



XXII



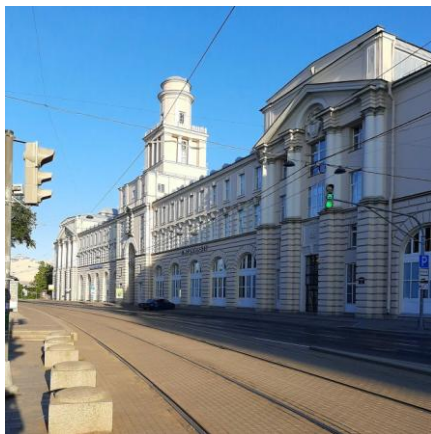
ИТМО

**НАЦИОНАЛЬНАЯ
КОНФЕРЕНЦИЯ
ПО ИСКУССТВЕННОМУ
ИНТЕЛЛЕКТУ
С МЕЖДУНАРОДНЫМ
УЧАСТИЕМ**



**СПб
ФИЦ
РАН**

КИИ-2025



**ТРУДЫ КОНФЕРЕНЦИИ
Том 2**

6-10 октября 2025 г.
Санкт-Петербург

XXII

Российская ассоциация
искусственного интеллекта

Федеральный
исследовательский центр
«Информатика и
управление» РАН

Национальный
исследовательский
университет ИТМО

Санкт-Петербургский
Федеральный
исследовательский
центр РАН

НАЦИОНАЛЬНАЯ КОНФЕРЕНЦИЯ ПО ИСКУССТВЕННОМУ ИНТЕЛЛЕКТУ С МЕЖДУНАРОДНЫМ УЧАСТИЕМ

КИИ-2025

ТРУДЫ КОНФЕРЕНЦИИ
Том 2

6-10 октября 2025 г.
Санкт-Петербург,
Национальный исследовательский
университет ИТМО

Санкт-Петербург
СПб ФИЦ РАН
2025

УДК 004.8+004.89+004.82+004.032.26(045)+004.9

ББК 32.813

Д 22

Организаторы конференции:

Российская ассоциация искусственного интеллекта

Федеральный исследовательский центр «Информатика и управление» РАН

Национальный исследовательский университет ИТМО

Санкт-Петербургский Федеральный исследовательский центр РАН

Д 22 Двадцать вторая Национальная конференция по искусственному интеллекту с международным участием, КИИ-2025 (Санкт-Петербург, 6-10 октября 2025 г.). Труды конференции. В 3-х томах. Т. 2. – СПб: Изд-во СПб ФИЦ РАН, 2025. – 385 с.

ISBN 978-5-6052274-4-1

Двадцать вторая Национальная конференция по искусственному интеллекту с международным участием КИИ-2025 продолжает традицию советских (российских) конференций, организуемых Российской ассоциацией искусственного интеллекта.

Во втором томе трудов публикуются доклады участников конференции, представленные на следующих секциях:

Секция 5 «Нечеткие модели, мягкие измерения и вычисления, биоинспирированные методы»,

Секция 6 «Интеллектуальные агенты, роботы, компьютерное зрение»,

Секция 7 «Интеллектуальное управление и поддержка принятия решений».

ББК 32.813

Рецензенты: академик РАН, ИПУ РАН *С.Н. Васильев*,
д.т.н., ФИЦ ИУ РАН *О.Г. Григорьев*

ISBN 978-5-6052274-4-1

© Авторы, 2025

© Российская ассоциация искусственного интеллекта, 2025

© Издательство СПб ФИЦ РАН, 2025



ВАЛЕРИЙ БОРИСОВИЧ ТАРАСОВ

16 февраля 1955 – 22 июля 2021

В этом году конференция посвящена 70-летию со дня рождения Валерия Борисовича Тарасова – выдающегося ученого в области искусственного интеллекта, ведущего отечественного специалиста по направлениям семиотического моделирования, многоагентных систем, нечетких систем, мягких вычислений и когнитивных измерений.

