15	0.08	0.24	0.07	0.21	0.06	0.08	0.03	0.13	0.06	0.18	0.09

0.17	0.12	0.34	0.64	0.35	0.72	0.67	-0.34	0.34	0.67	-0.72	-0.67
0.08	0.18	0.34	0.26	0.12	0.08	0.34	0.75	0.23	0.82	0.64	0.34
0.15	0.18	0.01	0.07	0.08	0.24	0.22	0.12	0.09	0.02	0.24	0.2
0.35	0.76	0.07	0.54	0.98	0.74	0.2	-0.87	0.46	-0.18	0.46	0.43
0.34	0.07	0.18	0.33	0.34	0.14	0.31	0.04	0.73	0.63	0.82	0.21
0.23	0.19	0.18	0.18	0.09	0.03	0.05	0.12	0.03	0.02	0.06	0.15
0.64	0.84	0.04	0.34	0.43	0.74	0.93	0.73	-0.93	0.63	0.46	-0.67
0.43	0.35	0.34	0.32	0.4	0.46	0.34	0.64	0.34	0.5	0.86	0.91
0.08	0.08	0.06	0.21	0.36	0.21	0.08	0.09	0.2	0.16	0.13	0.03

↓

0.35	0.07	0.12	0.8	0.24	0.74	0.34	0.72	-0.34	-0.34	0.34	0.63
0.12	0.48	0.25	0.27	0.33	0.46	0.18	0.34	0.02	0.68	0.73	0.5
0.07	0.18	0.12	0.09	0.08	0.21	0.1	0.22	0.21	0.07	0.08	0.16
0.43	0.37	0.76	0.54	0.35	0.67	0.93	0.73	-0.93	0.63	0.64	0.34
0.21	0.04	0.07	0.43	0.12	0.1	0.34	0.64	0.34	0.5	0.18	0.85
0.15	0.12	0.19	0.18	0.08	0.23	0.08	0.09	0.2	0.16	0.05	0.24
0.83	0.64	0.34	0.64	0.08	0.06	0.3	-0.48	0.86	0.04	-0.96	0.98
0.1	0.43	0.05	0.26	0.29	0.14	0.54	0.32	0.34	0.34	0.05	0.34
0.15	0.08	0.24	0.07	0.21	0.06	0.08	0.03	0.13	0.06	0.18	0.09

0.17	0.12	0.34	0.64	0.35	0.72	0.67	-0.34	0.34	0.67	-0.72	-0.67
0.08	0.18	0.34	0.26	0.12	0.08	0.34	0.75	0.23	0.82	0.64	0.34
0.15	0.18	0.01	0.07	0.08	0.24	0.22	0.12	0.09	0.02	0.24	0.2
0.35	0.76	0.07	0.54	0.98	0.74	0.2	-0.87	0.46	-0.18	0.46	0.43
0.34	0.07	0.18	0.33	0.34	0.14	0.31	0.04	0.73	0.63	0.82	0.21
0.23	0.19	0.18	0.18	0.09	0.03	0.05	0.12	0.03	0.02	0.06	0.15
0.64	0.84	0.04	0.34	0.43	0.74	-0.56	0.2	0.63	-0.18	0.46	-0.67
0.43	0.35	0.34	0.32	0.4	0.46	0.87	0.31	0.42	0.63	0.86	0.91
0.08	0.08	0.06	0.21	0.36	0.21	0.14	0.05	0.25	0.02	0.13	0.03

Рис. 1. Пример работы модифицированного оператора кроссинговера

После этого выполняется генетический алгоритм, происходит создание новой популяции решений и возможная корректировка управляющих параметров алгоритма с помощью нечеткого логического контроллера.

Следующий блок представляет собой модифицированный муравьиный алгоритм. Его задача попытаться улучшить полученное размещение.

0.35	0.07	0.12	0.8	0.24	0.74	0.34	0.72	-0.34	-0.34	0.34	0.63
0.12	0.48	0.25	0.27	0.33	0.46	0.18	0.34	0.02	0.68	0.73	0.5
0.07	0.18	0.12	0.09	0.08	0.21	0.1	0.22	0.21	0.07	0.08	0.16
0.43	0.37	0.76	0.54	0.35	0.67	-0.56	0.2	0.63	-0.18	0.64	0.34
0.21	0.04	0.07	0.43	0.12	0.1	0.87	0.31	0.42	0.63	0.18	0.85
0.15	0.12	0.19	0.18	0.08	0.23	0.14	0.05	0.25	0.02	0.05	0.24
0.83	0.64	0.34	0.64	0.08	0.06	0.3	-0.48	0.86	0.04	-0.96	0.98
0.1	0.43	0.05	0.26	0.29	0.14	0.54	0.32	0.34	0.34	0.05	0.34
0.15	0.08	0.24	0.07	0.21	0.06	0.08	0.03	0.13	0.06	0.18	0.09



0.35	0.07	0.12	0.46	0.24	0.74	0.34	0.72	-0.34	-0.34	0.34	0.63
0.12	0.48	0.25	0.27	0.33	0.46	0.18	0.34	0.02	0.68	0.73	0.5
0.07	0.18	0.12	0.09	0.08	0.21	0.1	0.22	0.21	0.07	0.08	0.16
0.43	0.37	0.76	0.54	0.35	0.67	-0.56	0.2	0.63	-0.18	0.64	0.34
0.21	0.04	0.07	0.43	0.12	0.1	0.87	0.31	0.42	0.63	0.18	0.85
0.15	0.12	0.19	0.18	0.08	0.23	0.14	0.05	0.25	0.02	0.05	0.24
0.83	0.64	0.34	0.64	0.08	0.06	0.3	-0.48	0.86	0.04	-0.96	0.98
0.1	0.43	0.05	0.26	0.29	0.14	0.54	0.32	0.34	0.34	0.05	0.34
0.15	0.08	0.24	0.07	0.16	0.06	0.08	0.03	0.13	0.06	0.18	0.09

Рис. 2. Пример работы модифицированного оператора мутации

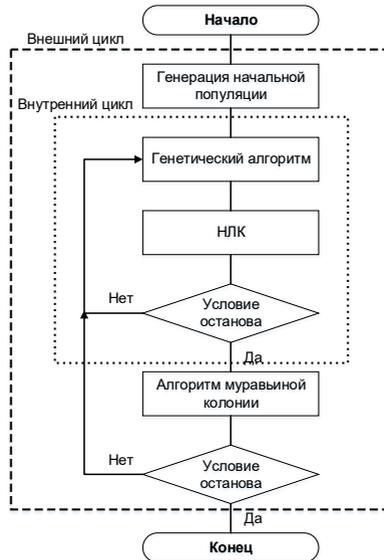


Рис. 3. Общая схема гибридного алгоритма

В конце происходит проверка выполнения критерия останова алгоритма, условия не выполнены, алгоритм переходит к следующей итерации, в противном случае выполнение алгоритма завершается.

5. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Временная сложность алгоритма – характеристика времени работы алгоритма, функционально зависящая от размерности задачи. Эффективность (скорость) работы параллельного генетического алгоритма совместного решения задач размещения и трассировки зависит от количества элементов схемы, количество цепей, числа итераций, количества хромосом популяции, а также количества эволюционных процессов (популяций).

Исследования проводились на двух конфигурации оборудования: Intel® Core(TM) i7-3630QM CPU @ 2.40 GHz, ОЗУ – 8ГБ (конфигурация 1) и Intel® Core(TM)2 Quad CPU Q8200 @ 2.40 GHz, ОЗУ – 4 ГБ (конфигурация 2). При выполнении алгоритма использовался нечёткий логический контроллер. Зависимость среднего времени выполнения алгоритма от числа размещаемых элементов показана на рис. 4.

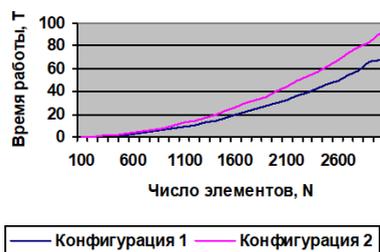


Рис.4. Зависимость времени работы от числа размещаемых элементов

Для полученной экспериментальным путем зависимости подбираем эмпирическую формулу, которая будет максимально соответствовать построенному графику. Анализ полученных экспериментальных данных показал, что характер полученной кривой наиболее точно описывается выражением:

$$y = ax^{\alpha}$$

Признаком такой функции является расположение опытных точек в области прямой линии в логарифмической системе координат:

$$\lg y = \lg a + \alpha \lg x.$$

Получим эмпирические формулы для обеих конфигураций. Для этого, подставив любые две точки, например (500; 1,88), (2500; 46,34) для первой конфигурации, и (500; 2,74), (2500; 62,96) – для второй. В результате решения системы 2-х уравнений получаем следующие значения переменных: $a_1 = 0,000011$, $\alpha_1 = 1,95$, $a_2 = 0,000015$, $\alpha_2 = 1,95$.

После подстановки вычисленных значений получаем выражение, отражающее зависимость времени выполнения алгоритма от количества размещаемых элементов для конфигураций 1 и 2:

$$f(x) = 0,000011x^{1,95};$$

$$f(x) = 0,000015x^{1,95}.$$

Исходя из того, что график зависимости, построенный аналитически, совпал с графиком зависимости на основе экспериментальных данных, можем сделать вывод, что зависимость носит квадратичный характер и может быть выражена формулой:

$$t(N) = aN^{1,95},$$

где t – время работы алгоритма, N – количество элементов, a – коэффициент, зависящий от технических характеристик ЭВМ.

В ходе вычислительных экспериментов проводились исследования влияния использования нечеткого логического контроллера на качество получаемых результатов. Значения параметров определялись по результатам работы блока обучения нечеткого логического контроллера. Алгоритм выполнялся на одном и том же наборе исходных данных при различном количестве итераций. Результаты работы алгоритма (с использованием нечеткого логического контроллера и без него) отражены на рис. 5.

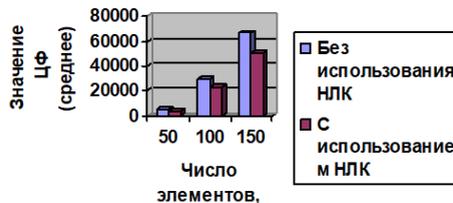


Рис. 5. Сравнение качества решений с использованием НЛК и без него

Анализ полученные в ходе вычислительных экспериментов данных, позволяет сделать вывод, что использование НЛК улучшает качество решения задачи в среднем на 25%. Эффективность использования контроллера повышается после введения блока обучения на основе модели искусственной нейронной сети.

Программная реализация предложенного алгоритма позволяет визуализировать процесс работы нечеткого логического контроллера и вывести на экран графики изменения среднего и минимального значения ЦФ, а также графики изменения значений вероятности выполнения основных логических операторов: кроссингвера, мутации и миграции в процессе эволюции (рис. 6, 7).

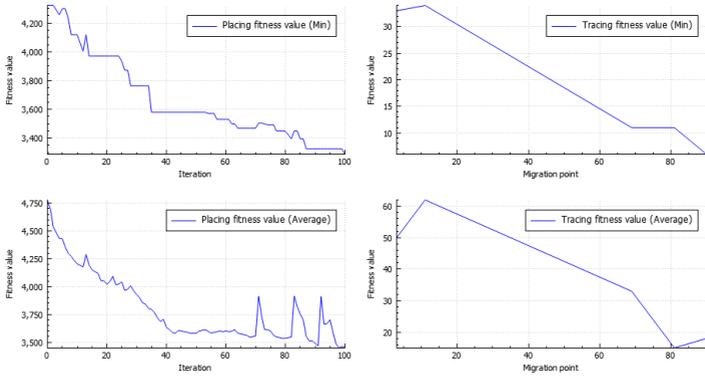


Рис. 6. Графики изменения среднего и минимального значений ЦФ

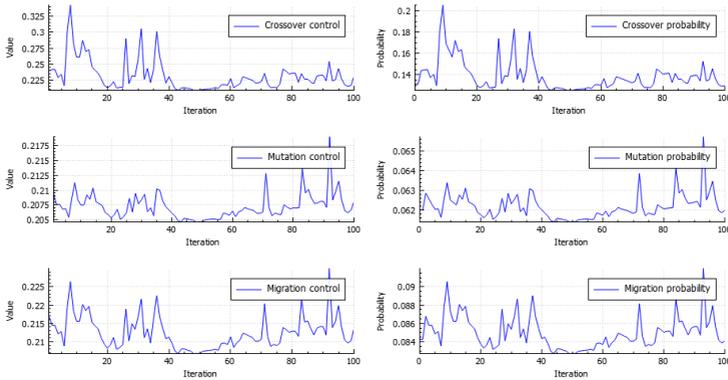


Рис. 7. Графики изменения выходных параметров нечёткого логического контроллера

Анализ полученных данных показывает, что на начальном этапе, когда в популяции наблюдается большое разнообразие особей, вероятность мутации ниже, чем на последующих итерациях, а вероятность кроссинговера выше. Затем, по мере уменьшения разнообразия популяции, растёт вероятность мутации и уменьшается вероятность кроссинговера. График изменения вероятности миграции ведёт себя аналогично графику изменения вероятности мутации.

6. ЗАКЛЮЧЕНИЕ

Рассмотрена задача размещения разногабаритных элементов, разработаны модифицированные генетические операторы, а также структура гибридного алгоритма. Выполнена программная реализация предложенного гибридного алгоритма, проведены серии вычислительных экспериментов для определения эффективности разработанного алгоритма.

Литература

1. Ярушкина Н.Г. Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004.
2. Гладков Л.А., Гладкова Н.В. Решение задач поиска и оптимизации решений на основе нечетких генетических алгоритмов и многоагентных подходов // Известия ТРТУ. Тематический выпуск «Интеллектуальные САПР». – 2006. – № 8 (63). – С. 83-88.
3. Гладков Л.А., Ясир М.Д., Гладкова Н.В. Гибридный метод решения задачи размещения элементов цифровых вычислительных устройств // Известия ЮФУ. Технические науки. – 2021. – № 4. – С. 231-243.
4. Sherwani Naveed. Algorithms for VLSI Physical Design Automation, Kluwer Academic Publishers, Boston/Dordrecht/London, 1995.
5. Cohoon J.P., Karro J., Lienig J. Evolutionary Algorithms for the Physical Design of VLSI Circuits. Advances in Evolutionary Computing: Theory and Applications, Ghosh, A., Tsutsui, S. (eds.) Springer Verlag, London, 2003.
6. Charles J. Alpert, Dinesh P. Mehta, Sachin S. Sapatnekar. Handbook of algorithms for physical design automation. CRC Press, New York, USA, 2009.
7. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы. – М.: Физматлит, 2010.
8. Емельянов В.В., Курейчик В.В., Курейчик В.М. Теория и практика эволюционного моделирования. – М.: Физматлит, 2003.
9. Гладков Л.А., Курейчик В.М., Курейчик В.В., Сороколетов П.В. Биоспирированные методы в оптимизации. – М.: Физматлит, 2009.
10. Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой. – М.: Изд-во МГТУ им. Баумана, 2016.
11. Борисов В.В., Круглов В.В., Федулов А.С. Нечеткие модели и сети. – М.: Горячая линия – Телеком, 2007. – 284 с.
12. Пегат А. Нечеткое моделирование и управление. М.: БИНОМ. Лаборатория знаний, 2009.
13. Herrera F., Lozano M. Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions. // Soft Computing. – Springer-Verlag, 2003. – 7 (2003). – P. 545-562.
14. Michael A., Takagi H. Dynamic control of genetic algorithms using fuzzy logic techniques // Proc. of the 5th International Conference on Genetic Algorithms. – Morgan Kaufmann, 1993. – P. 76-83
15. Гладков Л.А. Интегрированный алгоритм решения задач размещения и трассировки на основе нечетких генетических методов // Известия ЮФУ. Технические науки. – 2011. – № 7 (120). – С. 22-30.

БИОИНСПИРИРОВАННЫЙ АЛГОРИТМ ПЛАНИРОВАНИЯ ТРАЕКТОРИИ В УСЛОВИЯХ ЧАСТИЧНОЙ НЕОПРЕДЕЛЕННОСТИ И НАЛИЧИИ ПЕРЕМЕЩАЮЩИХСЯ ПРЕПЯТСТВИЙ*

*Веселов Г.Е., д.т.н., доцент, директор
ИКТИБ Южный федеральный университет
e-mail: gev@sfedu.ru*

*Лебедев Б.К., д.т.н., профессор
Южный федеральный университет
lebedev.b.k@gmail.com*

*Лебедев О.Б., к.т.н., доцент
Южный федеральный университет
lebedev.ob@mail.ru*

1. ВВЕДЕНИЕ

Существует множество классификаций, постановок и методов решения задач проектирования траекторий [1, 2]. Но в любом случае можно выделить два класса задач. В задачах первого класса предусматривается использование проекта (траектории) после того как она будет полностью спроектирована. В задачах второго класса процессы синтеза (прокладки) траектории и передвижения по ней подвижного объекта (ПО) связаны, причем перемещение по траектории происходит практически одновременно с ее формированием (прокладкой). Это связано с неопределенностью ограничений на карте местности, препятствующих прокладке траекторий. Наличие неопределенности обусловлено тем, что применительно к ПО, такие препятствия не локализованы, т.е. система управления предварительно не имеет никакой информации ни об их форме, ни о размере, ни, о параметрах их перемещения и расположении. Выявление ограничений, препятствующих прокладке траектории из текущей позиции, осуществляется в процессе построения траектории. Траектория строится динамически с учетом динамически изменяемых ограничений. В связи с этим процесс прокладки траектории осуществляется пошагово следующим образом. Последовательно на каждом шаге относительно текущей позиции ПО формируется локальная зона видимости (ЛЗВ). В пределах ЛЗВ с помощью радара ПО осу-

*Работа выполнена при финансовой поддержке РФФИ, проект №20-07-00260.

ществляется определение координат (локализация) всех препятствий и параметры их перемещения [3, 4]. Далее экстраполируются параметры перемещения подвижных объектов. После этого ПО в течение заданного интервала времени перемещается в пределах ЛЗВ с учетом прогнозируемых параметров размещения подвижных и не подвижных объектов по кратчайшему пути от текущей позиции к новой позиции, которая после ее достижения объявляется текущей.

Наибольшее распространение для построения маршрута на плоскости получили волновые алгоритмы [5]. Карта местности при волновой трассировке разбивается на квадраты (ячейки). Классический алгоритм волновой трассировки включает два этапа. На первом этапе в процессе распространения волны от источника к цели ячейкам дискретного рабочего поля (ДРП) присваиваются весовые оценки, связанные с принятым критерием оптимальности. На втором этапе алгоритма осуществляется построение пути. Для этого следует, начиная от ячейки – цели, двигаться в направлении, противоположном направлению распространения волны, переходя от ячейки с большим весом к смежной ячейке с меньшим весом до тех пор, пока не будет достигнута ячейка – источник. Ячейки ДРП, выбранные в ходе указанного процесса, и определяют искомое оптимальное соединение минимальной длины.

В последнее время для решения различных «сложных» задач всё чаще используются способы, основанные на применении методов случайного направленного поиска. В основе большинства этих алгоритмов лежат метаэвристики, заимствованные в природе [5, 6]. К таким методам относятся метод моделирования отжига, методы генетического поиска (эволюционная адаптация), методы роевого интеллекта, методы альтернативной поисковой адаптации на основе вероятностных обучающихся автоматов. Являясь по своей сути итерационными, алгоритмы на основе моделирования естественных процессов отличаются от обычных итерационных процедур «слепого поиска». В ряде работ [7–10] описываются подходы на основе различных эвристических методов к решению данной задачи, которые пока не показали слишком хороших результатов. Анализ методов решения сложных прикладных задач показывает, что применение любого одного алгоритма оптимизации (как классического, так и популяционного) далеко не всегда приводит к успеху. В гибридных (комбинированных) алгоритмах, объединяющих различные либо одинаковые алгоритмы, но с различными значениями свободных параметров, преимуществами одного алгоритма могут компенсировать недостатки другого. Поэтому одним из основных путей повышения эффективности решения оптимизационных задач в настоящее время является разработка гибридных популяционных алгоритмов [11].

В работе описывается гибридный алгоритм планирования траектории в условиях частичной неопределенности для двухмерного пространства, основанный на интеграции волнового и муравьиного алгоритмов, позволяющий строить в реальном масштабе времени траектории минимальной длины с одновременной оптимизацией ряда других критериев качества построенного пути.

2. ПОСТАНОВКА ЗАДАЧИ И СХЕМА БАЗОВОГО АЛГОРИТМА

Карта местности первоначально разбивается на множество ячеек с заданным шагом дискретизации и представляется в виде дискретного рабочего поля (ДРП). Модель такого рода называют **рецепторной**. Рецепторную модель карты можно представить в виде графа. Вершины графа соответствуют ячейкам. Если две ячейки смежны, то соответствующие этим ячейкам вершины графа связаны ребром. Вершины графа представляют все возможные размещения ПО в пространстве поиска.

Первой задачей является формирование локальной зоны видимости (ЛЗВ) на которой отражается пространство, доступное для перемещения ПО. Данные о препятствиях поступают от сенсорной системы ПО. Координаты препятствий совмещают с вершинами графа. Вершины помечаются как препятствия, и происходит удаление ребер вокруг каждой вершины. В пространстве сформированного графа происходит поиск оптимального пути.

Основными показателями при планировании траектории являются: *Длина (P_L)*, *Показатель безопасности (S_m)*, *Время выполнения задания (t_m)*, *Коэффициент успешности миссии (M)* [1,4].

P_L – длина всей траектории, пройденной ПО от начальной точки до цели.

S_m – минимальное расстояние между любым датчиком ПО и любым препятствием вдоль всей траектории. Этот показатель определяет максимальный риск в течение всего движения.

t_m – время, необходимое для завершения движения.

V – количество успешных миссий в недетерминированных средах со сложными препятствиями.

В общем случае планирование траектории производится в соответствии с алгоритмом, представленном ниже.

Алгоритм планирования траектории

1. $t=0$.
 2. Подвижный объект (ПО) помещается в текущую позицию (ТП) $p(t)$ локальной зоны видимости.
 3. На основе данных сенсорной системы ПО генерируется локальная зона видимости $z(t)$ относительно точки $p(t)$.
 4. Все сгенерированные ЛЗВ объединяются в одну объединенную ЛЗВ (ОЛЗВ) $Oz(t)$. Препятствия за пределами ОЛЗВ $Oz(t)$ не рассматриваются.
 5. По информации сенсорной системы ПО формируются координаты расположения в пределах ОЛЗВ множества препятствий в момент времени $t=0$: $K=\{k_{oi} \mid i=1,2,\dots,n\}$, $k_{oi}=(x_{oi}, y_{oi})$.
 6. Множество препятствий K разбивается на подмножества $K=K_1 \cup K_2$:
 K_1 – множество неподвижных препятствий;
 K_2 – множество подвижных препятствий в момент времени $t=0$;
 $K_0=\{k_{oi} \mid i=1,2,\dots,n\}$, $k_{oi}=(x_{oi}, y_{oi})$.
 7. Для каждого подвижного препятствия фиксируются параметры передвижения: скорость и направление передвижения каждого препятствия.
 8. Методами экстраполяции рассчитываются прогнозируемые координаты $K_p=\{k_{pi} \mid i=1,2,\dots,n\}$, $k_{pi}=(x_{pi}, y_{pi})$ подвижных препятствий через промежуток времени δt .
 9. На карте ЛЗВ отражается размещение всех препятствий (подвижных и неподвижных).
 10. Множество позиций, в которых размещены объединенное множество препятствий в моменты времени t и $(t+\delta t)$, рассматривается, как множество неподвижных препятствий.
 11. На карте ЛЗВ строится траектория между точками $p(t)$ и *цель*.
 12. Если точка *цель* входит в ОЛЗВ, то траектория сформирована полностью и переход к 15, иначе переход к п.8.
 13. На участке, построенной траектории, входящем в ЛЗВ $z(t)$, выбирается текущая позиция $p(t+1)$, ближайшая к позиции *цель* в которую перемещается ПО.
 14. $t=t+1$. Переход к п.3.
 15. Конец работы алгоритма.
- Интегральная оценка траектории рассчитывается по следующей формуле:

$$W=k_1P_L + k_2S_m + k_3t_m + k_4/V.$$

3. ПОСТРОЕНИЕ НА ОСНОВЕ ГИБРИДИЗАЦИИ ВОЛНОВОГО И МУРАВЬИНОГО АЛГОРИТМОВ ТРАЕКТОРИИ В УСЛОВИЯХ ЧАСТИЧНОЙ НЕОПРЕДЕЛЕННОСТИ

В общем случае интеграция основана на сращивании механизмов волнового и муравьиного алгоритмов и реализуется при выполнении двух основных этапов. На первом этапе путем распространения на ДРП волны от источника к цели формируется пространство поиска – R , представляющая совокупность связанных ячеек ДРП, достигнутых волной, с присвоенными им весами. На втором этапе муравьиным алгоритмом на базе сформированного пространства поиска R прокладывается маршрут.

Формируется рецепторная модель карты местности, включающая множество ячеек $E = \{e_i | i = 1, 2, \dots, n_e\}$. Каждая ячейка имеет координаты (x_i, y_i) . Задаются ячейка e_u , которая помечается как исходная позиция ИП, в которую помещается ПО и ячейка e_v , которая помечается как целевая позиция (цель). Изначально все ячейки считаются свободными.

Процесс планирования траектории итерационный и включает шесть этапов, выполняемых на каждой итерации l . На каждой итерации строится отдельный участок траектории, являющийся продолжением ранее построенного участка. А вся траектория является совокупностью отдельных участков, связывающая исходную позицию ПО с целевой позицией.

На первом этапе по данным радара, размещенного на данной итерации l в точке (ячейке) $ТП(l)$ (текущая позиция), формируется локальная зона видимости: определяются ее границы на модели карты местности и выявляются, содержащиеся в ней препятствия. По информации сенсорной системы ПО формируются координаты расположения множества препятствий в момент времени $t=0$: $K = \{k_{oi} | i = 1, 2, \dots, n\}$, $k_{oi} = (x_{oi}, y_{oi})$.

На втором этапе множество препятствий K разбивается на множество K_1 неподвижных объектов и множество подвижных объектов K_2 .

Для каждого подвижного препятствия радаром фиксируются параметры передвижения: скорость и направление передвижения каждого препятствия. Методами экстраполяции рассчитываются прогнозируемые координаты $K_p = \{k_{pi} | i = 1, 2, \dots, n\}$, $k_{pi} = (x_{pi}, y_{pi})$ подвижных препятствий через промежуток времени δt .

На карте ЛЗВ отражается размещение всех препятствий (подвижных и неподвижных).

Ячейки с препятствиями помечаются. Координаты точки $ТП(t)$ определяются на предыдущей итерации $(l-1)$. Данные радара, полученные на итерации l , объединяются с данными, полученными на основе экстраполяции.

На третьем этапе путем распространения на ДРП волны от источника (точка $ТП(t)$) к точке *цель* формируется пространство поиска траектории – $ППТ(l)$, представляющая совокупность связанных ячеек ДРП, достигнутых волной, с присвоенными им весами.

На четвертом этапе муравьиным алгоритмом на сформированной области $ППТ(l)$ прокладывается траектория $M(l)$ из точки *цель* к точке $ТП(l)$. Отметим, что траектория $M(l)$ состоит из двух частей. Вторая часть траектории $M2(l)$ проходит через зону свободную от препятствий, первая часть $M1(l)$ проходит через зону видимости с содержащимися в ней препятствиями.

На пятом этапе на первой части траектории часть $M1(l)$ выбирается входящая в ее состав ячейка e_k с максимальным весом, являющаяся концевой в составе $M1(l)$. Следующая в траектории $M(l)$ за ячейкой e_k ячейка e_n лежит уже во второй части траектории $M2(l)$.

На шестом этапе в $M1(l)$ выбирается ячейка e_p с меньшим в сравнении с ячейкой e_k весом на величину ε . Параметр ε является управляющим. ПО перемещается по траектории $M1(l)$ из точки $ТП(t)$ в ячейку e_p . Участок на первой части траектории часть $M1(l)$ между точками $ТП(t)$ и e_p включается в планируемую траекторию. Ячейка e_p далее рассматривается как новая текущая позиция ПО на следующей итерации ($l+1$). Участок $M2(l)$ удаляется.

4. МУРАВЬИНЫЙ АЛГОРИТМ ПРОКЛАДКИ МАРШРУТА

После распространения волны на ДРП от источника (точка $ТП(l)$) к точке *цель* и присвоение веса ячейкам ДРП муравьиным алгоритмом на ДРП из точки *цель* в точку $ТП(l)$ строится маршрут $M(l)$ [12].

Поиск решения осуществляется популяцией агентов на графе поиска решений $G=(E,U)$. Вершины множества E соответствуют ячейкам области R . Две вершины $e_i \in E$ и $e_j \in E$ связаны ребром u_k , если соответствующие им ячейки на модели местности в виде ДРП смежны и возможен переход соединения из одной ячейки в другую.

Каждая итерация l муравьиного алгоритма включает три этапа.

Предварительно на ребрах графа поиска решений $G=(E,U)$ откладывается начальное количество феромона – $\xi=Q/n_u$, где $n_u=|U|$, Q – управляющий параметр.

На первом этапе каждый агент a_k популяции находит решение. Агент a_k конструктивным алгоритмом строит на графе G маршрут $M_k(l)$ из вершины e_u в вершину e_v .

Рассчитывается оценка $F_k(l)$ k -го решения, полученного агентом a_k на l -ой итерации.

На втором этапе каждый агент a_k откладывает феромон на ребрах маршрута, построенного на графовой модели пространства поиска, в количестве пропорциональном оценке $F_k(l)$ k -го решения $\tau_k(l) = \delta / F_k(l)$. В работе используется циклический (ant-cycle) метод муравьиных систем [12]. В этом случае феромон откладывается совместно агентами на ребрах графа после полного формирования решений всей популяцией агентов.

На третьем этапе осуществляется общее испарение (уменьшение) количества феромона на всех ребрах графа поиска решений $G=(E,U)$ на некоторую величину, в соответствии с формулой $h_j = h_j(1 - \sigma)$, где σ – коэффициент обновления, h_j – суммарное количество феромона, отложенного муравьями на ребре $u_j \in U$.

Построение муравьем траектории осуществляется на базе сформированного пространства поиска $ППТ(l)$ после распространения на ДРП волны от источника к точке *цель*, с присвоением весов свободным ячейкам. Подобно волновому алгоритму муравей строит на $ППТ(l)$ маршрут, представляющий совокупность ячеек с последовательно уменьшающимся весом. Всеми муравьями будут построены маршруты одной минимальной длины, но отличающимися конфигурациями. В этом случае оценкой маршрута $M_k(l)$, построенного агентом a_k служит оценка его конфигурации.

$$F_k = \alpha f_{rk} + \beta f_{gk},$$

где α, β – управляющие параметры, которые подбираются экспериментально;

f_{rk} – число поворотов маршрута M_k ;

f_{gk} – число содержащих препятствия ячеек, смежных ячейкам маршрута M_k .

Чем меньше f_{rk} , тем меньше показатель **Время выполнения задания** – время, необходимое для завершения движения.