СОПРЕДСЕДАТЕЛИ КОНФЕРЕНЦИИ

Соколов И.А., акад. РАН, ФИЦ ИУ РАН, Москва

Васильев В.Н., член-корр. РАН, НИУ ИТМО, Санкт-Петербург

Ронжин А.Л., д.т.н., проф. РАН, СПб ФИЦ РАН, Санкт-Петербург

ПРОГРАММНЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

Сопредседатели Программного комитета

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

Котенко И.В., д.т.н., проф., СПб ФИЦ РАН, Санкт-Петербург

Заместители председателя Программного комитета

Грибова В.В., член-корр. РАН, ИАПУ ДВО РАН, Владивосток

Забейжайло М.И., д.ф.-м.н., проф., ФИЦ ИУ РАН, Москва

Ответственный секретарь Программного комитета

Подвесовский А.Г., к.т.н., доц., БГТУ, Брянск

Члены программного комитета

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

Афанасьева Т.В., д.т.н., РЭУ им. Г.В. Плеханова, Москва

Бобцов А.А., д.т.н., проф., НИУ ИТМО, Санкт-Петербург

Болодурина И.П., д.т.н., проф., Оренбургский ГУ, Оренбург

Боргест Н.М., к.т.н., доц., Самарский НИУ им. акад. С.П. Королева, Самара

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

Бухановский А.В., д.т.н., проф., НИУ ИТМО, Санкт-Петербург

Васильев С.Н., акад. РАН, ИПУ РАН, Москва

Визильтер Ю.В., д.ф.-м.н., проф., ГосНИИАС, Москва

Виноградов Д.В., д.ф.-м.н., ФИЦ ИУ РАН, Москва

Гаврилова Т.А., д.т.н., проф., СПбГУ, Санкт-Петербург

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

Городецкий В.И., д.т.н., проф., АО «Эврика», Санкт-Петербург

Еремеев А.П., д.т.н., проф., НИУ «МЭИ», Москва

Желтов С.Ю., акад. РАН, ГосНИИАС, Москва

Загорулько Ю.А., к.т.н., ИСИ СО РАН, Новосибирск

Ильин В.А., д.ф.-м.н., проф., НИУ ИТМО, Санкт-Петербург

Калюжная А.В., к.т.н., доц., НИУ ИТМО, Санкт-Петербург

Карпов А.А., д.т.н., проф., СПб ФИЦ РАН, Санкт-Петербург

Ковалев С.М., д.т.н., проф., РГУПС, Ростов-на-Дону

Колесников А.В., д.т.н., проф., БФУ, Калининград

Колоденкова А.Е., д.т.н., доц., Самарский НИУ, Самара

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

Кузнецов С.О., д.ф.-м.н., проф., НИУ ВШЭ, Москва

Лебедев О.Б., д.т.н., доц., ВАГШ ВС РФ, Москва

Лукашевич Н.В., д.т.н., проф., МГУ им. М.В. Ломоносова
Макаров Д.А., к.т.н., ФИЦ ИУ РАН, Москва
Мисник А.Е., к.т.н., доц. БРУ, Республика Беларусь, Могилев
Михеенкова М.А., д.т.н., проф., ФИЦ ИУ РАН, Москва
Мошкин В.С., к.т.н., доц., УлГТУ, Ульяновск
Насонов Д.А., к.т.н., доц., проф., НИУ ИТМО, Санкт-Петербург
Пальчунов Д.Е., д.ф.-м.н., доц., ИМ СО РАН, Новосибирск
Палюх Б.В., д.т.н., проф., ТвГТУ, Тверь
Панов А.И., д.ф.-м.н., доц., МФТИ, Москва
Редько В.Г., д.ф.-м.н., проф., НИИСИ РАН, Москва
Ройзензон Г.В., к.т.н., доц., ФИЦ ИУ РАН, Москва
Рыбина Г.В., д.т.н., проф., НИЯУ МИФИ, Москва
Смирнов И.В., д.ф.-м.н., доц., ФИЦ ИУ РАН, Москва
Стефанюк В.Л., д.т.н., проф., ИППИ РАН, Москва
Сулейманов Д.Ш., акад. АН РТ, ИПС АН РТ, Казань
Тельнов Ю.Ф., д.э.н., проф., РЭУ, Москва
Уткин Л.В., д.т.н., проф., СПбПУ, Санкт-Петербург
Финн В.К., д.т.н., проф., ФИЦ ИУ РАН, Москва
Хачумов В.М., д.т.н., проф., ФИЦ ИУ РАН, Москва
Шалфеева Е.А., д.т.н., ИАПУ ДВО РАН, Владивосток
Яковлев К.С., к.ф.-м.н., ФИЦ ИУ РАН, Москва

ОРГАНИЗАЦИОННЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

Сопредседатели организационного комитета

Борисов В.В., д.т.н., проф., филиал НИУ МЭИ, Смоленск
Заколдаев Д.А., к.т.н., доц., НИУ ИТМО, Санкт-Петербург

Члены организационного комитета

Благосклонов Н.А., ФИЦ ИУ РАН, Москва
Волошина Н.В., к.т.н., доц., НИУ ИТМО, Санкт-Петербург
Воробьева А.А., к.т.н., доц., НИУ ИТМО, Санкт-Петербург
Давыдов В.В., к.т.н., НИУ ИТМО, Санкт-Петербург
Десницкий В.А., к.т.н., доц., СПб ФИЦ РАН, Санкт-Петербург
Левигун Д.С., к.т.н., СПб ФИЦ РАН, Санкт-Петербург
Кириллова Е.А., д.э.н., доц., филиал НИУ «МЭИ», Смоленск
Попов И.Ю., к.т.н., доц., НИУ ИТМО, Санкт-Петербург
Синяевский Ю.В., к.т.н., филиал НИУ «МЭИ», Смоленск
Солопов Р.В., к.т.н., доц., филиал НИУ «МЭИ», Смоленск
Чечулин А.А., к.т.н., доц., СПб ФИЦ РАН, Санкт-Петербург

ИНФОРМАЦИОННАЯ ПОДДЕРЖКА КОНФЕРЕНЦИИ

ООО «Лаборатория информационных технологий», Смоленск

ПРЕДИСЛОВИЕ

Двадцать вторая Национальная конференция по искусственному интеллекту с международным участием КИИ-2025 продолжает традицию советских (российских) конференций, организуемых Российской ассоциацией искусственного интеллекта (РАИИ).

Конференция посвящена памяти Валерия Борисовича Тарасова, 70-летие со дня рождения которого отмечается в этом году – выдающегося ученого в области искусственного интеллекта, который внес существенный вклад в развитие методов семиотического моделирования, в теорию агентов и многоагентных систем, теорию нечетких систем, мягких вычислений и когнитивных измерений. Валерий Борисович Тарасов был участником Учредительного съезда Советской ассоциации искусственного интеллекта (ныне РАИИ). С 1992 по 2000 год являлся вице-президентом РАИИ, с 2000 года – членом Научного совета РАИИ. Он был одним из основателей Российской ассоциации нечетких систем (ныне Ассоциации нечетких систем и мягких вычислений).

Федеральный проект «Искусственный интеллект» направлен на решение междисциплинарных проблем, сформулированных в Национальной стратегии развития искусственного интеллекта, включая исследования как в области фундаментальной и прикладной науки, так и в образовательной сфере. Эти проблемы постоянно находятся в фокусе внимания членов РАИИ. Их отражение можно видеть в докладах конференции, организаторами которой являются Российская ассоциация искусственного интеллекта, Федеральный исследовательский центр «Информатика и управление» РАН, Национальный исследовательский университет ИТМО, Санкт-Петербургский Федеральный исследовательский центр РАН.

Тематика конференции охватывает следующие основные направления искусственного интеллекта: инженерия знаний; интеллектуальный анализ данных; моделирование рассуждений; интеллектуальный анализ текстов, большие языковые модели; нечеткие модели, мягкие измерения и вычисления, биоинспирированные методы; интеллектуальные агенты, роботы, компьютерное зрение; интеллектуальное управление и поддержка принятия решений; машинное обучение, нейросетевые методы, нейроинформатика; инструментальные средства и технологии проектирования интеллектуальных систем; прикладные интеллектуальные системы.

Представлены пленарные доклады видных ученых и специалистов в области искусственного интеллекта. Из поданных на конференцию 138 секционных докладов Программным комитетом после рецензирования отобраны 102 доклада.

В конференции принимают участие более 280 ученых, исследователей и специалистов из академических институтов, университетов, отраслевых институтов и организаций Москвы, Санкт-Петербурга, Белгорода, Брянска, Владивостока, Донецка, Иркутска, Казани, Калининграда, Красноярского края, Махачкалы, Московской области, Новосибирска, Пермского края, Ростова-на-Дону, Самары, Смоленска, Таганрога, Твери, Ульяновска, Уфы, Ханты-Мансийска, Челябинска, Ярославля, Могилёва (Республика Беларусь), Суйчжоу, (Китай), Ташкента (Узбекистан), Хайдарабада (Индия).

Программный комитет КИИ-2025

УДК 519.612

doi: 10.15622/rcai.2025.032

**АГЕНТНО-ОРИЕНТИРОВАННЫЙ АЛГОРИТМ
РЕШЕНИЯ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ
УРАВНЕНИЙ БОЛЬШОЙ РАЗМЕРНОСТИ**

Д.А. Берёза (*dbereza@sfedu.ru*)

Л.А. Гладков (*lagladkov@sfedu.ru*)

Н.В. Гладкова (*nvgladkova@sfedu.ru*)

Южный федеральный университет, Таганрог

В работе решается задача построения эффективного метода решения систем линейных алгебраических уравнений (СЛАУ) большой размерности. Отмечена высокая актуальность и практическая значимость решения поставленной задачи. Выполнен анализ различных подходов к построению математических моделей объектов проектирования. Приведена математическая постановка задачи. Предложен механизм снижения размерности решаемой системы уравнений на основе интеграции эволюционных алгоритмов и методов оптимизации на основе теории Парето-оптимальности. Разработан гибридный алгоритм решения СЛАУ на основе генетического поиска и многоагентной организации. Разработана архитектура программной подсистемы для решения СЛАУ на основе использования сервисно-ориентированного подхода. Выполнена программная реализация предложенного гибридного алгоритма и про-

ведены серии вычислительных экспериментов. Полученные результаты в целом подтвердили эффективность предлагаемого подхода и позволили наметить перспективные направления дальнейшего развития исследований.