Чем меньше f_{gk} , тем выше **Показатель безопасности** – число позиций ПО на минимальном расстоянии между ПО и любым препятствием вдоль всей траектории.

Цель оптимизации – минимизация значения F_k .

Отметим, что показатели P_L, S_m, t_m, V не являются объектами оптимизационных процедур. Объектом оптимизации является показатель $F_k = \alpha f_{rk} + \beta f_{gk}$, минимизация которого на базе эвристик гибридного алгоритма планирования косвенно способствует минимизации интегральной оценки траектории:

$$W = k_1 P_L + k_2 S_m + k_3 t_m + k_4 V.$$

5. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Для проведения экспериментов была использована процедура синтеза контрольных примеров с известным оптимумом по аналогии с известным методом РЕКУ (Placement Examples with Known Upper bounds on wirelength).

Для задачи планирования траектории созданы четыре набора примеров с известным оптимумом по длине траектории и оптимальным значением F_{opt} оценки конфигурации траектории F_k . Оптимизация проводилась по критерию минимальной длины траектории и оптимального значения оценки конфигурации и опиралась на методику, изложенную в работе [1, 4].

На основе обработки экспериментальных исследований была построена средняя зависимость качества решений от числа итераций (рис. 1). Оценкой качества служит величина F_k/F_{opt} , где F – оценка полученного решения. Исследования показали, что число итераций, при которых алгоритм находил лучшее решение лежит в пределах 68-80. Из графика видно, что в среднем на 76-ой итерации решение близко к оптимальному.

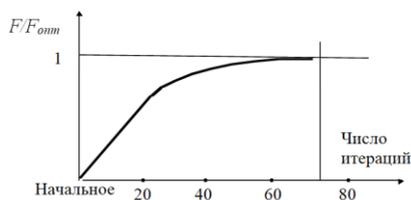


Рис. 1. Средняя зависимость качества решений от числа итераций

Общая оценка временной сложности при любом подходе к гибридизации лежит в пределах $O(n^2)$ – $O(n^3)$.

Сравнение значений критерия, полученных гибридным алгоритмом на тестовых примерах с известным оптимумом показало, что у 60% примеров полученное решение было оптимальным, у 15% примеров решения были на 5% хуже, а у 25% примеров решения были хуже не более, чем на 2%.

Сравнительный анализ производился на тестовом наборе сцен, приведенном в работе [1]. Сравнение разработанного гибридного алгоритма с известными алгоритмами [9–15] по интегральной оценке F_T траектории W показало, что при меньшем времени работы у полученных с помощью разработанного алгоритма решений значения F_T меньше в среднем на 6%.

На рис. 2 а,б,в – приведены примеры проложенных соединений.

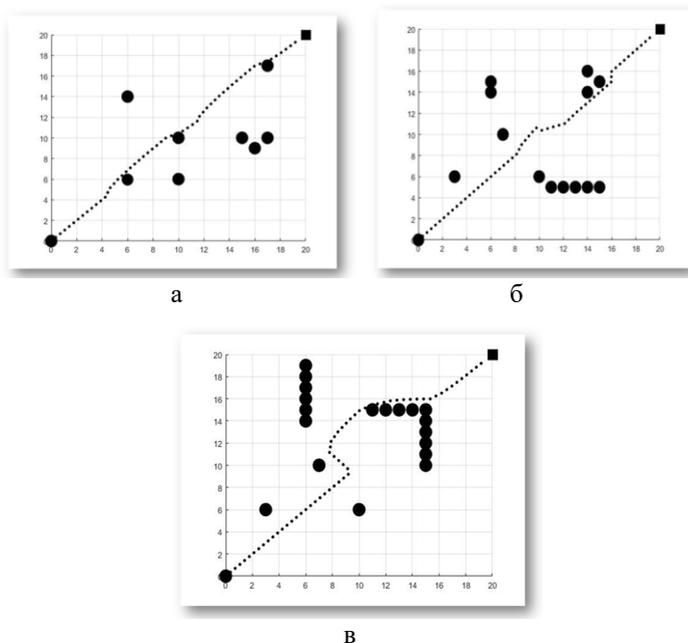


Рис. 2. Примеры сцен с проложенными траекториями

6. ЗАКЛЮЧЕНИЕ

Несмотря на достаточно большое число разработанных моделей, и алгоритмов управления подвижными объектами в условиях неопределенности исследователи часто сталкиваются с рядом проблем, к числу которых относятся трудность в обосновании качества результатов анализа, учитывающего специфику конкретной задачи.

К числу перспективных тенденций относятся разработка гибридных методов. В гибридных (комбинированных) алгоритмах, объединяющих различные, либо одинаковые алгоритмы, но с различными значениями свободных параметров, преимущества одного алгоритма могут компенсировать недостатки другого. В работе предлагается гибридный алгоритм, основанный на интеграции волнового и муравьиного алгоритмов, позволяющий строить в реальном масштабе времени траектории минимальной длины с одновременной оптимизацией ряда других критериев качества построенного пути. Отличительная особенность заключается в том, что перемещение подвижного объекта по траектории происходит производится практически одновременно с ее

формированием (прокладкой), в режиме on-line. Ключевая проблема, которая была решена в данной работе – это методика срачивания механизмов муравьиного и волнового алгоритмов. Источником усовершенствования может быть более детальная проработка вопросов, связанных с формированием зоны видимости и выбором (сменой) текущих исходных позиций, что ускорит процесс достижения подвижным объектом целевого состояния.

Литература

1. Пшихопов В.Х. и др. Интеллектуальное планирование траекторий подвижных объектов в средах с препятствиями / под ред. В.Х. Пшихопова. – М.: Физматлит. – 2014.
2. Пшихопов В.Х., Медведев М.Ю. Управление подвижными объектами в определенных и неопределенных средах. – М.: Наука. – 2011.
3. Гузик В.Ф., Переверзев В.А., Пьявченко А.О., Сапрыкин Р.В. Принципы построения экстраполирующего многомерного нейросетевого планировщика интеллектуальной системы позиционно-тракторного управления подвижными объектами // Известия ЮФУ. Технические науки. – 2016. – № 2 (175). – С. 67-80.
4. Пшихопов В.Х., Медведев М.Ю., Гуренко Б.В. Алгоритмы адаптивных позиционно-тракторных систем управления подвижными объектами // Проблемы управления. – 2015. – № 4. – С. 66-74.
5. Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учебное пособие. – 2-е изд. – М: Изд-во МГТУ им. Н.Э. Баумана, 2016. – 448 с.
6. Caro G.D., Ducatelle F., Gambardella L.M. AntHocNet: An adaptive nature inspired algorithm for routing in mobile ad hoc networks // European Transactions on Telecommunications. – 2005. – No. 16 (5). – P. 443-455.
7. Hoefler T., Snir M. Generic Topology Mapping Strategies for Large-scale Parallel Architectures. – University of Illinois at Urbana-Champaign Urbana, 2011. – P. 75-85.
8. Нейдорф Р.А., Полях В.В., Черногоров И.В., Ярахмедов О.Т. Исследование эвристических алгоритмов в задачах прокладки и оптимизация маршрутов в среде с препятствиями // Известия ЮФУ. Технические науки. – 2016. – № 3 (176). – С. 127-143.
9. Fatemeh K.P., Fardad F., Reza S.N. Comparing the Performance of Genetic Algorithm and Ant Colony Optimization Algorithm for Mobile Robot Path Planning in the Dynamic Environments with Different Complexities // Journal of Academic and Applied Studies. – 2013. – Vol. 3 (2). – P. 29-44.
10. Chen S., Eshaghian M.M. A fast recursive mapping algorithm // Department of computer and information science. – New Jersey, USA: New Jersey, 2013. – P. 219-227.

11. Raidl G.R. A Unified View on Hybrid Metaheuristics // Lecture Notes in Computer Science. – Springer-Verlag, 2006. – P. 1-12.
12. Лебедев О.Б. Модели адаптивного поведения муравьиной колонии в задачах проектирования. – Таганрог: Изд-во ЮФУ., 2013. – 199 с.
13. Veselov G.E., Lebedev B.K., Lebedev O.B. Management of Behavior of a Swarm of Robots Applicable to the Tasks of Monitoring a Some Territory // Artificial Intelligence Methods in Intelligent Algorithms. – Springer, Czech Republic, 2020. – Vol 2. – P. 324-332.
14. Веселов Г.Е., Лебедев Б.К., Лебедев О.Б. Управление движением группы мобильных роботов в колонне // Научно-практический журнал Информатизация и связь. – 2021. – № 3. – С. 7-11.
15. Веселов Г.Е., Лебедев Б.К., Лебедев О.Б. Адаптивное управление передвижением коллектива мобильных роботов развернутых в линию // Интегрированные модели и мягкие вычисления в искусственном интеллекте: Сборник научных трудов X-й Международной научно-технической конференции (ИММВ-2021). – Смоленск: Универсум, 2021. – С. 238-246.

БИОИНСПИРИРОВАННЫЙ АЛГОРИТМ ПОСТРОЕНИЯ ПАРАСОЧЕТАНИЯ*

*Лебедев О.Б., к.т.н., доцент
Южный федеральный университет
e-mail: lebedev.ob@mail.ru
Жиглатый А.А., аспирант
Южный федеральный университет
artemiy.zhiglaty@gmail.com*

1. ВВЕДЕНИЕ

Задача определения паросочетаний – одна из фундаментальных задач в области математической оптимизации. Подмножество ребер $M \subseteq U$ двудольного графа $G=(V,U)$ называется паросочетанием в графе G , если каждой вершине $v \in V$ инцидентно не более одного ребра из M . Максимальное паросочетание – это такое паросочетание M в графе G , которое не содержится ни в каком другом паросочетании этого графа, то есть к нему невозможно добавить ни одно ребро, которое бы являлось несмежным ко всем рёбрам паросочетания [1]. Наибольшее паросочетание (или максимальное по размеру паросочетание [2]) – это такое паросочетание, которое содержит максимальное количество рёбер. Совершенным паросочетанием называется паросочетание, в котором участвуют все вершины графа.

Алгоритмы построения паросочетания применяются в задачах о назначении, планирования и размещения при проектировании топологии СБИС, покрытия, транспортных задачах, при проектировании инженерных сетей, коммуникаций, построения систем поддержки принятия решений в неопределенных условиях, и т.п. Задачи такого типа относятся к переборным задачам с экспоненциальной временной сложностью. В этой связи разрабатывают различные эвристики для построения алгоритмов с полиномиальной временной сложностью. Существуют алгоритмы определения паросочетаний в графе, основанные на использовании потоков в сетях [1, 2], имитационного моделирования [3], генетического поиска [4] и других эвристик, которые обеспечивают приемлемые результаты при решении задач малой и средней сложности. Часто эта процедура используется в итерационных

*Работа выполнена при финансовой поддержке РФФИ, проект №20-07-00260.

структурах. Это предъявляет повышенные требования к качеству и времени решения задачи нахождения максимального паросочетания. Возникшие потребности в решении задач большой и очень большой размерности является побудительным мотивом исследований и разработок новых эффективных алгоритмов. Анализ литературы показывает, что наиболее успешными в этих условиях являются математические методы, в которых заложены принципы природных механизмов принятия решений [4–6]. К таким методам можно отнести, прежде всего, методы моделирования отжига [7], метод эволюционного моделирования и генетические алгоритмы [8], эволюционной адаптации [9], алгоритмы роевого интеллекта [10] и муравьиные алгоритмы [11, 12]. Идея муравьиного алгоритма – моделирование поведения муравьёв, связанного с их способностью быстро находить кратчайший путь от муравейника к источнику пищи. Основу поведения муравьиной колонии составляет самоорганизация, обеспечивающая достижения общих целей колонии на основе низкоуровневого взаимодействия благодаря которому, в целом, колония представляет собой разумную многоагентную систему.

Для повышения эффективности, усиления сходимости алгоритма и способности выхода из локальных оптимумов предложен подход к построению алгоритма паросочетания на основе модели адаптивного поведения муравьиной колонии. Предложенный алгоритм отличается от канонического тем, что в качестве модели пространства поиска решений используется звездный граф.

Предложенный подход обеспечивает более широкий обзор пространства решений и более высокую вероятность локализации глобального экстремума задачи

В работе излагается методика решения задачи нахождения максимального паросочетания в графе, и родственных ей задач раскраски графа и выделения клик в графе, основанная на моделировании адаптивного поведения муравьиной колонии.

2. ПОСТАНОВКА ЗАДАЧИ НАХОЖДЕНИЯ ПАРСОЧЕТАНИЯ В ГРАФЕ

Пусть дан двудольный граф $G=(V,U)$, представленный на рисунке 1. $V=\{v_i|i=1,2,\dots,n_i\}$. $U=\{u_i|i=1,2,\dots,8\}$. $U=\{u_j|j=1,2,\dots,n_j\}$. $n_i=10$, $n_j=11$.

Паросочетание P такого графа определяется как множество попарно несмежных рёбер, то есть рёбер, не имеющих общих вершин [1]. Например: для графа G (рис.1) паросочетание $P=\{u_3, u_5, u_8, u_{11}\}$. Будем говорить, что вершина графа G принадлежит паросочетанию, понимая под этим, что она инцидентна некоторому ребру паросочетания.

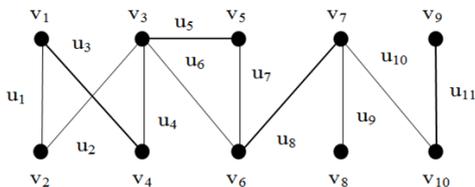


Рис. 1. Пример графа G

Построим граф $G_r=(U^r, V)$ – реберный для графа G . Вершины графа G_r – соответствуют рёбрам графа G . Пара вершин (u_i, u_j) в графе G_r связаны ребром в том и только в том случае, если в графе G пара рёбер (u_i, u_j) смежны, т.е. инциденты одной вершине [1].

Для примера, представленного на рис. 1, реберный граф G_r графа G имеет вид, представленный на рис. 2.

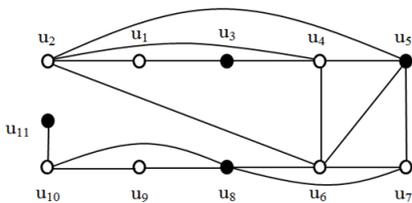


Рис. 2. Реберный граф G_r

Множество $U^r_0 \subset U^r$ вершин графа $G_r=(U^r, V)$ называется внутренне-устойчивым, если любые две вершины $u^r_i \in U^r_0$ и $u^r_j \in U^r_0$ не являются смежными [2]. Максимальное число вершин во внутренне-устойчивом множестве графа U^r называется числом внутренней устойчивости и обозначается как $\alpha(G_r)$. Иногда число внутренней устойчивости называют также числом независимости графа G_r [3].

Подмножество вершин $U^r_0=\{u_3, u_5, u_8, u_{11}\}$ (рис.2) графа $G_r=(U^r, V)$ является внутренне устойчивым, т.к. любые две вершины подмножества U^r_0 не смежны. Таким образом, паросочетанию в графе G соответствует внутренне-устойчивое подмножество реберного графа G_r графа G . Максимальному по мощности паросочетанию в графе G соответствует предельное внутренне-устойчивое подмножество (содержащее наибольшее число вершин) реберного графа G_r .

Рассмотрим задачу о клике. Кликкой графа G называется максимальное по включению множество $V_0 \subset V$ вершин графа G , любые две из которых являются смежными. Другими словами, клика графа есть подмножество его вершин, такое, что между каждой парой вершин этого подмножества существует ребро.

Максимальная клика – это клика, которая не может быть расширена путём включения дополнительных смежных вершин, т.е. нет клики большего размера, включающей все вершины данной клики. Наибольшая клика – это клика максимального размера для данного графа [4].