Ключевые слова: системы автоматизированного проектирования, сверхбольшие интегральные схемы, многоагентные системы, сервис-ориентированные архитектуры, эволюционные вычисления.

Введение

Математическую модель объекта проектирования можно определить как множество различного рода математических элементов (переменные, упорядоченные и неупорядоченные массивы чисел и т.д.), отражающее моделируемые свойства этих элементов и взаимосвязи между ними.

Процесс построения математической модели проектируемого объекта состоит в объединении математических описаний отдельных элементов в единое целое на основе универсальных физических законов и уравнений математической физики.

В процессе автоматизированного проектирования сложного объекта, например, интегральной схемы, математическая модель формируется на основе набора уравнений множества элементарных элементов, большинство из которых являются стандартными и, следовательно, могут быть взяты из специализированных библиотек.

Сегодня на рынке представлено множество программных комплексов для схемотехнического моделирования, которые имеют один общий и весьма существенный недостаток: качество и скорость работы программной системы существенно зависит от размерности решаемой задачи. Для решения систем дифференциальных уравнений в этих программах применяются численные методы Рунге – Кутты или Гира, для систем нелинейных уравнений – метод Ньютона, а для решения систем линейных алгебраических уравнений (СЛАУ) – метод Гаусса или LU-разложение [Гридин, 2008], [Баландин и др., 2000], [Рено, 2007].

При решении реальных задач проектирования, например, проектирования сверх- и ультрабольших интегральных схем, необходимы разработка и использование новых эффективных алгоритмических и программных инструментов на всех этапах проектирования [Shervani, 1995], [Charles etc, 2009].

Актуальность данной задачи объясняется тем, что классические численные методы, которые дают удовлетворительные результаты при решении задач малой и средней размерности, не могут эффективно использоваться при проектировании СБИС в силу высокой вычислительной сложности и размерности данных задач. Это привело к появлению новых подходов к решению задач проектирования с использованием распределенных вычислений, различных метаэвристических, в том числе эволюцион-

ных и биоинспирированных, алгоритмов поиска и оптимизации [Гладков и др., 2024], [Гладков и др., 2025], [Карпенко, 2014], [Гладков и др., 2009], [Cohoon etc, 2003].

Для снижения вычислительной сложности и сокращения времени решения задач проектирования используют также алгоритмы, позволяющие распараллеливать процесс поиска, в том числе параллельные генетические алгоритмы. Также перспективным направлением является организация процесса вычислений на основе использования агентных организаций и архитектур взаимодействия [Тарасов и др., 2006], [Майоров и др., 2015], [Скобелев, 2013], [Гридин и др., 2014].

1. Постановка задачи

С точки зрения схемотехнического проектирования важнейшей задачей является создание математической модели, которая бы адекватно отображала соответствующие характеристики объекта проектирования в виде набора математических уравнений, и позволяла использовать существующие программные пакеты для решения задачи моделирования.

Любой технический объект может быть однозначно описан набором внутренних, внешних и выходных параметров. Для построения математической модели объекта, имеющего n различных характеристик используют обычно вектор внутренних параметров, каждый из которых отражают какую-либо зависимость, функцию и т.д. [Бойко и др., 2006], [Баландин, 2007].

Эффективность применения численных методов для решения задач анализа и синтеза оптимальных электрических параметров проектируемых полупроводниковых приборов в значительной мере определяется выбором начальной области, которая называется часто областью сходимости. В случае, когда местоположение начального решения выбрано неправильно и находится вне области сходимости значительно повышает риск того, что оптимальное решение не будет найдено [Влах и др., 1998].

Рост сложности проектирования, совершенствование технологий изготовления, плотности и глубины интеграции элементов привели к значительному увеличению размерности систем обыкновенных дифференциальных (ОДУ) и линейных (ЛАУ) уравнений, составляющих основу математических моделей элементной базы в микроэлектронике. Классические численные методы решения систем уравнений при проектировании не всегда обеспечивают необходимое качество и точность, кроме того, традиционные численные методы (например, градиентные методы) обладают еще одним существенным недостатком – при моделировании многоэкстремальных функций они не могут преодолевать впадины образующиеся в ландшафте функции и не способны находить выход из локальных экстремумов. Одним из возможных подходов, позволяющих снизить сложность моделирования и размерность решаемых задач является допущение различных упрощений.

[illegible]

В результате был предложен модифицированный эволюционный алгоритм решения СЛАУ. С этой точки зрения задача заключается в нахождении решения системы линейных алгебраических уравнений:

где $A = \{a_{ij}\}$ – прямоугольная матрица коэффициентов, а x и b – искомый и заданный векторы координат $x_i, b_i \in R^n, i, j = 1, \dots, n$.

Задача состоит в нахождении оптимального значения следующей целевой функции:

$$f = \|Ax - b\| + 0.1\|x\| + F_p \rightarrow \min ,$$

где F_p – это показатель эффективности, вычисляемый по методу Парето. Схема предлагаемого эволюционного алгоритма на основе принципа Парето-оптимальности представлена на рис. 1.