Дополнение графа (обратный граф) – граф G_o , имеющий то же множество вершин, что и заданный граф G , но в котором две несовпадающие вершины смежны тогда и только тогда, когда они не смежны в G [2].

Нетрудно видеть, что при переходе от графа G к его дополнению G_o каждая клика в G переходит в независимое множество в G_o . Отсюда следует, что задача выделения клики в графе G сводится к задаче выделения независимого множества вершин в графе G_o .

Таким образом, в основе процедур построения максимального паросочетания, звездный граф выделения в графе клик лежит одна общая процедура формирования в графе $G(V,U)$ внутренне-устойчивого множества вершин $X_I \subset X$. Задачу нахождения максимальных независимых множеств иногда называют «упаковкой вершин».

3. МЕХАНИЗМЫ ВЫДЕЛЕНИЯ В ГРАФЕ НЕЗАВИСИМОГО ПОДМНОЖЕСТВА ВЕРШИН НА ОСНОВЕ МОДЕЛИРОВАНИЯ АДАПТИВНОГО ПОВЕДЕНИЯ МУРАВЬИНОЙ КОЛОНИИ

Пусть дан граф $G(V,U)$, где V – множество вершин, $|V|=n$, U – множество ребер, $|U|=m$. Построим граф $G_r=(U^r,V^r)$ – реберный для графа G . Множество вершин $U^r=\{u^r_j|i=1,2,\dots,n_j\}$ графа G_r соответствует множеству ребер U графа G . Множество ребер $V^r=\{v^r_i|i=1,2,\dots,m_i\}$ графа G_r соответствует множеству вершин V графа G .

Сформулируем задачу формирования в графе $G_r=(U^r,V^r)$ внутренне-устойчивого множества вершин $U^r_0 \subset U^r$ как задачу разбиения.

Необходимо разбить множество U^r графа G_r на два непустых и непересекающихся подмножества U^r_0 и U^r_2 , таких, что любые две вершины $u^r_i \in U^r_0$ и $u^r_j \in U^r_0$ не являются смежными, $U^r_0 \cup U^r_2 = U^r$, $U^r_0 \cap U^r_2 = \emptyset$. Пусть $|U^r_0|=n_0$, $|U^r_2|=n_2$, $n_0+n_2=n_j$. Критерий оптимизации – число вершин $F=n_0$ в подмножестве U^r_0 . Цель оптимизации – максимизация критерия F .

Для поиска решения задачи формируется звездный граф поиска решений $H(U^r \setminus S_0, E)$, определяемый вершиной S_0 , который состоит из всех ребер E , связывающих концевую точку S_0 с множеством всех вершин U^r (рис. 3). Вершина S_0 связывается со всеми вершинами множества U^r .

В процессе решения участвует коллектив агентов. Каждый из агентов a_k формирует свое решение – свое подмножество $U^r_{ok} \subset U^r$ вершин звездного графа H , такое, что любые две вершины $u^r_i \in U^r_{ok}$ и $u^r_j \in U^r_{ok}$ в графе $G_r = (U^r, V^r)$ не являются смежными. k – номер агента.

Обозначим множество ребер звездного графа H , связывающих вершины сформированного внутри устойчивого подмножества U^r_{ok} с вершиной S_0 как E_{ok} , $E_{ok} \subset E$.

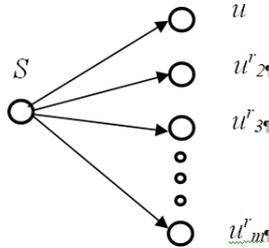


Рис. 3. Звездный граф H

Моделирование поведения агентов в процессе формирования каждым из них своего подмножества U^r_{ok} связано с распределением и учетом количества феромона на множестве E_{ok} ребер графа H . На начальном этапе на всех ребрах звездного графа H откладывается одинаковое (небольшое) количество феромона ζ/m , где $m = |E|$.

Процесс поиска решений итерационный. Каждая итерация l включает три этапа. На первом этапе каждый агент популяции конструктивным алгоритмом находит решение U^r_{ok} , рассчитывает оценку решения $\zeta_k(U^r_{ok})$ и значение степени пригодности полученного решения $\delta_k(U^r_{ok})$ (количество феромона, соответствующее оценке).

Первый этап осуществляется следующим образом.

Формирование решения (множества U^r_{ok}) осуществляется последовательно (пошагово). На каждом шаге t агент a_k применяет вероятностное правило выбора следующей вершины для включения ее формируемое множество $U^r_{ok}(t)$.

На первом шаге в формируемое множество $U^r_{ok}(t)$, где $t=1$, включается одна их вершин $u^r_j \in U^r$ графа H , смежная вершине S_0 .

На каждом шаге $t=1, 2, 3, \dots, n_l$ у агента a_k есть список вершин, уже включенных в формируемое множество – $U^r_{ok}(t-1)$ и список оставшихся (свободных) вершин $U^r_{ck}(t-1)$.

$$U^r_{ck}(t-1) = U^r \setminus U^r_{ok}(t-1), U^r_{ok}(t-1) \cup U^r_{ck}(t-1) = U^r.$$

На конечном шаге $t=n_0$ агентом a_k будет сформировано множество $U^{r_{ok}}(n_0)=U^{r_{ok}}$, $|U^{r_{ok}}(n_0)|=n_0$ такое, что любые две вершины $u^r_i \in U^{r_{ok}}$ и $u^r_j \in U^{r_{ok}}$ в графе $G_r=(U^r, V^r)$ не являются смежными. Шаг $t=n_0$ является конечным, если после его выполнения ни одна из вершин множества $U^{r_{ck}}(t)$ не может быть включена в $U^{r_{ok}}(t)$.

На шаге t формируется множество $U^{r_{nk}}(t) \subset U^{r_{ck}}(t-1)$ вершин, не смежных вершинам множества $U^{r_{ok}}(t-1)$ (рис. 4).

Вероятность включения в формируемое агентом a_k на базе звездного графа H множество $U^{r_{ok}}(t)$ вершины $u^r_j \in U^{r_{nk}}(t)$, не имеющей в графе $G_r=(U^r, V^r)$ связей с вершинами множества $U^{r_{ok}}(t-1)$, пропорциональна суммарному количеству феромона на ребре, связывающему вершину $u^r_j \in U^{r_{nk}}(t)$, $U^{r_{nk}}(t) \subset U^r$ с вершиной S_0 в графе H .

Формируется множество ребер $E_{ok}(t-1) \subset E$ звездного графа H , связывающих вершины сформированного подмножества $U^{r_{nk}}(t-1)$ с вершиной S_0 .

Агент просматривает все ребра $(S_0, u^r_j) \in E_{ok}(t-1)$, $u^r_j \in U^{r_{nk}}(t-1)$. Для каждого ребра $e_j \in E_{ok}(t-1)$ рассчитываются два параметра:

ϕ_j – суммарное количество феромона на ребре e_j (в массиве коллективной эволюционной памяти КЭП₀);

s_j – число связей на графе $G_r=(U^r, V^r)$ между u^r_j и $U^{r_{ck}}(t-1)$.

По формуле (1) – при мультипликативной свертке, либо по формуле (2) – при аддитивной свертке определяется потенциальная степень предпочтения F_{jk} включения u^r_j

$$F_{jk} = (\phi_j) / (s_{jk} + 1)^\beta \quad (1)$$

$$F_{jk} = (\alpha \phi_j) + (\beta / (s_{jk} + 1))^\beta, \quad (2)$$

где α, β – управляющие параметры, которые подбираются экспериментально.

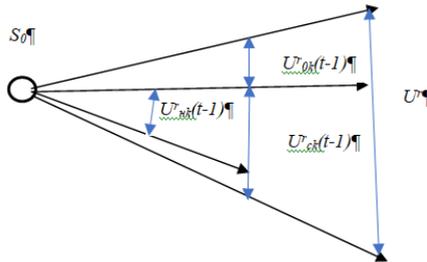


Рис. 4. Формирование множества $U^{r_{nk}}(t-1)$ вершин, не смежных с вершинами множества $U^{r_{ok}}(t-1)$

Вероятность $P_{ik}(t)$ включения вершины $u'_j \in U'_{nk}(t)$ в формируемое $U'_{ok}(t)$, определяется следующим соотношением:

$$P_{ik}(t) = F_{ik} / \sum_i F_{ik} . \quad (3)$$

Агент с вероятностью $P_{ik}(t)$ выбирает одну из вершин, которая включается в подмножество $U'_{ok}(t)$ и исключается из подмножества $U'_{ck}(t-1)$.

Количество φ_k откладываемого агентом a_k феромона на ребрах $E_{ok} \subset E$ звездного графа H , связывающих вершины $u'_j \in U'_{ok}$ сформированного внутри устойчивого подмножества U'_{ok} с вершиной S_o , пропорционально числу вершин n_o сформированного подмножества U'_{ok} . Чем больше n_o , тем больше феромона φ_k будет отложено агентом a_k на каждом ребре $(S_o, u'_j) \in E_{ok}$ звездного графа H , построенного на вершинах множества U'_{ok} . Следовательно, с большей вероятностью агенты будут выбирать такие вершины при синтезе собственного внутренне-устойчивого подмножества.

В работе используется циклический (ant-cycle) метод муравьиных систем. В этом случае рассчитанное агентом количество феромона $\delta_k(U'_{ok})$ предварительно откладывается в буферном массиве коллективной эволюционной памяти (КЭП_o) в ячейках соответствующих ребрам множества $E_{ok} \subset E$.

На втором этапе, после полного формирования всеми агентами решений на текущей итерации, феромон ω_j , накопленный в j -ой ячейке в буферном массиве КЭП_o, добавляется в каждую j -ю ячейку основного массива коллективной эволюционной памяти КЭП_o.

$$\phi_j = \phi_j + \omega_j.$$

Для избежания преждевременной сходимости используется отрицательная обратная связь в виде испарения феромона. После отложения феромона в основной массив коллективной эволюционной памяти КЭП_o, на третьем этапе происходит общее испарение феромона на множестве ребер E звездного графа H в соответствии с формулой:

$$\phi_j = \phi_j \cdot (1 - \rho), \quad (5)$$

где ϕ_j – уровень феромона на ребре $e_j \in E_{ok}$, ρ – коэффициент обновления.

После выполнения всех действий на итерации находится агент с лучшим решением, которое запоминается. Далее осуществляется переход на следующую итерацию. Временная сложность этого алгоритма зависит от времени жизни колонии l (число итераций), количества вершин графа n и числа муравьев m , и определяется как $O(l \cdot n^2 \cdot m)$.

После формирования в реберном графе $G_r=(U',V)$ графа $G(X,U)$ внутренне-устойчивого множества вершин $U'_0 \subset U'$ осуществляется переход от графа G_r к исходному графу G для построения паросочетания или выделения в графе клики.

Для преодоления локального барьера, используются подходы, основанные на сочетании различных видов эволюции. В частности эффективность показала композитная архитектура бионического поиска, заключающаяся в последовательной работе муравьиного и эволюционного алгоритма нахождения максимального паросочетания, рассмотренного в работе [11].

4. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Алгоритм нахождения паросочетания в графе реализован в виде программы **ПАР**.

Исследование программы **ПАР** проводились на примерах с известным оптимумом F_{opt} по методологии **ВЕКУ** [13].

Контрольные примеры для исследования содержали до 1000 вершин.

Для оценки качества используется безразмерная величина $\xi = F_{opt}/F$, где F – оценка решения полученного, полученного с помощью разработанного алгоритма. Зависимости показателя качества ξ от числа итераций и размера популяции представлены на рис. 5 и рис. 6.

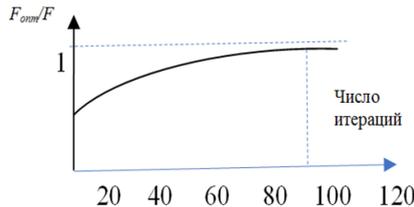


Рис. 5. Зависимость показателя ξ от числа итераций

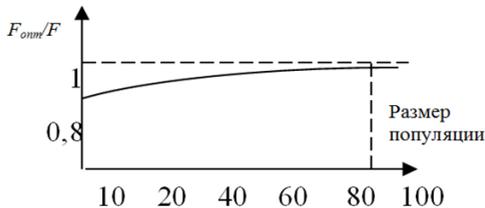


Рис. 6. Зависимость показателя ξ от размера популяции

Временная сложность алгоритма $O(n^2)$.

Сравнение с известными алгоритмами показало, что при меньшем времени работы новый алгоритм дает более качественные решения.

5. ЗАКЛЮЧЕНИЕ

В работе рассматривается поисковый популяционный алгоритм. Для поиска решения задачи в отличие от канонической парадигмы используется звездный граф поиска решений.

Основными достоинствами разработанного алгоритма являются:

- Низкая трудоемкость: временная сложность алгоритма $O(n^2)$.
- Реализация алгоритма отличается быстродействием и точностью вычислений.

Полученные в работе результаты можно использовать при решении оптимизационных задач, таких как: задачи о выделении внутренне-устойчивых множеств в графе, максимальной клике, проблема определения паросочетаний, раскраске графов и составление расписаний, задаче о назначениях.

Литература

1. Андерсон Д. Дискретная математика и комбинаторика. – М.: Вильямс, 2003.
2. Кормен К., Лейзерсон Ч., Ривест Р. Алгоритмы, построение и анализ. – М.: МЦМНО, 2000.
3. Свами М., Тхуласираман К. Графы, сети и алгоритмы. – М.: Мир, 1984.
4. Лебедев Б.К., Лебедев О.Б. Гибридный биоинспирированный алгоритм размещения базовых стандартных библиотечных элементов при проектировании топологии полузаказной СБИС // Известия ЮФУ. Технические науки. – 2019. – № 3. – С. 97-110.
5. Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учебное пособие. – 2-е изд. – М: Изд-во МГТУ им. Н.Э. Баумана, 2016. – 448 с.
6. Лебедев О.Б. Покрытие на основе методов роевого интеллекта // Проблемы разработки перспективных микро- и наноэлектронных систем: Сборник трудов под общ. ред. академика РАН А.Л. Стемпковского. – 2016. – С. 187-193.
7. Лебедев О.Б. и др. Распределение ресурсов на основе гибридных моделей роевого интеллекта // Научно-технический вестник информационных технологий, механики и оптики. – 2017. – № 6. – С. 1063-1073.
8. Lebedev O.B. Hybrid bioinspired algorithm of 1.5 dimensional bin-packing // Proceedings of the Third International Scientific Conference «Intelligent Information Technologies for Industry (ITI'18)». – 2018. – P. 254-263.

9. Лебедев О.Б. Модели адаптивного поведения муравьиной колонии в задачах проектирования. – Изд-во ЮФУ. Таганрог, 2013. – 199 с.
10. Лебедев О.Б. Ко-эволюционный алгоритм разбиения // Проблемы разработки перспективных микро- и нанoeлектронных систем: Сборник трудов под общ. ред. академика РАН А.Л. Стемпковского. – 2020. – № 3. – С. 79-86.
11. Лебедев Б.К., Лебедев О.Б. Муравьиные алгоритмы разбиения, использующие представление задачи, отличные от канонического // Вестник Ростовского государственного университета путей сообщения – Ростов-на-Дону: Изд-во РГУПС. – 2016. – № 3 (63). – С. 42-47.
12. Лебедев Б.К., Лебедев О.Б., Жиглатый А.А. Размещение элементов СБИС на основе моделей роевого интеллекта // Проблемы разработки перспективных микро- и нанoeлектронных систем. Сборник трудов под общ. ред. академика РАН А.Л. Стемпковского. – М.: ИППМ РАН, 2020. – С. 118-126.
13. Cong J., Romesis M., Xie M. Optimality, Scalability and Stability Study of Partitioning and Placement Algorithms // Proc. of the International Symposium on Physical Design. – Monterey, CA, 2003. – P. 88-94.