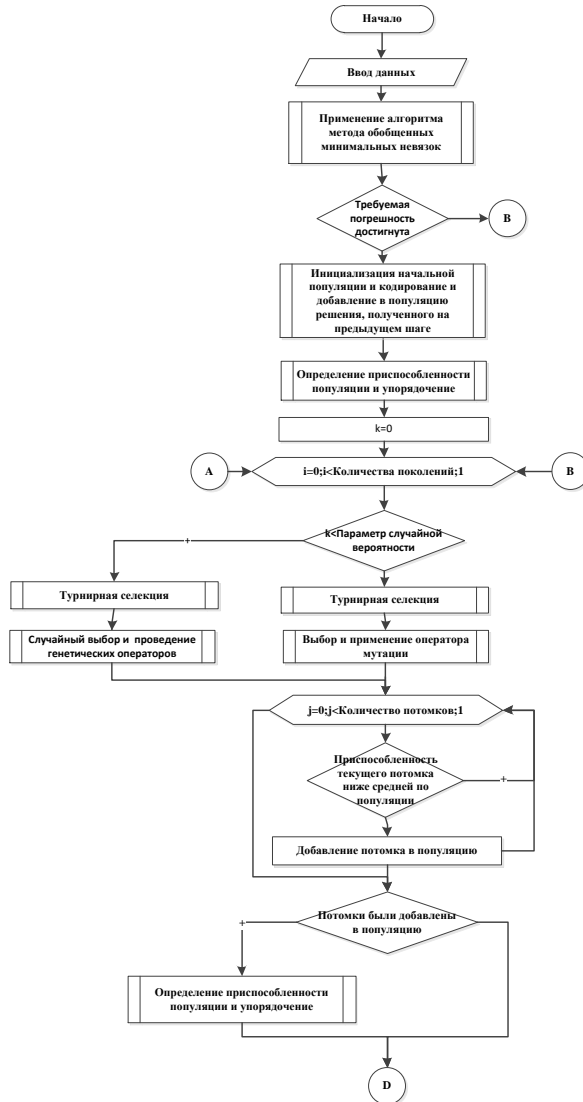


Рис. 1. Схема эволюционного алгоритма на основе принципа Парето

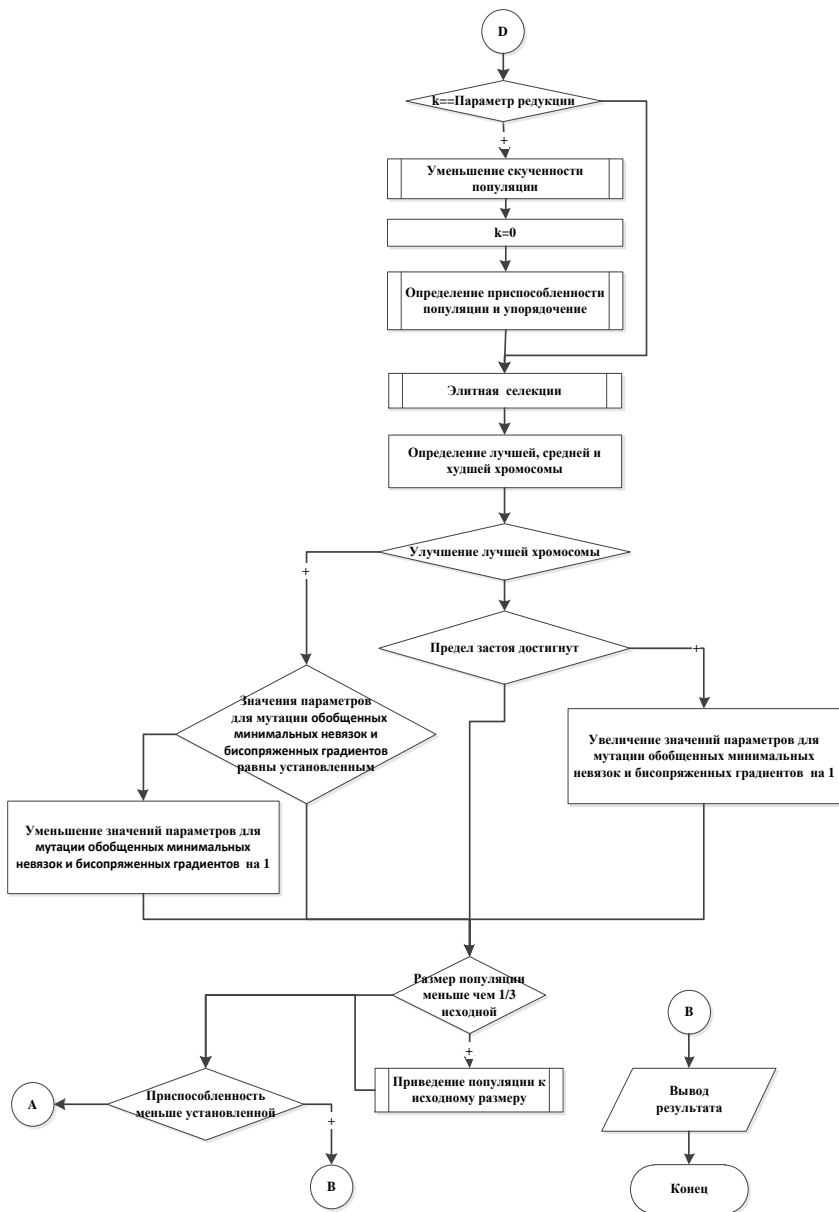


Рис. 1. Схема эволюционного алгоритма на основе принципа Парето (продолжение)

Структура предложенного алгоритма, отражающая последовательность действий, выполняемых в процессе вычислительного агента, приведена на рис. 2.

1. Φ_0 – вектор начального приближения решения;
2. $r_0 = b - A\Phi_0$;
3. \bar{r}_0 – произвольный вектор, такой, что $(\bar{r}_0, r_0) \neq 0$;
4. $\rho_0 = \alpha = \omega_0 = 1$;
5. $v_0 = p_0 = 0$;
6. для $k = 1, 2, 3, \dots$
7. $\rho_k = (\bar{r}_0, r_{k-1})$; $\beta = \left(\frac{\rho_k}{\rho_{k-1}}\right)\left(\frac{\alpha}{\omega_{k-1}}\right)$;
8. $p_k = r_{k-1} + \beta(p_{k-1} - \omega_{k-1}v_{k-1})$;
9. определение вектора u из решения системы $Mu = p_k$;
10. $v_k = Au$;
11. $\alpha = \frac{\rho_k}{(\bar{r}_0, v)}$;
12. $s = r_{k-1} - \alpha v_k$;
13. определение вектора z из решения системы $Mz = s$;
14. $t = Az$;
15. $w_k = \begin{pmatrix} t, s \end{pmatrix}$;
16. $\Phi_k = \Phi_{k-1} + \alpha u + \omega_k z$;
17. если Φ_k достигло требуемой точности – выход из цикла;
18. $r_k = s - \omega_k t$;
19. конец цикла по k

Рис. 2. Структура алгоритма работы вычислительного агента

2. Разработка масштабируемой мультиагентной среды

В процессе разработки архитектуры программной подсистемы было предложено использовать сервисно-ориентированный подход, в основе которого лежит идея построения подсистемы на основе взаимодействия множества различных служб (сервисов), функционирующих независимо друг от друга, каждая из которых является ответственной за реализацию одной функции. Также удобно использовать в данном случае модульный подход, суть которого состоит в обеспечении взаимозаменяемости и возможности гибко изменять функционал системы без «коренной» перестройки ее схемы. Данное свойство позволяет нам провести ассоциацию между объектами «модуль» и «агент» [Митра и др., 2003], [Baqais, 2017].

В классическом понимании термином «агент» обозначается все, что действует. Согласно принятым искусственному интеллекту и теории многоагентных систем подходе программный агент отличается от обычной компьютерной программы наличием способности к самостоятельному (автономному) функционированию в течении продолжительного периода времени, возможностью осуществлять взаимодействие с другими агентами и внешней средой, адаптироваться к изменениям и наличием цели, на достижение которой направлена его деятельность [Тарасов, 2002], [Рассел и др., 2022].

Архитектура предложенной масштабируемой многоагентной системы показана на рис. 3.

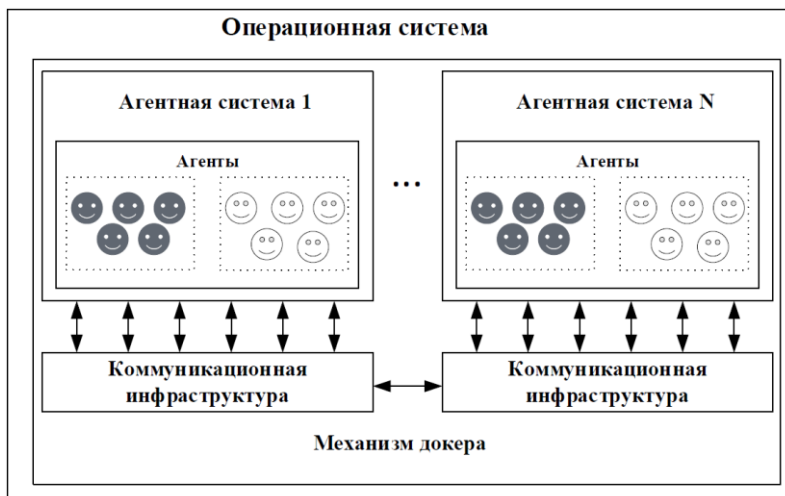


Рис. 3. Архитектура масштабируемой многоагентной системы

Контейнеризация представляет собой методологию создания программного продукта, где приложение или сервис вместе с их зависимостями и настройками (файлы, описывающие манифест развертывания) собираются в единое целое – контейнер. Такой контейнерный образ может быть протестирован как цельная единица и затем запускаться на узле операционной системы как копия созданного образа [Митра и др., 2003].

В процессе создания контейнеров и их последующей загрузки в регистры (как локальные, так и удаленные) удобно использовать инструмент Docker. Этот инструмент способствует формированию образа контейнера, включающего в себя все необходимые файлы операционной системы и приложения, а также обеспечивает его упаковку для переноса между различными платформами. Внутри контейнера обычно размещается какой-либо дистрибутив Linux, реже – упрощенная версия Windows Server.

Для инициализации процесса развертывания применяются файлы без указания расширения, имя которых совпадает с "Dockerfile", если таких файлов в каталоге только один, или же файлы с любым именем и расширением "dockerfile". Начальная строка в этих файлах традиционно содержит информацию о базовом образе (основной образ операционной системы или фреймворка), после чего следует процесс установки и конфигурирования компонентов приложения.

Docker предоставляет возможность разделения процессов сборки и доставки приложения в различных контейнерах в ходе создания образа, что отражено в одном манифесте. В таком контейнере для сборки нет необходимости в указании команды запуска. Этот подход применяется при

развертывании микросервисов на Node.js, поскольку приложения, написанные на React, требуют преобразования в оптимизированный код перед запуском. В качестве базового образа используется образ Node.js, который базируется на операционной системе Alpine Linux, специально адаптированной для эффективной работы в контейнерной среде.

Применение инструмента Docker Compose [Митра и др., 2003] облегчает процесс взаимодействия между контейнерами, их сборку и последующее их размещение. В контексте данной работы контейнеры именуются сервисами, и их параметры (включая настройки окружения, ограничения на использование ресурсов и прочее) определяются через файл `docker-compose.yml`, который является единственным в рабочей директории.

На рис. 4. представлен обобщенный агентно-ориентированный алгоритм решения больших систем линейных уравнений (СЛАУ).

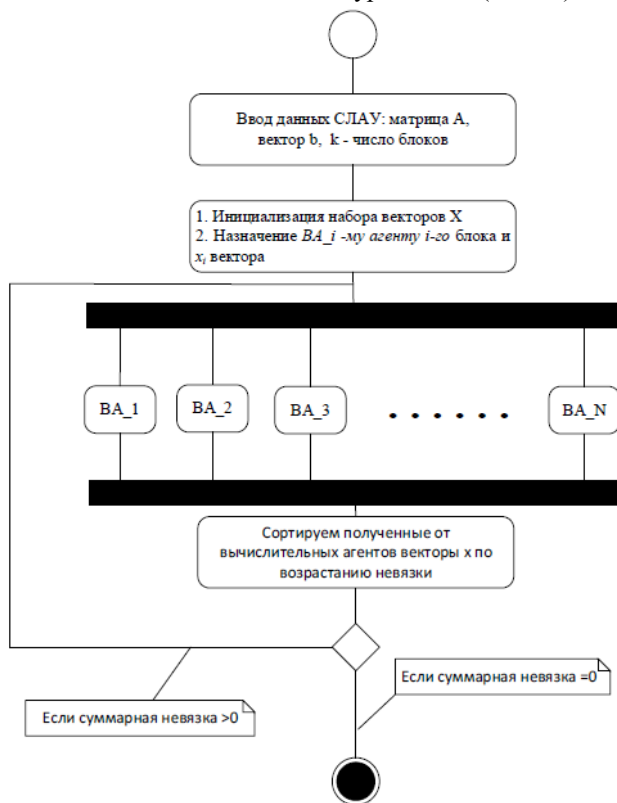


Рис. 4. Обобщенная схема алгоритма решения СЛАУ большого размера

3. Программная реализация и вычислительные эксперименты

Предложенный гибридный алгоритм был реализован в виде программной подсистемы. В процессе моделирования была поставлена задача изучить функционирование данной программной подсистемы, определить взаимозависимости между точностью и временем решения систем линейных уравнений (СЛАУ) и их размерностью, а также проверить эффективность предложенного агентно-ориентированного метода решения СЛАУ и метода решения уравнений, описывающих поведение нанометровых СБИС [Береза и др., 2025].

Критерием для оценки эффективности алгоритма является минимальное значение целевой функции, которое достигается при заданном количестве итераций и отражает точность полученного решения СЛАУ.

В ходе исследования применялась платформа для многоагентной симуляции. Эксперименты были организованы на базе тестовых систем линейных алгебраических уравнений, разработанных на этапе схемотехнического проектирования.

Для проверки эффективности разработанного алгоритма проводились серии вычислительных экспериментов для решения систем линейных алгебраических уравнений разной размерности. Для сокращения времени и объема вычислений было принято решение выбрать следующие значения параметров тестирования: доверительная вероятность – 90%; допустимая ошибка – 5%. Это позволило сократить размер экспериментальной выборки до 50. Результаты вычислительных экспериментов показаны в табл. 1.

Таблица 1
Результаты вычислительных экспериментов

№ п/п серии	Размер серии	Размер СЛАУ	Время решения (сек.)
1	50	500*500	2302
2	50	600*600	4264
3	50	700*700	4553
4	50	800*800	8123
5	50	900*900	12777

В результате выполнения статистической обработки полученных данных была построена теоретическая зависимость времени работы подсистемы от размерности решаемой задачи:

$$y = 0,0313x^2 - 16,693x + 2744,9.$$

Сравнение характера данной теоретической зависимости с результатами, полученными в ходе проведения вычислительных экспериментов, подтвердили первоначальные предположения. Характер построенных зависимостей (рис. 5) позволяет сделать однозначный вывод о том, что временная сложность разработанного гибридного алгоритма является полиномиальной.