РЕШЕНИЕ ЗАДАЧ ТРАНСПОРТНОГО ТИПА С ИСПОЛЬЗОВАНИЕМ БИОИНСПИРИРОВАННЫХ МЕТОДОВ

*Щеглов С.Н., к.т.н., доцент
Южный федеральный университет
e-mail: srg_sch@mail.ru*

1. ВВЕДЕНИЕ

Среди приложений математического программирования большое место занимают проблемы производственно-транспортного типа. Задачи транспортного типа занимают особое место в классе приоритетных направлений исследования информационных технологий, так как решение транспортных задач имеет актуальное значение в промышленности, на транспорте, в системах связи и других отраслях народного хозяйства [1–3].

Основное место среди прикладных задач транспортного типа, занимают задачи построения транспортных маршрутов, которые позволяют до минимума сократить пробег транспортных средств или минимизировать затраты на перевозку грузов. Маршрутизация перевозок – это наиболее совершенный способ организации потоков грузов с предприятий, оказывающий существенное влияние на ускорение оборота транспорта при рациональном и эффективном его использовании.

Для данного класса комбинаторных задач, отсутствуют эффективные классические методы и алгоритмы решения. Эти задачи характеризуются конечным, но весьма большим числом возможных решений. Их можно поставить как задачи целочисленного программирования, но и в этом случае отсутствуют эффективные алгоритмы.

Поэтому, разработка методов и алгоритмов для решения задач транспортного типа, осуществляющаяся на протяжении многих лет, является по-прежнему, актуальной проблемой.

Моделирование бионических процессов в естественных и искусственных системах в настоящее время связано с созданием новых моделей и методов, направленных на решение сложных проблем. Для решения задач транспортного типа на основе бионического поиска, разработаны новые структурные схемы и алгоритмы, модифицированы базисные структуры оптимизационного процесса, которые позволяют расширить область поиска решений без увеличения времени ра-

боты и сократить преждевременную сходимость алгоритмов. Далее рассмотрим применение бионического поиска на примере решения задач об экстремальных путях.

2. ПОСТАНОВКА ЗАДАЧИ

Решение задач данного типа сводится к решению задачи о нахождении кратчайшего пути на ориентированном графе [3].

Задача о кратчайшем пути в графе в общем виде может быть сформулирована следующим образом.

Дан граф $G=(X, U)$, каждому ребру этого графа приписаны веса, задаваемые матрицей $C = [c_{i,j}]$. Задача о кратчайшем пути состоит в нахождении кратчайшего пути от заданной начальной вершины $s \in X$ до заданной конечной вершины $t \in X$, при условии, что такой путь существует, т.е. при условии $t \in R(s)$, где $R(s)$ – множество, достижимое из вершины s .

Можно выделить два основных варианта представления решения задачи: кодирование в виде списка вершин или кодирование в виде списка ребер. Применение первого варианта представления решения является неэффективным в виду большого количества получаемых нереальных решений. Поэтому наиболее работоспособным, остается вариант представления решения задачи о кратчайшем пути в графе в виде матрицы весов.

$$C = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1m} \\ c_{21} & c_{22} & \dots & c_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ c_{n1} & c_{n2} & \dots & c_{nm} \end{bmatrix},$$

где c_{ij} – веса, приписанные ребру графа.

Постановка задачи позволяет использовать различные множества альтернативных решений для получения квазиоптимальных результатов.

Поскольку решение задачи о кратчайшем пути является творческим процессом, то важной ее частью является построение и выбор эффективной архитектуры алгоритмов. Поэтому, предлагается использовать бионические (модифицированные генетические, эволюционные) методы поиска оптимальных решений с точки зрения лица принимающего решение (ЛПР) базирующиеся на комплексных и эволюционных критериях, адаптированные для решения задач данного типа. [2–6].

Рассмотрим функцию оценки пригодности. Функция оценки пригодности (моделирующая функция) необходима для определения качества, ценности индивидуальности. Очевидно, что моделирующая функция создается с учетом специфики конкретной задачи.

Для задачи о кратчайшем пути значение целевой функции (ЦФ) будет равно [3]:

$$F = \sum_{i,j=0}^n c_{ij}(x_{ij}) \rightarrow \min .$$

В процессе решения задачи необходимо минимизировать моделирующую функцию. Чем меньше значение функции, тем более ценным является решение.

3. ПРИМЕНЕНИЕ БИОИНСПИРИРОВАННОГО ПОИСКА ДЛЯ РЕШЕНИЯ ЗАДАЧ ОБ ЭКСТРЕМАЛЬНЫХ ПУТЯХ

Основной трудностью решения задачи об экстремальном пути с большим числом локальных оптимумов является предварительная сходимость алгоритмов, т.е. попадания решения в локальный оптимум.

При решении задач об экстремальных путях, эффективно используют стратегии, концепции, методы и механизмы моделирования на основе интегрированного поиска. В связи с этим предлагается схема интегрированного поиска для решения задач транспортного типа, приведенная на рис. 1 [3–7]. В блоке предварительного решения возможно использование «быстрых» алгоритмов (неявного перебора, метод ветвей и границ и т.п.) для получения первоначального решения. Далее, на основе знаний и опыта экспертов (блоки ЭС, БД, БЗ), происходит выбор наиболее приемлемого подхода для решения поставленной задачи. ИПА – алгоритмы, инспирированные природными системами. АС – модифицированный алгоритм Ant Colony, РИ – алгоритм на основе роевого интеллекта, БА, ГА, ЖА, ЭА и МО – бионический, генетический, жадный, эволюционный и моделирования отжига алгоритмы решения транспортных задач. Затем происходит оценка эффективности, полученного решения.

В случае получения удовлетворительного результата окончательный вариант выдается пользователю (заказчику). Иначе происходит изменение управляющих параметров, процесс повторяется либо до достижения критерия останова, либо получения приемлемого решения.

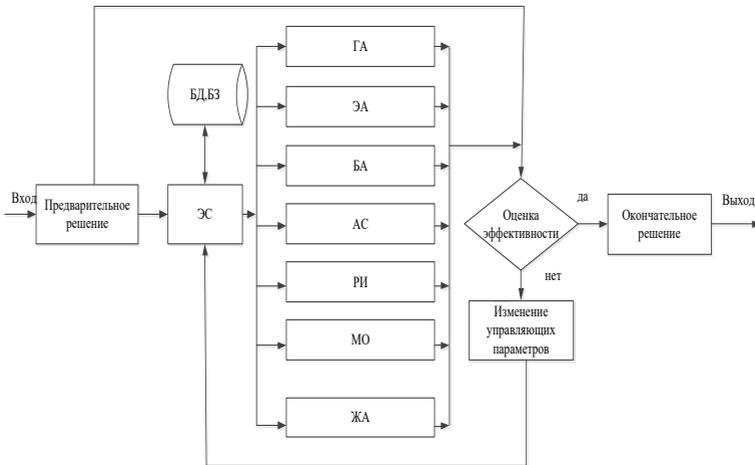


Рис. 1. Упрощенная схема интегрированного поиска

Данный подход позволяет распараллелить процесс нахождения приемлемого решения для задач большой размерности.

Рассмотрим более подробно применение бионического поиска, использованного в приведенной методике

Бионический поиск (БП) – это последовательное преобразование одного конечного нечеткого множества альтернативных решений в другое. Само преобразование называется алгоритмом поиска или ГА.

К достоинствам применения бионического алгоритма (БА) для решения задачи об экстремальном пути можно отнести:

Возможность выполнения двух видов поиска: эволюционного (ВСА – одной генерации $\approx O(n)$) и генетического (ВСА – одной генерации $\approx O(n) - O(n^3)$). Кроме того, выбор начальных решений осуществляется из «оптимизационных» методов нахождения кратчайшего пути.

Бионические алгоритмы, по мнению автора, дадут лучшие решения, т.е. хорошая приспособленность к внешней изменяющейся среде.

БА состоит в параллельной генерации наборов квазиоптимальных альтернативных решений с возможной «миграцией» решений между этими наборами.

Бионический алгоритм (БА) представляет собой кортеж:

БА= \langle ME, POP, OS, ГА, ЭА, F, ОР, критерий останова \rangle ,
 где ME – модели эволюции; POP = (p01, ..., p0λ) – исходная популяция; p0i – решение задачи, представленное в виде хромосомы;

OS – оператор отбора; ГА – генетический алгоритм; ЭА – эволюционный алгоритмы нахождения кратчайшего пути; F – функция пригодности; ОР – оператор редукции, корректирующий размер популяции альтернативных решений.

Структурная схема бионического поиска для задачи об экстремальном пути представлена на рис. 2.

В настоящее время существует большое количество различных модифицированных генетических, эволюционных алгоритмов, операторов отбора, редукции, возможности, применения которых широко известны для решения рассматриваемой задачи. Поэтому, более подробно остановимся на следующих блоках.

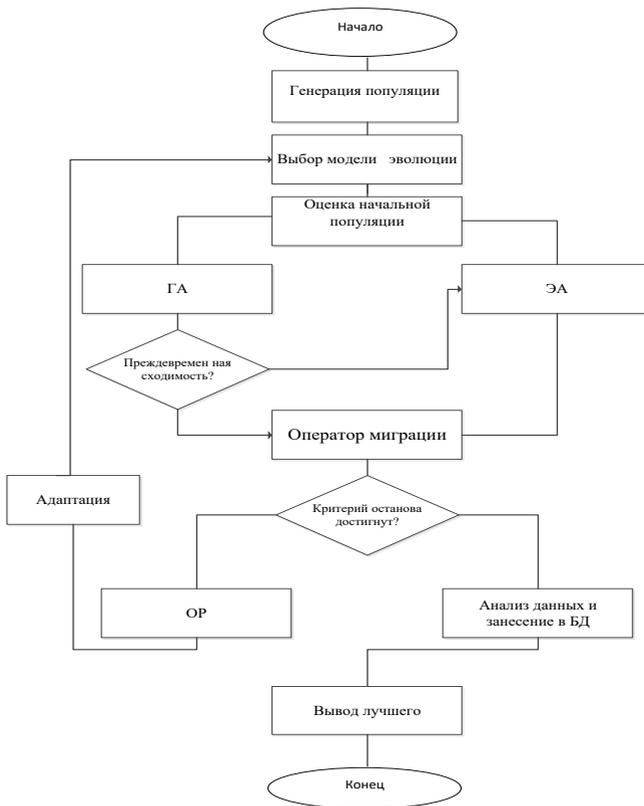


Рис. 2. Обобщенная схема бионического поиска для задачи об экстремальном пути

Блок «Выбор модели эволюции». В настоящее время считается важным объединение всех видов и моделей эволюций в интегрированную многоуровневую модель [5–8]. На рис. 3 приведена условная упрощенная интегрированная схема эволюции [8]. Следует отметить, что блоки 1–5 соответствуют различным схемам моделей эволюции.

Блок адаптации. Предназначен для выбора и реализации различных стратегий и механизмов адаптации. Еще одно предназначение данного блока состоит в настройке и изменении порядка использования и применения различных генетических операторов и схем поиска. Результаты работы блока адаптации оказывают непосредственное влияние на процесс перестройки текущей популяции альтернативных решений и создания на ее основе новой популяции.

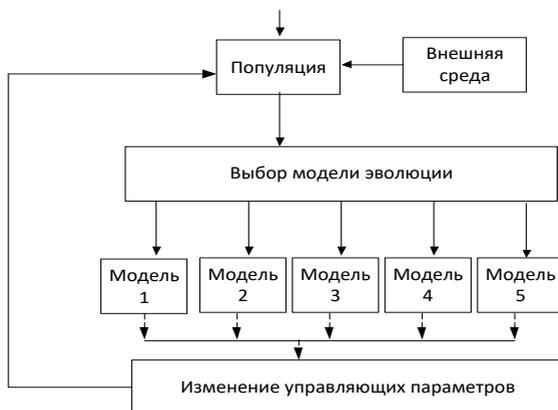


Рис. 3. Условная упрощенная интегрированная схема эволюции

Блок анализа данных и занесение в БД. Данный блок собирает и анализирует решения, получаемые в процессе выполнения алгоритмов. Каждому решению (индивиду) в результате проведенного анализа присваивается определённый ранг (перспективное, неперспективное, тривиальное и др.). При этом принимаются во внимание задаваемые на входе алгоритма ограничения области поиска. Проводимое в рассматриваемом блоке ранжирование текущей популяции альтернативных решений позволяет повысить эффективность бионического поиска за счет большей структурированности множества альтернативных решений и дает возможность динамического регулирования направления поиска. Таким образом, появляется дополнительный инструмент для самоадаптации и настройки параметров бионического поиска.

4. ВЫЧИСЛИТЕЛЬНЫЕ ЭКСПЕРИМЕНТЫ

Целью экспериментальных исследований – является анализ скорости сходимости разработанных алгоритмов.

В качестве критерия оптимальности алгоритмов используется минимальное значение целевой функции. При этом рассматривается определенное количество итераций алгоритма (число генерируемых поколений).

Проведение экспериментальных исследований преследовало выяснение следующих вопросов: адекватность механизмов бионического поиска для решаемых задач и эффективность предложенных алгоритмов.

Исследование эффективности предложенных алгоритмов заключается в определении пространственной и временной сложности и сравнении полученных бионическими алгоритмами результатов с оптимальными или, если нахождение оптимального результата невозможно, с результатами, полученными другими алгоритмами [10-12]. По выше предложенным и описанным алгоритмам, был разработан комплекс программ, который позволяет эффективно использовать преимущества генетического и эволюционного инструментария.

Исследование механизмов бионического поиска заключалось в определении влияния управляющих параметров, таких как тип селекции, тип кроссинговера, вероятность переноса гена из одной хромосомы в другую (для модифицированного кроссинговера), вероятность мутации и размер популяции.

Основными этапами экспериментальных исследований являются:

- проведение серии экспериментов для фиксированных значений общих параметров разработанных алгоритмов;
- проведение серии экспериментов с заданными изменяемыми параметрами моделей и алгоритмов;
- анализ экспериментальных данных;
- проведение серии экспериментов для различных тестовых функций и их анализ;
- определение оптимальных и квазиоптимальных параметров алгоритмов решения задач об экстремальном пути на графе.

С целью определения оптимальных значений управляющих параметров были произведены серии экспериментов, в которых исследовалось поведение целевой функции на одних и тех же наборах входных данных для различных значений варьируемых параметров:

- генетического поиска (ГП);
- вероятность применения ГО (ОК, ОИ);

- число итераций ГП;
- эволюционного поиска (ЭП);
- вероятность применения оператора мутации;
- число итераций ЭП.

Эксперименты проводились на различных тестовых примерах. Значения вероятностей генетических операторов изменялись дискретно от 0% до 100% с шагом 10%. Размер популяции также изменялся дискретно от 0 до 200 с шагом 20. Уровень миграции для параллельного бионического алгоритма принимал все свои возможные значения. Для исследования вероятности ОК значение вероятности для ОМ было установлено равным 0%, для исследования вероятности ОМ значение вероятности для ОК было установлено равным 0%, размер популяции при этом был равен 100 особям. Выход из процедуры осуществлялся при сходимости алгоритмов к некоторому локальному оптимуму (минимуму). При исследовании поведения ЦФ в зависимости от варьирования вероятностей применения генетических операторов учитывалась также стратегия моделирования самого ГО. В ходе проведения экспериментальных исследований оценивались зависимости времени работы алгоритма от числа генераций генетического и эволюционного поисков.

Исходными данными для работы программного комплекса нахождения решения задачи об экстремальном пути являются: количество вершин и ребер графа, матрицы смежности, размер популяции, количество поколений (итераций), вероятность применения генетических операторов.

Были проведены экспериментальные исследования влияния параметров бионического алгоритма (БА) и параллельного бионического алгоритма (ПБА) на изменение значений целевой функции (ЦФ) и времени работы: количество итераций NG , размер популяции N_p , вероятности кроссинговера, мутации и инверсии.

Параметры алгоритмов, при которых проводились исследования, были следующими:

- простой генетический алгоритм (ПГА): количество итераций – 100, размер популяции – 50, вероятность ОК – 90%, вероятность ОМ – 15%, селекция – случайная, отбор – элитный;
- бионический алгоритм (БА): число итераций – 100, размер популяции – 50, вероятность ОК – 90%, вероятность ОМ – 15%, селекция – случайная, отбор – элитный;
- параллельный бионический алгоритм (ПБА): число итераций – 100, размер популяции – 50, вероятность ОК – 90%, вероятность ОМ – 15%, селекция – случайная, отбор – элитный, количество подпопуляций – 4, уровень миграций – средний.

Была проведена серия экспериментов для разного набора тестовых примеров, различающихся количеством вершин графа. Для каждого алгоритма на каждом тестовом примере было произведено 3 запуска и на их основе определено среднее время работы алгоритма. Усредненные результаты экспериментов на рис. 4, 5.

Результаты исследований позволяют сделать вывод о том, что временная сложность рассмотренных алгоритмов бионического поиска не выходит за пределы полиномиальной зависимости, и может быть выражена формулой: $O(N^2)$, где N – число вершин графа (размер решаемой задачи).

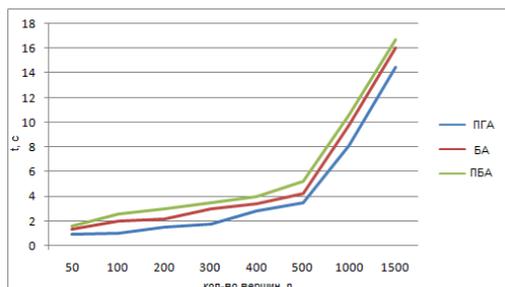


Рис. 4. Графики зависимости времени решения задачи о кратчайшем пути от числа вершин

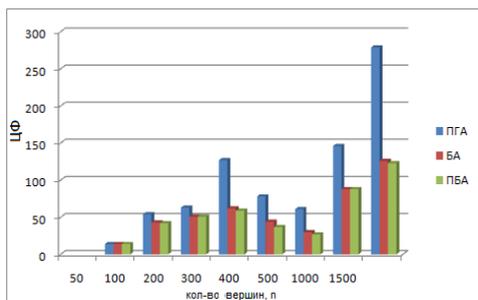


Рис. 5. Гистограмма зависимости изменения ЦФ от количества вершин графа для задачи о кратчайшем пути

5. ЗАКЛЮЧЕНИЕ

В заключение следует отметить, что в данной работе представлен интегрированный подход к решению задач транспортного типа. Приведены требования к построению алгоритмов решения задач транспортного типа. Показаны схема интегрированного поиска и ус-

ловная упрощенная интегрированная схема эволюции. Представлены некоторые результаты работы программного комплекса нахождения решения задачи об экстремальном пути. Рассмотренный подход позволяет строить алгоритмы решения задач транспортного типа с локальными оптимумами за полиномиальное время. Сложность алгоритмов имеет в среднем квадратичный порядок.

Литература

1. Чернышев Ю.О., Басова А.В., Полуян А.Ю. Решение задач транспортного типа генетическими алгоритмами: моногр. – Ростов-на-Дону: Изд-во ЮФУ, 2008. – 73 с.
2. Курейчик В.М., Курейчик В.В. Эволюционные, синергетические и гомеостатические стратегии в искусственном интеллекте: состояние и перспективы. Новости искусственного интеллекта. – 2000. – № 3. – С. 39-67.
3. Kureichik V.V., Kravchenko Y.A. Bioinspired algorithm applied to solve the travelling salesman problem // World Applied Sciences Journal. – 2013. – No. 22 (12). – P. 1789-1797.
4. Kurejchik V.V., Kurejchik V.M. On genetic-based control // Автоматика и телемеханика. – 2001. – № 10. – С. 174-187.
5. Бова В.В., Курейчик В.В. Интегрированная подсистема гибридного и комбинированного поиска в задачах проектирования и управления // Известия ЮФУ. Технические науки. – 2010. – № 12 (113). – С. 37-43.
6. Курейчик В.В., Курейчик Вл.Вл. Архитектура гибридного поиска при проектировании. Известия ЮФУ. Технические науки. – 2012. – № 7 (132). – С. 22-27.
7. Гладков Л.А., Гладкова Н.В. Новые подходы к построению систем анализа и извлечения знаний на основе гибридных методов // Известия ЮФУ. Технические науки. – 2010. – № 7 (108). – С. 146-153.
8. Kureichik V.V., Kureichik V.M., Sorokoletov P.V. Analysis and a survey of evolutionary models // Journal of Computer and Systems Sciences International. – 2007. – Vol. 46, No. 5. – P. 779-791.
9. Курейчик В.М., Курейчик В.В., Родзин С.И. Модели параллелизма эволюционных вычислений // Вестник Ростовского государственного университета путей сообщения. – 2011. – № 3. – С. 93-97.
10. Бова В.В., Курейчик В.В., Лежебоков А.А. Многоуровневый алгоритм решения задач транспортной логистики на основе методов роевого интеллекта // Вестник ростовского государственного университета путей сообщения. – 2013. – № 3 (51). – С. 113-118.
11. Запорожец Д.Ю., Курейчик В.В. Гибридный алгоритм решения задач транспортного типа // Известия ЮФУ. Технические науки. – 2013. – № 7 (144). – С. 80-85.
12. Курейчик В.В., Курейчик В.М., Родзин С.И. Теория эволюционных вычислений. – М.: Физматлит, 2012.

ABSTRACTS

Plenary reports

ABOUT THE SCIENTIFIC HERITAGE OF VALERY TARASSOV

A.A. Averkin, PhD

e-mail: averkin2003@inbox.ru

*The Federal Research Center “Computer Science and Control”
of Russian Academy of Sciences,
Laboratory of AI, neurotechnologies and business analytics
Plekhanov Russian University of Economics
Center for Artificial Intelligence of the University “Dubna”*

Valery Tarassov was one of the leading national experts in the field of AI. Valery Tarassov obtained fundamental results in the field of AI methodology, system approach in AI and synergetic AI, semiotic modeling, agent theory and multi-agent systems, theory of fuzzy sets, soft computing and measurement. He actively developed and implemented such areas as intelligent computer-aided design systems virtual enterprises, organizational design, enterprise modeling and re-engineering, knowledge management and methods for creating networked and intelligent enterprises. He developed a number of innovations in the field of engineering education and staff training, primarily based on intelligent and networking technologies, open education and virtual departments.

Keywords: synergetic artificial intelligence, semiotic modeling, agent theory and multi-agent systems, theory of fuzzy sets, soft computing and measurement.

BIOINSPIRED MODELS OF BEHAVIOR. SYSTEMIC CRISIS OR NEW HORIZONS?

V.E. Karpov, PhD

e-mail: karpov_ve@mail.ru

NRC Kurchatov Institute

The paper considers the problematic areas of such a direction of research as bioinspired behavioral models. It is shown how the ideas about the goals and methods of research have been transformed. It is shown that the reason for the “modesty” of achievements lies in the fact that this direction has not become truly interdisciplinary. Along with examples of research disunity, the paper gives an example of a different kind. We are talking about a paradigm of models of social behavior, for which close interdisciplinarity is fundamentally important. At the same time, it is shown that there are fundamental problems in this direction, connected primarily with the problems of mutual understanding between “biologists” and “cybernetics”. We are talking about fundamental differences in determining the criteria for the expediency of behavior, in understanding what behavior is in general.

Keywords: behavior models, bioinspiration, animat, models of social behavior, artificial life.

INTEGRATED AND HYBRID SYSTEMS OF ARTIFICIAL INTELLIGENT: METHODOLOGICAL PROBLEMS AND ISSUES OF TERMINOLOGY

*B.A. Kobrinskii, DSc, Professor
e-mail: kba_05@mail.ru*

*The Federal Research Center "Computer Science and Control"
of Russian Academy of Sciences,
Problem Artificial Intelligence Institute*

Modern requirements for intelligent decision support systems put before developers the task of a synergistic approach to the processing of knowledge extracted from different sources, including big data. However, in practice there is a big gap in the integration of systems implemented using different approaches, especially in the processing of linguistic and image information. The existing distinction between integrated and hybrid systems is of a formal nature. The paper considers various options for solving the problem of integration of intelligent systems.

Keywords: integrated systems, hybrid systems, semantic interoperability, ontologies, fragmentary addition, explainable artificial intelligence.

FROM FUZZY TO PROBABILISTIC OPTIMIZATION

*A.V. Yazenin, DSc, Professor
e-mail: yazenin.av@tversu.ru*

Yu.E. Egorova

j.e.egorova@gmail.com

*I.S. Soldatenko, Ph.D., Associate Professor
e-mail: soldis@tversu.ru
Tver State University*

The report is dedicated to the 100th anniversary of Professor Lotfi Zadeh. It presents the history of the emergence and development of probabilistic optimization. The main milestones in the development of the direction, the scientific articles of Professor Lotfi Zadeh, containing the results that influenced the evolution of this direction from fuzzy to modern probabilistic optimization, are noted. The architecture of the models is given and the classification of methods of probabilistic optimization is carried out for various decision-making principles under conditions of probabilistic, probabilistic uncertainty, the strongest and weakest t-norms describing the interaction of fuzzy parameters. The problem of choosing an investment portfolio in conditions of hybrid uncertainty is considered as an application of probability-probabilistic optimization.

Keywords: probabilistic optimization, probability measure, possibility, expected possibility, necessity, fuzzy random variable, indirect method, direct method, equivalent deterministic analog, equivalent stochastic analog, strongest t-norm, weakest t-norm, stochastic quasi-gradient method, minimal risk portfolio, quasi-efficient portfolio.

INTEGRATION OF FEATURE DESIGN METHODS AND KNOWLEDGE PROCESSING METHODS IN MACHINE LEARNING TASKS

*N.G. Yarushkina, DSc, Professor
e-mail: jng@ulstu.ru
Ulyanovsk State Technical University*

The purpose of the presented research is to search for effective integration of models, methods for constructing machine learning features, methods for evaluating data sets, methods for forming subject recommendation systems, methods for intelligent analysis of process dynamics in predictive analytics tasks, taking into account the requirements of trust intelligence in the design of industry solutions.

Keywords: machine learning, recommendation systems, intelligent analysis of process dynamics, deep feature synthesis.

Section 1

INTERPRETED CLASSIFIER MODEL BASED ON FUZZY LOGISTIC REGRESSION

A.I. Dolgiy, PhD

JSC NIIAS

A.E. Kolodenkova, DSc

Samara State Technical University

e-mail: anna82_42@mail.ru

S.M.Kovalev, DSc, Professor

e-mail: ksm@rfniias.ru

Rostov branch JSC NIIAS

In this article, the authors solve the problem of developing a generalized classifier model based on fuzzy logistic regression (log model), which has interpretability properties, that is, the ability to justify the process of deriving the results of the classifier in a way understandable to humans, as well as the ability to integrate the knowledge of specialists into the process of designing, training and adapting classifiers.

Keywords: fuzzy logistic regression, fuzzy mapping, classifier based on logistic regression.

PROBABILISTIC MODELS IN INFORMATION TECHNOLOGIES

A.V Ilyakhinskii, PhD

e-mail: ilyahinsky-aleks@bk.ru

Institute of Mechanical Engineering Problems RAS

V.N Bogatkov, PhD

e-mail: vnbgtk@mail.ru

Tver State Technical University

It is shown that the multivariate Dirichlet distribution has sufficient flexibility and realism, which favorably distinguishes it from statistical models based on the normal distribution. beta distributions, gamma distributions and Poisson distributions in describing processes in **complex** multilevel hierarchically organized systems

Keywords: statistical model, probability density function, Dirichlet distribution, entropy, self-organization.

**METHOD AND PARALLEL IMPLEMENTATION OF FUZZY
CLASSIFICATION OF OBJECTS WITH NON-SINGLETON
FUZZIFICATION**

S.A. Karatach

e-mail: karatach1998@yandex.ru

V.G. Sinuk, PhD

e-mail: vgsinuk@mail.ru

Belgorod State Technological University named after V.G. Shukhov

This article proposes a method for classifying objects whose attribute values contain linguistic information. The proposed method involves the use of fuzzy inference systems of logical type based on the decomposition theorem. Multithreaded and parallel implementations were performed, for which a comparative analysis of both the speed of the algorithm and the obtained accuracy for the KDD99 dataset was performed.

Keywords: fuzzy classification, fuzzy inputs, decomposition theorem, parallel technologies.

**COMPARISON OF THE DATA QUALITY OF GEOMAGNETIC
OBSERVATORIES INCLUDED IN THE WORLD INTERMAGNET
NETWORK USING BIG DATA METHODS**

A.G. Korobeynikov, DSc, Professor

e-mail: Korobeynikov_A_G@mail.ru

SPbF IZMIRAN

E.R. Muntyan, PhD

e-mail: ermuntyan@sfedu.ru

SFedU

This paper presents the results of solving the problem of comparing data on the geomagnetic field obtained from five geomagnetic observatories included in the INTERMAGNET – International Real-Time Magnetic Observatory Network. Graphs of the initial observational data for the period from January 1, 2018 to February 28, 2022 are presented. A brief analysis of these data is given.

Keywords: big data, matlab, geomagnetic observatories, intermagnet.

AUTOMATED NON-DESTRUCTIVE TESTING COMPLEX BASED ON ARTIFICIAL INTELLIGENCE METHODS

A.G. Korobeynikov, DSc, Professor

e-mail: Korobeynikov_A_G@mail.ru

SPbF IZMIRAN, ITMO University

V.L. Tkalich, DSc, Professor

e-mail: vera_leonidovna_tkalich@mail.ru

O.I. Pirozhnikova, PhD., Associate Professor,

e-mail: O.Pirozhnikova10@gmail.com

M.E. Kalinkina

e-mail: mariia_kalinkina@mail.ru

ITMO University

The work is devoted to the actual problem of creating domestic competitive complexes for non-destructive testing of metal structures operating at strategically important transport infrastructure facilities. The developed complex has the functions of artificial intelligence, which allows you to flexibly adjust the diagnostic parameters in the probing adaptive impact on the monitored object. The software of the complex used for computer simulation is based on the principles of machine learning when processing electrical signals from a piezoelectric transducer that have undergone a wavelet transform procedure. This complex is designed for the correct examination of the service life of metal structures used in their operation at nuclear reactors of icebreakers and in the railway transport infrastructure.

Keywords: non-destructive testing, artificial intelligence, flaw detection, monitoring object.

A HYBRID ALGORITHM FOR SEARCHING ANOMALIES IN TIME SERIES TAKING INTO ACCOUNT THE FEATURES OF THE SUBJECT AREA

D.S. Kurilo, Assistant of department

e-mail: dimokkurilo@gmail.com

V.S. Moshkin, PhD

e-mail: postforvadim@ya.ru

I.A. Moshkina, PhD

e-mail: i.timina@ulstu.ru

Ulyanovsk State Technical University

The paper describes the results of the development of a software system that implements the method of searching for anomalies using a neural network and their subsequent analysis based on intelligent systems.

Keywords: time series, neural networks, ontology, anomaly detecting.

A SOUND AND COMPLETE INFERENCE SYSTEM FOR PROPOSITIONAL FUZZY LOGIC WITH ESTIMATES

Gerald S. Plesniewicz, PhD
Moscow Power University
e-mail :salve777@mail.ru

We have defined a propositional logic LE whose sentences are estimates of truth estimates for statements of Zadeh's fuzzy propositional logic L. (Zadeh's logic for the algebra of fuzzy sets plays a role similar to the role of classical propositional logic for the algebra of ordinary crisp sets.). We have built a consistent and complete inference system for the LE logic of estimates.