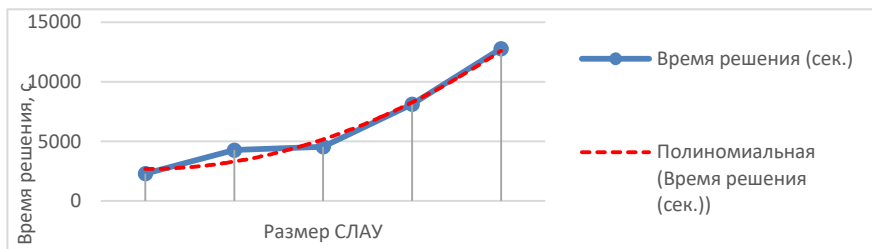


Рис. 5. Зависимость времени решения от размера СЛАУ

Заключение

На основании полученных экспериментальных данных можно сделать вывод о том, что разработанный гибридный алгоритм может эффективно использоваться для решения систем линейных алгебраических уравнений, имеет квадратичную временную сложность и демонстрирует стабильность функционирования вне зависимости от роста размерности задачи. В дальнейшем планируется продолжить исследования в направлении повышения эффективности предлагаемого подхода за счет использования биоинспирированных методов поиска.

Список литературы

- [Баландин и др., 2000] Баландин М.Ю., Шурина Э.П. Методы решения СЛАУ большой размерности. – Новосибирск: Изд-во НГТУ. 2000.
- [Береза и др., 2025] Береза Д.А., Гладков Л.А., Гладкова Н.В. Разработка агентно-ориентированного алгоритма решения систем линейных алгебраических уравнений большой размерности. // Известия ЮФУ. Технические науки. – 2025. – № 1. – С. 22-37.
- [Бойко и др., 2006] Бойко А.Я., Безруков А.Е., Русаков А.С., Ткачев Д.Ф., Хапаев М.М. Новый алгоритм вычисления двумерных емкостей в задаче экстракции емкости. // II Всероссийская научно-техническая конференция «Проблемы разработки перспективных микроэлектронных систем»: Сб. научных тр. под общ. ред. А.Л. Стемпковского. – М.: ИППМ РАН, 2006.
- [Влах и др., 1998] Влах И., Сингхал К. Машинные методы анализа и проектирования электронных схем: пер. с англ. – М.: Радио и связь, 1988.

- [Гладков, 2000] Гладков Л.А. О некоторых подходах к построению гибридных интеллектуальных систем для решения графовых задач // Новости искусственного интеллекта. – 2000. – № 3. – С. 71-90.
- [Гладков и др., 2009] Гладков Л.А., Курейчик В.В., Курейчик В.М., Сороколетов П.В. Биоинспирированные методы в оптимизации. – М.: Физматлит, 2009.
- [Гладков и др., 2011] Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы. – М.: Физматлит, 2011.
- [Гладков и др., 2024] Гладков Л.А., Кравченко Ю.А., Курейчик В.В., Родзин С.И. Интеллектуальные системы: модели и методы метаэвристической оптимизации. Монография. – Чебоксары: Среда, 2024.
- [Гридин, 2008] Гридин В.Н. Численно-аналитическое моделирование радиоэлектронных схем. – М.: Наука, 2008.
- [Гридин и др., 2014] Гридин В.Н., Дмитриевич Г.Д., Анисимов Д.А. Архитектура распределенных сервис-ориентированных систем автоматизированного проектирования // Известия ЮФУ. Технические науки. – 2014. – № 7(156). – С. 51-58.
- [Карпенко, 2014] Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой. – М.: МГТУ им. Баумана, 2014.
- [Майоров и др., 2015] Майоров И.В., Скобелев П.О. Модель динамики агентов потребностей и возможностей // Труды XVII Международной конференции «Проблемы управления и моделирования в сложных системах», Самара, 2015. – Самара: СНЦ РАН, 2015. – С. 79-87.
- [Митра и др., 2003] Митра Р., Надареишвили И. Микросервисы. От архитектуры до релиза. – СПб.: Питер, 2023.
- [Потапов, 2007] Потапов Ю. Технология экстракции паразитных параметров для моделирования межсоединений // Технологии в электронной промышленности. – 2007. – № 6. – С. 22-26.
- [Рассел и др., 2022] Рассел С., Норвиг П. Искусственный интеллект: современный подход. – М.: Издательский дом «Вильямс», 2022.
- [Рено, 2007] Рено Н.Н. Численные методы. – М.: КДУ, 2007.
- [Скобелев, 2013] Скобелев П.О. Ситуационное управление и мультиагентные технологии: коллективный поиск согласованных решений в диалоге // Онтологии проектирования. – 2013. – № 2(8). – С. 26-48.
- [Тарасов, 2002] Тарасов В.Б. От многоагентных систем к интеллектуальным организациям. – М.: Эдиториал УРСС, 2002.
- [Тарасов и др., 2006] Тарасов В.Б., Голубин А.В. Эволюционное проектирование: на границе между проектированием и самоорганизацией // Известия ТРТУ. Технические науки. – 2006. – № 8(63). – С. 77-82.
- [Baqais, 2017] Baqais A.A.B. A Multi-view Comparison of Various Metaheuristic and Soft Computing Algorithms // I.J. Mathematical Sciences and Computing. – 2017. – 1.3 (4). – P. 8-19.
- [Charles etc, 2009] Alpert, Charles J., Mehta, Dinesh P., Sapatnekar, Sachin S. Handbook of algorithms for physical design automation. – CRC Press, New York, USA, 2009.
- [Cohon etc, 2003] 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.
- [Shervani, 1995] Shervani N. Algorithms for VLSI physical design automation. – USA, Kluwer Academy Publisher, 1995.

УДК 004.89

doi: 10.15622/rcai.2025.033

МЯГКИЕ СИТУАЦИОННО-КОГНИТИВНЫЕ МОДЕЛИ ДЛЯ ИНТЕЛЛЕКТУАЛЬНОЙ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ¹

В.В. Борисов (*vbor67@mail.ru*)
А.С. Федулов (*fedulov_a@mail.ru*)
С.А. Федулова (*svfed67@mail.ru*)

Филиал Национального исследовательского университета «МЭИ»
в г. Смоленске, Смоленск

Предлагаются мягкие ситуационно-когнитивные модели (МСКМ), основанные на полиморфной гибридизации нечетких ситуационных и когнитивных моделей, обеспечивающей: их структурно-параметрическое соответствие; взаимообусловленность процедур построения и настройки в рамках единой полиморфной модели; их совместное взаимозависимое использование для интеллектуальной поддержки принятия ситуативных решений. Рассмотрен разработанный метод построения и настройки МСКМ, в рамках которого для оценки воздействия ситуативных решений на зависимые признаки и учета распределенного во времени их воздействия предлагается использовать результаты сценарного моделирования в когнитивной сетевой структуре МСКМ, являющейся двойственной по отношению к ситуационной сетевой структуре МСКМ. Рассмотрен пример использования МСКМ для интеллектуальной поддержки принятия решений.

Ключевые слова: мягкие ситуационно-когнитивные модели, полиморфная гибридизация, поддержка принятия решений.

Введение

В настоящее время активно развиваются методы и средства интеллектуальной поддержки принятия решений для обеспечения ситуационной осведомленности и выработки решений в зависимости от различных поведенческих стратегий и сценариев достижения целевых ситуаций [Kim et al., 2021], [Chon et al., 2023].

¹ Работа выполнена в рамках государственного задания (проект № FSWF-2023-0012).