Keywords: fuzzy logic, Zadeh's fuzzy logic, estimates for fuzzy logic statements, analytic tableaux.

APPLICATION OF TEMPORAL NEURO-FUZZY PETRI NETS IN INFORMATION-TECHNOLOGICAL SYSTEMS

S.A. Prokopenko
National Research University "MPEI"
Inter-state educational institution of higher education "Belarusian-Russian
university"
e-mail: puss95@yandex.by
A.V. Bobryakov, DSc, Professor
National Research University "MPEI"

Tools for modeling and analyzing processes in complex systems are found everywhere. The rapid development of information technology, high competition, a large amount of heterogeneous and different-quality information necessary for work are forcing small manufacturing enterprises to implement and use information technology systems for their work. There are no universal approaches to the creation of information technology systems. Each group of industries needs its own models and methods for management. Constantly tightening requirements for the quality of manufactured products, high competition, exactingness of the customer force us to change the established production processes. Effective management of production processes becomes more complex due to the influence of internal and external factors. For effective management and modeling, the existing approaches are no longer relevant, and it is necessary to modernize them using modern technologies.

Keywords: Petri nets, information technological systems, complex systems.

INVESTIGATION OF THE PROBLEM OF CONTROLLING THE ACETYLENE PRODUCTION PROCESS BASED ON FUZZY LOGIC

G.N. Sanaeva, PhD

e-mail: gsanaeva@nirhtu.ru

Novomoskovsk Institute MUCTR

V.N. Bogatkov, DSc

e-mail: vnbgtk@mail.ru

Tver State Technical University

A variant of the structure of the acetylene production management system based on the use of fuzzy logic is presented

Keywords: control, technological safety, acetylene production, fuzzy logic.

APPLICATION OF THE SPOT MODEL TO SOLVING INVERSE PROBLEMS AND ARTIFICIAL INTELLIGENCE

N.A. Simonov, PhD

e-mail: nsimonov@ftian.ru

Valiev Institute of Physics and Technology, RAS

The need to develop new approaches for modeling and solving artificial intelligence problems is associated with significant limitations for the use of traditional methods in solving practical problems. According to the author, one of the fundamental shortcomings of the existing neuron model and neural networks is the modeling of intelligence using mathematical operations with real numbers and the use of numerical methods. This, in principle, is inadequate to the nature of human thinking, which is based on mental representation and logical processing of information using mental images as basic elements. In this paper, we consider the possibility of using the spot model for solving inverse problems and for artificial intelligence problems. It is assumed that the spot model can represent any qualitative and quantitative information. An important property of the spot model is that mental images can be represented on its basis, which allow us to build a semantic space. This conclusion, as well as the proposed mathematical apparatus of L4 numbers, allow us to consider the developed theory of spots as a promising mathematical basis for creating a universal (strong) artificial intelligence.

Keywords: mental images; semantic information; computer reasoning; machine learning; Universal Artificial Intelligence

INFERENCE METHODS FOR MAMDANI AND TAKAGI-SUGENO FUZZY SYSTEMS WITH NON-SINGLETON FUZZIFICATION BASED ON FUZZY TRUTH VALUE

V.G. Sinuk, PhD
e-mail: vgsinuk@mail.ru
S.V. Kulabukhov
e-mail: qlba@ya.ru

Belgorod State Technological University named after V.G. Shukhov

The paper derives the result of logical inference for widely used fuzzy systems in case of non-singleton fuzzification. The use of the fuzzy-truth-values-based approach made it possible to reduce the computational complexity from exponential down to polynomial and to generalize the conditions for logical inference. The most commonly used defuzzification methods have been considered in application to the obtained inference results.

Keywords: fuzzy truth value, computational complexity, firing level.

BIFURCATION PRECURSORS DETECTION IN DYNAMICAL SYSTEMS BASED ON FUZZY MODEL STRUCTURE ANALYSIS

A.V. Sukhanov, PhD
JSC "NILAS"
Rostov State Transport University, Southern University
e-mail: a.suhanov@rfniias.ru

S.M. Kovalev, DSc, Professor
JSC "NILAS"

Rostov State Transport University
e-mail: ksm@rfniias.ru

I.G. Akperov, PhD, DSc, Professor
Southern University
e-mail: rector@iubip.ru

I.A. Olgezyer, PhD
JSC "NILAS"

Rostov State Transport University
e-mail: iohan@rfniias.ru

The paper considers the models having the double plasticity property, which allows them to model dynamical systems properly even when they have complex time-varying environment. Some of them are described in this paper. Simple computational experiment showing the importance is proposed. A future work is given at the end of the paper.

Keywords: Double plasticity, structure adaptation, anomaly detection, bifurcation.

AUTOMATED CONTROL OF THE GREEN TEA DRYING PROCESS USING A PREDICTIVE FUZZY-PID CONTROLLER

K.B. Pham

e-mail: phambang79520897405@gmail.com

V.N. Bogatkov, DSc, Professor

e-mail: vnbgtk@mail.ru

Tver State Technical University

This article presents the design and simulation of an automated control system (ACS) based on a fuzzy-PID controller algorithm to control the drying process of green tea in a conveyor belt dryer. This ACS simulation was carried out using Matlab and Simulink. In general, the results of numerical simulations show that the fuzzy-PID controller is stable and reliable in terms of input noise suppression.

Keywords: green tea drying, fuzzy-PID controller, ACS simulation, conveyor belt dryer.

FUZZY LOGICAL-LINGUISTIC ALGORITHM FOR DETECTING INCIDENTS IN CYBERPHYSICAL SYSTEMS

N.V. Chikalov

e-mail: nik.chikalow2011@yandex.ru

ITMO University

M.E. Berezin

e-mail: adfpbm@gmail.com

Saint Petersburg State University of Aerospace Instrumentation

Gatchin Y.A., DSc

ITMO University

e-mail: gatchin1952@mail.ru

Polyakov V.I., PhD

ITMO University

e-mail: v_i_polyakov@mail.ru

Incidents belong to the class of tasks that are not regulated by real-time rules, that is, an incident may already take place, but the selected model and algorithm have a low threshold of sensitivity to data changes, or erroneously issue a conclusion about the absence of an incident. Therefore, it is necessary to reduce the number of destructive impacts by switching to a new management strategy, which will be based on the principles of forecasting and warning. The purpose of the study is to improve methods for detecting incidents in cyber-physical systems. Object of research: processes of information interaction in cyber-physical systems. Subject of study: Incident detection algorithm in cyber-physical systems.

Keywords: incidents, cyber-physical system, fuzzy sets, logical linguistic algorithm.

Section 2

PREDICTIVE ANALYSIS OF SURFACE DEFORMATION OF MATERIALS BASED ON ACOUSTIC EMISSION DATA

A.A. Alexandrov

e-mail: alea@sfsedu.ru

A.V. Chernov, DSc, Professor

e-mail: cherno@sfsedu.ru

D.S. Polyanichenko

e-mail: poliani@sfsedu.ru

O.O. Kartashov, PhD

e-mail: okartashov@sfsedu.ru

M.A. Butakova, DSc, Professor

e-mail: mbutakova@sfsedu.ru

The Smart Materials Research Institute at the Southern Federal University

The report deals with the problem of predictive analysis of surface deformation of materials using machine learning. Surface plastic deformation can occur as a result of various mechanical processes, including the impact of a tool impacting the surface of a material being deformed. Such a process is inevitably accompanied by the appearance and propagation of sound waves of various ranges in the structure of the material. One of the means of passive acoustic data acquisition is electronic acoustic emission sensors that convert acoustic signals into electrical ones, the parameters of which are further used to assess the characteristics of deformation changes in the material. An approach to forecasting the data of time series of acoustic emission measurements based on the use of recurrent neural network models is proposed. The approach is based on transient detection of the appearance of surface deformation, preprocessing and transformation of time series for machine learning applications. By searching for hidden patterns in the acoustic emission time series data, the artificial neural network model with a long short-term memory provides interval prediction of the surface plastic deformation process with high accuracy. The main results of the study are deposited sets of pre-processed acoustic emission data, a method for estimating and predicting the parameters of the process of surface plastic deformation of the materials under study based on machine learning algorithms.

Keywords: machine learning, deep learning, acoustic emission.

APPROACH TO THE ANALYSIS OF PERSONAL CHARACTERISTICS OF USERS OF SOCIAL NETWORKS USING MACHINE LEARNING

I.A. Andreev

e-mail: ares-ilya@ya.ru

V.S. Moshkin, PhD

e-mail: postforvadim@ya.ru

N.G. Yarushkina., DSc

e-mail: jng@ulstu.ru

Ulyanovsk State Technical University

The paper presents an approach to determining the psychological characteristics of a user of social networks through the analysis of text messages in social networks. The task of psycholinguistic analysis is the task of binary classification. The approach proposed in the framework of the work is to classify the author's texts of the user using machine learning, while the results of the analysis of user surveys, as well as a set of author's text data from the pages of social networks, are used as training data. Two models were used as classifiers: the support vector machine and the random forest method. The AUC ROC model was used to evaluate efficacy. The best average results were shown by the SVM method when determining the values of the Neuroticism class (0.72 when objects were divided into training and test sets in the proportion of 70/30), the maximum results were SVM, the Extraversion class (0.97 at 70/30) and the Consciousness class (0.93 at 70/30).

Keywords: machine learning, social networks, psycholinguistic analysis.

PROCESSING OF NORMATIVE LEGAL ACTS USING NEURAL NETWORK MODELS

M.A. Anikin

e-mail: maxim.anikin1@yandex.ru

Bauman Moscow State Technical University

M.N. Koroleva, PhD, Associate Professor

e-mail: maria.svyatkina@gmail.com

Bauman Moscow State Technical University

STC Systema

This paper considers the possibilities of using modern machine learning methods in the field of natural language processing for the analysis of normative legal acts in Russian. A functional model of the text analysis is proposed. The results of the work of existing pre-trained models on a sample of normative legal acts are presented.

Keywords: machine learning, neural networks, normative legal act, natural language processing, artificial intelligence.

NEURAL NETWORK FOR BOOLEAN SATISFIABILITY PROBLEM

A.N. Aparnev

e-mail: apich238@gmail.com

Moscow Power Energetics Institute

One of the promising areas of AI is the neural-symbolic integration. Often, the symbolic component is fuzzy logic with t-norm. The hybrid system solving SAT problem was developed to determine how the choice of t-norm affects training process. It was shown, that choice of the Lukasiewicz t-norm slowing training process.

Keywords: Fuzzy logic, artificial neural networks, sat, neural-symbolic integration.

PREDICTING AN OPPONENT'S INTENTIONS IN AN ENVIRONMENT WITH OPPOSITION

V.I. Gorodetsky, DSc, Professor

e-mail: vladim.gorodetsky@gmail.com

JSC Eureca

The paper considers the problem of competition / rivalry between two complex systems operating in a common environment. The particular paper's concerns one of the fundamental tasks of this problem, which is predicting group behavior of the objects composing the opponent/adversary, its intentions and potential threat it poses for the opposite side. For this task, the paper proposes an approach to solving this task, as well as a generalized functional architecture of the system implementing this approach. The novelty of the proposed approach is that it uses the concept of a digital twin of the opponent's behavior, which is involved both at the stage of machine learning of the system for predicting the opponent's intentions, and at the decision-making stage. The methodological novelty of the approach lies in the use of a scenario model of the competitor's behavior as a means of structuring the group behavior of distributed objects that form a competing system.

Keywords: system intention, system goal, rivalry, opposition, group behavior, machine learning, prediction.

TRAINING OF RANDOM FORESTS WITH KERNEL SPLITS

D.A. Devyatkin

e-mail: devyatkin@isa.ru

Federal Research Center "Computer Science and Control" RAS

During the analysis of texts, images, structured data, it is often necessary to solve the problem of modeling discrete processes. The best accuracy of solving this problem could theoretically be achieved with discrete models. One of the most well-known ways to build those models with machine learning methods is Random Forest. The Random Forest is based on decision trees with one-dimensional decision stamps. Those trees can be built with the CART, C4.5, or ID3 algorithms. Random forests show high accuracy in various data and text mining tasks, but their applicability is limited in the case of multi-dimensional sparse data analysis. To analyze such data, decision trees with linear (oblique) decision stamps could be applied; however, most of the existing approaches to training such trees have low computational efficiency or are sensitive to the values of many hyperparameters. In addition, when training forests of such trees on small datasets, overfitting have a negative impact. Therefore, it is necessary to use various regularization that make it possible to find a compromise between the complexity of the resulting models and their generalizing ability. The paper proposes an algorithm for training random forests of decision trees with kernel decision stamps (linear, polynomial, Gaussian). The construction of these stamps is carried out by optimizing the loss function, like that used in the support vector machine. Therefore, computationally efficient algorithms implemented in the LibSVM, Liblinear and ThunderSVM libraries can be applied to train stamps. The experiments show that regularized random forests with kernel stamps are superior in accuracy and recall of classification to random forests of decision trees, as well as forests of oblique trees trained by other methods.

Keywords: Decision tree, random forest, kernel forest, SVM, regularization.

COMPARISON OF APPROACHES TO THE TEXT DESCRIPTIONS GENERATION

M.A. Ipatov

e-mail: mipatov0pls@ya.ru

A.S. Mokhov, Ph.D

e-mail: MokhovAS@mpei.ru

National Research University "MPEI"

This article is devoted to solving the problem of generating text descriptions for an online store using neural networks of the "Transformer" architecture. The pre-trained ruGPT3Medium and ruT5-large models were fine-tuned to create descriptions of laptops based on their characteristics. Studies have shown some advantages of the ruGPT3Medium-type model, and revealed a number of features of using such models to solve similar problems.

Keywords: natural language processing, text generation, neural networks, transformers, T5, GPT3.

PREDICTION OF THE COORDINATES OF A DYNAMIC OBJECT UNDER CONDITIONS OF INACCURACY OF MEASUREMENTS BY MEANS OF NEURAL NETWORK TECHNOLOGIES

B.V. Palyukh, DSc, Professor

e-mail: pboris@tstu.tver.ru

V.K. Kemaikin, PhD, Associate Professor

e-mail: vk-kem@mail.ru

A.S. Polygaev, postgraduate student

e-mail: a.polygaev@gmail.com

Tver State Technical University

The article deals with the issues of tracking moving objects in the secondary processing of a radar station using neural network technologies. A proposal for the architecture of a deep neural network for solving the problem of predicting the coordinates of a dynamic object under conditions of inaccurate measurements.

Keywords: artificial neural network, neural network with long short-term memory, Kalman filter, time series forecasting, measurement filtering, extrapolation.

A FRAMEWORK FOR INTERPRETABLE MACHINE LEARNING METHOD SELECTION

A.V. Suvorova, PhD

e-mail: asuvorova@hse.ru

A.V. Smirnova

HSE University

Interpretable Machine Learning (IML) methods provide valuable insights into model behaviour, but due to variety of available interpretation tools, the particular solution can be not the optimal fit to the needs of the target users. To support decision making in model building and exploration, we propose a modular framework for algorithm selection. The framework is based on the overview of properties of IML methods, available metrics and existing guidelines for explanation systems development and evaluation.

Keywords: explainable AI, quality metrics, algorithm selection.