Вопросы ситуационного представления и принятия решений отражены в работах [Поспелов, 2021], [McCarthy, 2002], [Lin, 2007]. Нечеткие ситуационные модели (НСМ) рассмотрены в статьях [Мелихов и др., 1990], [Федунов, 2002], [Борисов и др., 2009], [Борисов и др., 2021]. Ограничениями этих моделей являются: сложность задания нечетких отношений взаимовлияния ситуационных признаков и их нестационарность; проблема учета влияния ситуативных решений на взаимозависимые ситуационные признаки; сложность оценки одновременного воздействия нескольких ситуационных решений на взаимозависимые ситуационные признаки; недостаточный учет фактора времени и продолжительности воздействия ситуативных решений.

Другим классом моделей для обеспечения ситуационной осведомленности в условиях неопределенности являются нечеткие когнитивные модели (НКМ) [Kosko, 1986], [Силов, 1995], [Carvalho et al., 2000], [Borisov et al., 2004], [Федулов, 2005], [Thulukkanam et al., 2015], [Захарова и др., 2020]. Ограничениями этих моделей для поддержки принятия решений является необходимость включения в их состав признаков для непосредственной оценки проблемных ситуаций.

Ранее проводились отдельные исследования, направленные на гибридизацию НСМ и НКМ для интеллектуальной поддержки принятия решений [Борисов и др., 2016], особенностью которых являлось применение НКМ только на этапе построения НСМ. При этом отсутствуют подходы к учету нечетких отношений взаимовлияния между ситуационными признаками при использовании НКМ для построения и настройки НСМ, а также для оценки влияния управляющих воздействий, реализуемых в НКМ, на взаимозависимые ситуационные признаки НСМ.

В работе [Борисов и др., 2024] предложен метод полиморфной гибридизации НСМ и НКМ, включающий в себя: определение свойств этих моделей и разделение их описаний на подгруппы (не трансформируемых, уточняемых и трансформируемых в результате гибридизации моделей); определение типов полиморфных отношений между описаниями свойств НСМ и НКМ; построение матрицы полиморфных отношений между описаниями свойств НСМ и НКМ; задание первоначальных описаний свойств для НСМ и для НКМ; замыкание полиморфных отношений между описаниями свойств НСМ и НКМ; формирование структуры и обоснование финальной совокупности описаний МСКМ с учетом результатов замыкания полиморфных отношений. Метод обеспечивает: структурно-параметрическое соответствие между НСМ и НКМ; взаимообусловленность процедур построения структур и структурно-параметрической настройки этих моделей в рамках единой полиморфной модели; совместное использование НСМ и НКМ для интеллектуальной поддержки принятия решений.

На основании метода полиморфной гибридизации в статье предлагается новая разновидность мягких ситуационно-когнитивных моделей для интеллектуальной поддержки принятия решений.

1. Метод построения и настройки мягких ситуационно-когнитивных моделей

Метод построения и настройки МСКМ состоит из следующих этапов.

Этап 1. Задание множества ситуационных признаков ситуационной сетевой структуры МСКМ и терм-множеств их нечетких значений:

$$P = \{P_i \mid i \in 1, \dots, I\}, \quad P_i = \langle T_i, D_i \rangle, \quad \forall i \in 1, \dots, I,$$

где $T_i = \{T_1^{(i)}, T_2^{(i)}, \dots, T_{J_i}^{(i)}\}$ – терм-множество ситуационного признака P_i ; J_i – число термов ситуационного признака P_i ; D_i – базовое множество нечетких значений ситуационного признака P_i .

Этап 2. Формирование множества эталонных нечетких ситуаций ситуационной сетевой структуры МСКМ, соответствующих множеству сочетаний термов ситуационных признаков.

Множество эталонных нечетких ситуаций $Se = \{Se_q \mid q \in 1, \dots, Q\}$ формируется относительно термов нечетких ситуационных признаков и не содержит нечетко равные ситуации при заданном пороге их нечеткого сходства. Сама Se_q представляется в виде:

$$\forall q \in 1, \dots, Q \quad \tilde{Se}_q = \left\{ \langle \mu_q(\tilde{P}_i) / P_i \rangle \mid i \in 1, \dots, I, \tilde{P}_i = \left\{ \mu_i(T_j^{(P_i)'}) / T_j^{(P_i)} \mid j \in 1, \dots, J_i \right\} \right\},$$

при условии: $\exists! \mu_i(T_j^{(P_i)'}) = 1, \quad \forall l \neq i \quad \mu_l(T_j^{(P_i)'}) = 0$.

Этап 3. Формирование эталонной ситуационной сетевой структуры МСКМ, представляющей собой нечеткий ориентированный взвешенный граф, вершины которого соответствуют эталонным нечетким ситуациям, а дуги – ситуативным решениям (СР), взвешенным степенями предпочтения для переходов по этим ситуациям.

Множество СР $G = \{G_{ik} \mid i \in 1, \dots, I, k \in 1, \dots, K_i\}$ включает в себя СР, направленные на изменения значений отдельных ситуационных признаков и приводящие к переходам из одной эталонной нечеткой ситуации в другую. Каждое $G_{P_i,k}$, влияющее на признак P_i , представляется в виде:

$$G_{P_i,k} = \langle Tg_{P_i,k}, Eg_{P_i,k}, Dg_{P_i,k} \rangle,$$

где $Tg_{P_i,k} = \{\langle \text{«Увеличить»}, \text{«Уменьшить»}, \text{«Не изменять»} \rangle\}$ – терм-множество направления СР $G_{P_i,k}$; $Eg_{P_i,k} = \{\langle \text{«Слабо»}, \text{«Средне»}, \text{«Сильно»} \rangle\}$ – терм-множество силы СР $G_{P_i,k}$; $Dg_{P_i,k}$ – шкала силы СР $G_{P_i,k}$.

Каждое СР $G_{P_i,k}$, применяемое к Se_q , сводится к воздействию на признак P_i и представляется нечетким отношением воздействия $\tilde{G}_{P_i,k}$.

Воздействие СР $G_{P_i,k}$ на P_i реализуется нечеткой max-min-композицией между нечетким множеством \tilde{P}_i , представляющим значение изменяемого признака P_i , и нечетким отношением $\tilde{G}_{P_i,k}$, представляющим СР $G_{P_i,k}$

$$\tilde{P}_i' = \tilde{P}_i \circ \tilde{G}_{P_i,k}.$$

В случае взаимозависимости ситуационных признаков воздействие СР на один из них приводит к изменению зависимого признака. Для оценки такого воздействия СР предлагается использовать результаты когнитивного моделирования распространения воздействия СР по когнитивной сетевой структуре МСКМ, являющейся двойственной к ситуационной сетевой структуре МСКМ.