WAYS TO IMPROVE THE TRAINING QUALITY OF NEURAL NETWORK DIAGNOSTIC MODELS OF COMPLEX TECHNICAL OBJECTS

M.V. Fedotov, PhD

e-mail: fedotov-mv@vnikti.com

JSC "Scientific-Research and Design-Technology Institute of Rolling Stock" (JSC "VNIKTI")

A.L. Sharapov

e-mail: sharapov-al@vnikti.com

JSC "Scientific-Research and Design-Technology Institute of Rolling Stock" (JSC "VNIKTI")

V.V. Grachev, DSc

e-mail: v_grach@mail.ru

Federal State Budgetary Educational Institution of Higher Education "Emperor Alexander I St. Petersburg State Transport University"

The options for improving the quality of training samples formation for neural network diagnostic models are studied on the example of diesel locomotive systems. The data applied was obtained owing to monitoring using analytical models and generative adversarial networks (GAN). The proposed solutions can significantly expand the range of tasks to be solved and improve the accuracy of technical diagnostics.

Keywords: diagnostics, technical condition, neural network diagnostic model, training sample, analytical model, generative adversarial network.

A COMPARATIVE STUDY OF DIAGNOSING PARKINSON'S DISEASE USING MACHINE LEARNING

Nekoula Haddad

e-mail: nekoula@inbox.ru

Bauman Moscow State Technical University

K.S. Myshenkov, DSc, Professor

e-mail: myshenkovks@bmstu.ru

Bauman Moscow State Technical University

Ghassan Haddad, DSc, Professor

e-mail: dr.eng.ghassan.haddad@gmail.com

Damascus University

Parkinson's disease (PD) is a long-term degenerative disease of the central nervous system. This study used two different datasets to classify PD using machine learning. The PD classification using the gait dataset performed better than classification using the keystone dataset, with an accuracy of 78% and 65% respectively.

Keywords: diagnostics, Parkinson's disease, machine learning, gradient boosting method, classification

**IMPLEMENTATION OF MACHINE LEARNING METHODS
FOR PREDICTION OF THE RISK OF MORTALITY AFTER CORONARY
BYPASS**

K.I. Shakhgeldyan, DSc, Professor

e-mail: carinash@vvsu.ru

Vladivostok State University of Economics and Service

Far Eastern Federal University

V.Yu. Rublev, cardiovascular surgeon, postgraduate student,

Far Eastern Federal University

e-mail: dr.rublev.v@gmail.com

B.I. Geltser, DSc, professor

e-mail: boris.geltser@vvsu.ru

Corresponding Member of Russian Academy of Sciences,

Far Eastern Federal University

B.O. Shcheglov

e-mail: b.shcheglov@mail.ru

Far Eastern Federal University

V.V. Kosterin

e-mail: kraiaasherdev@gmail.com

Far Eastern Federal University

S.N. Shcheglova, PhD, Associate Professor

e-mail: schegloff@mail.ru

North-Eastern State University

In this study, the authors conducted a comparative analysis of the influence of predictors on the risk of short-term postoperative mortality based on the quality metrics of multivariate and univariate logistic regression, stochastic gradient boosting, random forest and artificial neural network models using binarized and continuous data.

Keywords: predictive models, stochastic-gradient boosting, artificial neural network, features selection, postoperative short-term mortality.

Section 3

DEVELOPMENT AND RESEARCH OF A HYBRID ALGORITHM FOR SOLVING THE PLACEMENT PROBLEM

*G.E. Veselov, DSc, Associate Professor
e-mail: gev@sfnedu.ru*

*L.A. Gladkov, PhD, professor
e-mail: leo_gladkov@mail.ru*

*Yaser M.J., postgraduate student
e-mail: yasir_82@mail.ru
Southern Federal University*

The article deals with the problem of placing different-sized elements. The structure of a hybrid algorithm for solving the problem of placing various-sized elements is proposed, modified genetic operators, as well as components of the hybrid algorithm are developed. The hybrid algorithm is implemented in the form of an application program, and a series of computational experiments have been carried out.

Keywords: design automation, placement problem, hybrid methods, bioinspired algorithms, genetic algorithms, fuzzy control.

BIOINSPIRED PLANNING ALGORITHM TRAJECTORIES UNDER CONDITIONS OF PARTIAL UNCERTAINTY AND THE PRESENCE OF MOVING OBSTACLES

*G.E. Veselov, DSc, Associate Professor
e-mail: gev@sfnedu.ru*

*B.K. Lebedev, DSc, Associate Professor
e-mail: lebedev.b.k@gmail.com*

*O.B. Lebedev, Ph.D., Associate Professor
e-mail: lebedev.ob@mail.ru
Southern Federal University*

The paper describes a bioinspired trajectory planning algorithm under partial uncertainty, based on the integration of wave and ant algorithms. The process of laying the trajectory is carried out step by step. At each step relative to the current position of the moving object and moving obstacles, a zone is formed, within which all obstacles are localized using the radar, after which a separate section of the trajectory is built, which is a continuation of the previously constructed section. And the entire trajectory is a collection of individual sections. The time complexity of this algorithm depends on the lifetime of the colony l (number of iterations), the number of graph vertices n and the number of ants m , and is defined as $O(l \cdot n^2 \cdot m)$.

Keywords: trajectory planning, partial uncertainty, moving obstacles, wave algorithm, ant colony algorithm, hybridization.

BIOINSPIRED ALGORITHM FOR CONSTRUCTING A PAIRING

*O.B. Lebedev, Ph.D., Associate Professor
e-mail: lebedev.ob@mail.ru*

*A.A. Zhiglaty
e-mail: artemiy.zhiglaty@gmail.com
Southern Federal University*

A bioinspired method for solving a set of related combinatorial-logical problems on graphs is proposed: the formation of a graph matching, the selection of an internally stable set of vertices, and the selection of a graph clique. A modified ant colony paradigm is described that uses, in contrast to the canonical method, the mechanisms for generating solutions on the search space model in the form of a star graph. The time complexity of the algorithm, obtained experimentally, coincides with theoretical studies and for the considered test problems is $O(n^2)$.

Keywords: matching, internally stable set, clique, adaptation, modification, ant colony, search space models, star graph.

SOLUTION OF PROBLEMS OF TRANSPORT TYPE USING BIOINSPIRED METHODS

*S.N. Shcheglov, Ph.D., Associate Professor
e-mail: srg_sch@mail.ru
Southern Federal University*

In this paper we consider the use of search engine bionic methods for solving the transport type. For this class of combinatorial problems, no efficient classical methods and algorithms of solution. These tasks are characterized by a finite but very large number of possible solutions. They can be put as a problem of integer programming, but in this case there are no effective algorithms. Therefore, development of methods and algorithms for solving the transport type which is carried out for many years, is still actual problem. The formulation of the problem. Shows the scheme of integrated search, which allows you to parallelize the process of finding an acceptable solution for large scale problems. Reviewed structural diagram of bionic search for the problem of an extreme way. The results of computational experiments. The results allow to conclude that the time complexity of the considered algorithms bionic search does not go beyond polynomial dependence, and can be expressed by the formula: $O(N^2)$, where N is the number of vertices of the graph (the size of solvable problem).

Keywords: The algorithm, method, graph models, heuristics, structure, scheme, experiment, research, genetic search, adaptation, artificial intelligence, evolution, solution.

АВТОРСКИЙ УКАЗАТЕЛЬ

Аверкин А.Н.	12	Мохов А.С.	224
Акперов И.Г.	137	Мошкин В.С.	85, 177
Александров А.А.	168	Мошкина И.А.	85
Андреев И.А.	177	Мунтян Е.Р.	74
Аникин М.А.	182	Мышенков К.С.	268
Апарнев А.Н.	189	Ольгейзер И.А.	137
Березин М.Е.	156	Палкох Б.В.	234
Бобряков А.В.	104	Пирожникова О.И.	80
Богатиков В.Н.	60, 110, 145	Плесневич Г.С.	91
Бутакова М.А.	168	Поляков В.И.	156
Веселов Г.Е.	287, 298	Поляниченко Д.С.	168
Гатчин Ю.А.	156	Польгаев А.С.	234
Гельцер Б.И.	274	Прокопенко С.А.	104
Гладков Л.А.	287	Рублев В.Ю.	274
Городецкий В.И.	200	Санаева Г.Н.	110
Грачев В.В.	258	Симонов Н.А.	117
Девяткин Д.А.	213	Синюк В.Г.	67, 127
Долгий А.И.	53	Смирнова А.В.	247
Егорова Ю.Е.	47	Солдатенко И.С.	47
Жиглатый А.А.	309	Суворова А.В.	247
Иляхинский А.В.	60	Суханов А.В.	137
Ипатов М.А.	224	Ткалич В.Л.	80
Калинкина М.Е.	80	Фам К.Б.	145
Каратач С.А.	67	Федотов М.В.	258
Карпов В.Э.	25	Хаддад Г.	268
Карташов О.О.	168	Хаддад Н.	268
Кемайкин В.К.	234	Чернов А.В.	168
Кобринский Б.А.	37	Чикалов Н.В.	156
Ковалев С.М.	53, 137	Шахгельдян К.И.	274
Колоденкова А.Е.	53	Шарапов А.Л.	258
Коробейников А.Г.	74, 80	Щеглов Б.О.	274
Королева М.Н.	182	Щеглов С.Н.	319
Костерин В.В.	274	Щеглова С.Н.	274
Кулабухов С.В.	127	Язенин А.В.	47
Курило Д.С.	85	Ярушкина Н.Г.	51, 177
Лебедев Б.К.	298	Ясир М.Д.	287
Лебедев О.Б.	298, 309		

СОДЕРЖАНИЕ

Предисловие	6
-------------------	---

ПЛЕНАРНЫЕ ДОКЛАДЫ

<i>Аверкин А.Н.</i> О научном наследии В.Б. Тарасова.....	12
<i>Карпов В.Э.</i> Биоинспирированные модели поведения. Системный кризис или новые горизонты?.....	25
<i>Кобринский Б.А.</i> Интегрированные и гибридные системы искусственного интеллекта: методологические проблемы и вопросы терминологии.....	37
<i>Язенин А.В., Егорова Ю.Е., Солдатенко И.С.</i> От нечеткой до возможно-вероятностной оптимизации.....	47
<i>Ярушклина Н.Г.</i> Интеграция методов конструирования признаков и методов обработки знаний в задачах машинного обучения.....	51

СЕКЦИЯ 1. НЕЧЁТКИЕ МОДЕЛИ, МЯГКИЕ ВЫЧИСЛЕНИЯ, ИЗМЕРЕНИЯ И ОЦЕНКИ

<i>Долгий А.И., Колоденкова А.Е., Ковалев С.М.</i> Интерпретируемая модель классификатора на основе нечеткой логистической регрессии.....	53
<i>Иляхинский А.В., Богатиков В.Н.</i> Вероятностные модели в информационных технологиях.....	60
<i>Каратач С.А., Синюк В.Г.</i> Метод и параллельная реализация нечеткой классификации объектов с несинглтонной фаззификацией.....	67
<i>Коробейников А.Г., Мунтян Е.Р.</i> Сравнение качества данных геомагнитных обсерваторий, входящих в мировую сеть INTERMAGNET, с использованием методов Big Dat.....	74
<i>Коробейников А.Г., Ткалич В.Л., Пирожникова О.И., Калинкина М.Е.</i> Автоматизированный комплекс неразрушающего контроля на базе методов искусственного интеллекта.....	80
<i>Курило Д.С., Мошкин В.С., Мошкина И.А.</i> Гибридный алгоритм поиска аномалий временных рядов с учетом особенностей предметной области.....	85
<i>Плесневич Г.С.</i> Состоятельная и полная система дедукции для пропозициональной нечеткой логики с оценками.....	91
<i>Прокопенко С.А., Бобряков А.В.</i> Применение темпоральных нейро-нечетких сетей Петри в информационно-технологических системах.....	104
<i>Санаева Г.Н., Богатиков В.Н.</i> Исследование задачи управления процессом производства ацетилена на основе нечёткой логики.....	110
<i>Симонов Н.А.</i> Применение модели пятен к решению обратных задач и искусственному интеллекту.....	117

<i>Синюк В.Г., Кулабухов С.В.</i> Методы вывода для систем типа Мамдани и Такаги-Сугено с несинглтонной фаззификацией на основе нечеткого значения истинности.....	127
<i>Суханов А.В., Ковалев С.М., Акперов И.Г., Ольгейзер И.А.</i> Выявление предвестников бифуркаций динамической системы на основе анализа структуры ее нечеткой модели.....	137
<i>Фам К.Б., Богатииков В.Н.</i> Исследование динамических свойств системы управления с нечеткими регуляторами процесса сушки зеленого чая.....	145
<i>Чикалов Н.В., Березин М.Е., Гатчин Ю.А., Поляков В.И.</i> Нечеткий логико-лингвистический алгоритм обнаружения инцидентов в киберфизических системах.....	156

СЕКЦИЯ 2. МАШИННОЕ ОБУЧЕНИЕ, НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

<i>Александров А.А., Чернов А.В., Поляниченко Д.С., Карташов О.О., Бутакова М.А.</i> Предиктивный анализ поверхностной деформации материалов на основе данных акустическая эмиссии.....	168
<i>Андреев И.А., Мошкин В.С., Ярушикина Н.Г.</i> Подход к анализу личностных характеристик пользователей социальных сетей с применением машинного обучения.....	177
<i>Аникин М.А., Королева М.Н.</i> Обработка нормативных правовых актов с применением нейросетевых моделей.....	182
<i>Апарнев А.Н.</i> Нейронная сеть для определения выполнимости формул булевой логики.....	189
<i>Городецкий В.И.</i> Предсказание намерений конкурента в среде с противодействием.....	200
<i>Девяткин Д.А.</i> Построение случайных лесов деревьев решений с применением ядерных разделителей.....	213
<i>Ипатов М.А., Мохов А.С.</i> Сравнение подходов к генерации текстовых описаний.....	224
<i>Палюх Б.В., Кемайкин В.К., Польшаев А.С.</i> Прогнозирование координат динамического объекта в условиях неточности измерений методами нейросетевых технологий.....	234
<i>Суворова А.В., Смирнова А.В.</i> Проектирование инструментария для выбора методов интерпретируемого машинного обучения.....	247
<i>Федотов М.В., Шаранов А.Л., Грачев В.В.</i> Способы повышения качества обучения нейросетевых диагностических моделей сложных технических объектов.....	258

<i>Хаддад Н., Мышенков К.С., Хаддад Г.</i> Сравнительное исследование диагностики болезни Паркинсона с помощью машинного обучения.....	268
<i>Шахгельдян К.И., Рублев В.Ю., Гельцер Б.И., Щеглов Б.О., Костерин В.В., Щеглова С.Н.</i> Методы машинного обучения для прогнозирования риска внутригоспитальной летальности после коронарного шунтирования.....	274

**СЕКЦИЯ 3. БИОИНСПИРИРОВАННЫЕ ПОДХОДЫ,
ЭВОЛЮЦИОННОЕ МОДЕЛИРОВАНИЕ,
ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ**

<i>Веселов Г.Е., Гладков Л.А., Ясир М.Д.</i> Разработка и исследование гибридного алгоритма решения задачи размещения элементов ЭВА.....	287
<i>Веселов Г.Е., Лебедев Б.К., Лебедев О.Б.</i> Биоинспирированный алгоритм планирования траектории в условиях частичной неопределенности и наличии перемещающихся препятствий.....	298
<i>Лебедев О.Б., Жиглатый А.А.</i> Биоинспирированный алгоритм построения парасочетания.....	309
<i>Щеглов С.Н.</i> Решение задач транспортного типа с использованием биоинспирированных методов.....	319
Abstracts	329
Авторский указатель	348

Научное издание

Сборник трудов
XI Международной научно-практической конференции
«Интегрированные модели и мягкие вычисления
в искусственном интеллекте» ИММВ-2022.
Труды конференции в 2-х томах
Том 1

Российская ассоциация искусственного интеллекта

Подписано в печать 11.05.2022 г.
Формат 60x84¹/₁₆. Тираж 200 экз. Усл. печ. л. 20,5.

ISBN 978-5-604-2802-5-6