Этап 4. Задание множества концептов когнитивной сетевой структуры МСКМ и терм-множеств их нечетких значений, которое взаимно однозначно соответствует множеству ситуационных признаков ее ситуационной сетевой структуры:

$$C = \{C_i \mid i \in 1, \dots, I\}, \quad C_i = \langle T_i, D_i \rangle, \quad \forall i \in 1, \dots, I,$$

где C – множество концептов C_i когнитивной сетевой структуры МСКМ; $T_i = \{T_1^{(i)}, T_2^{(i)}, \dots, T_{J_i}^{(i)}\}$ – терм-множество концепта C_i , соответствующее терм-множеству ситуационного признака P_i ; J_i – число термов концепта C_i , соответствующее числу термов ситуационного признака P_i ; D_i – базовое множество нечетких значений концепта C_i , соответствующее базовому множеству нечетких значений ситуационного признака P_i .

Этап 5. Формирование когнитивной сетевой структуры МСКМ, в качестве которой выберем нечеткую реляционную темпоральную когнитивную модель [Borisov et al., 2024], которая представляется в виде:

$$FRTCM = \langle C, R \rangle, \quad C = \{C_i \mid i = 1, \dots, I\}, \quad R = \{R_l \mid l = 1, \dots, I\},$$

$$R_l = \{\tilde{r}_{ij}(t-l) \mid l = 0, \dots, L_j^i, j = 1, \dots, J^i\},$$

$$C_i : \tilde{C}_i(t) = \tilde{F}_i \left(\left\{ \begin{array}{l} \tilde{C}_i(t-k), \tilde{r}_{ii}(t-k) \mid k = 1, \dots, L_i^i \\ \tilde{C}_j(t-l), \tilde{r}_{ij}(t-l) \mid j = 1, \dots, J^i, l = 1, \dots, L_j^i \end{array} \right\} \right), \quad i = 1, \dots, I,$$

где C – множество концептов, соответствующих ситуационным признакам; I – число концептов, соответствующее числу ситуационных признаков; R – множество нечетких отношений влияния концептов друг на друга; R_l – подмножество нечетких отношений влияния концептов, воздейст-

вующих на концепт C_i ; J^i – число концептов, непосредственно воздействующих на концепт C_i ; $\tilde{r}_{ii}(t-k)$ – нечеткое отношение влияния концепта C_i на себя в момент $(t-k)$; L_i^i – максимально значимое значение временного лага при влиянии концепта C_i на себя; $\tilde{r}_{ij}(t-l)$ – нечеткое отношение влияния концепта C_j на концепт C_i в момент $(t-l)$; L_j^i – максимально значимое значение временного лага при влиянии концепта C_j на концепт C_i ; $\tilde{C}_i(t)$, $\tilde{C}_i(t-k)$, $\tilde{C}_j(t-l)$ – нечеткие значения концептов C_i и C_j в соответствующие моменты времени.

Значения L_i^i и L_j^i определим в зависимости от максимального количества переходов для возможных «путей» между концептами C_i и C_j , значения которых меньше указанного установленного порога.

Этап 6. Параметрическая настройка когнитивной сетевой структуры МСКМ с целью минимизации ошибок сценарного моделирования:

$$\delta C \rightarrow \min, \quad \delta C : (\delta C_1, \dots, \delta C_I),$$

где δC – ошибка сценарного моделирования в целом; δC_i – ошибка сценарного моделирования для концепта C_i .

Она состоит в подстройке значений $\tilde{r}_{ii}(t-m)$, $m=1, \dots, L_i^i$ и $\tilde{r}_{ij}(t-l)$ $j=1, \dots, J^i$, $l=1, \dots, L_j^i$ моделей сценарной динамики (см. ниже) для всех ее концептов при наличии соответствующих обучающих выборок:

$$\forall C_i : \left(\left\{ \left(\tilde{C}_i^{(k)}(t-1), \dots, \tilde{C}_i^{(k)}(t-L_i^i) \right) \mid j \in 1, \dots, J^i \right\}, \tilde{C}_i^{(k)}(t) \right), \quad k=1, \dots, K_i,$$

где K_i – число примеров обучающей выборки для концепта C_i .

Этап 7. Задание моделей сценарной динамики для всех концептов когнитивной сетевой структуры МСКМ, которые учитывают как нечеткие значения, так и приращения значений концептов [Borisov, 2025]:

$$\tilde{C}_i(t) = \tilde{C}_i(t-1) \oplus \left(\bigoplus_{m=1}^{L_i^i} \left(\Delta \tilde{C}_i(t-m) \otimes \tilde{r}_{ii}(t-m) \right) \right) \oplus \left(\bigoplus_{j=1}^{J^i} \left(\bigoplus_{l=1}^{L_j^i} \left(\Delta \tilde{C}_j(t-l) \otimes \tilde{r}_{ij}(t-l) \right) \right) \right),$$

где $\Delta \tilde{C}_i(t-m)$ – нечеткое приращение значения концепта C_i в $(t-m)$; $\Delta \tilde{C}_j(t-l)$ – нечеткое приращение значения концепта C_j в $(t-l)$; \otimes – нечеткое «взвешивание» нечеткого приращения значения концепта, например, алгебраическое произведение; $\bigoplus_{m=1}^{L_i^i}$, $\bigoplus_{l=1}^{L_j^i}$, \oplus – нечеткое агрегирование влияний концептов.

Этап 8. Задание и моделирование распространения воздействия СР с использованием когнитивной сетевой структуры МСКМ. Для каждой Se_q , $q \in 1, \dots, Q$ выполняется следующая процедура.

Шаг 1. Устанавливаются значения концептов, соответствующие значениям ситуационных признаков каждой из эталонных ситуаций ситуационной сетевой структуры МСКМ.

Шаг 2. Выполняется воздействие СР $G_{p_i,k}$, $i \in 1, \dots, I$, $k \in 1, \dots, K_i$ на соответствующий концепт C_i , изменяющий его значение:

$$\tilde{C}_i'(t - \tau) = \tilde{C}_i(t - \tau) \circ \tilde{G}_{p_i,k},$$

где τ – максимально значимое значение временного лага влияния концептов в когнитивной сетевой структуре МСКМ.

Шаг 3. С использованием моделей сценарной динамики запускается процесс сценарного моделирования для всех концептов, по результатам которого определяются их установившиеся значения.

Шаг 4. Фиксируются изменения значений концептов, по которым определяются нечеткие значения ситуационных признаков промежуточных ситуаций $Str = \{Str_m \mid m \in 1, \dots, M\}$.

Этап 9. Формирование фактической ситуационной сетевой структуры МСКМ, ситуационные переходы которой взвешены в зависимости от различных стратегий принятия СР на основе предлагаемого способа.

Шаг 1. С учетом порога нечеткой близости (сходства) нечетких ситуаций выполняется кластеризация эталонных и промежуточных ситуаций. В результате определяются центры кластеров для подмножеств нечетко схожих эталонных и промежуточных ситуаций из Se и Str , и формируется множество фактических ситуаций $Sf = \{Sf_k \mid k \in 1, \dots, K\}$.

Значения показателей этих центров кластеров представляют собой значения ситуационных признаков фактических нечетких ситуаций.

Шаг 2. Определение СР, применение которых приводит к соответствующим фактическим ситуациям в фактической ситуационной сетевой структуре МСКМ.

Требуется «восстановить» матрицы комбинированных нечетких отношений \tilde{G}_{Sf_k, Sf_l}^* , задающие фактические СР G_{Sf_k, Sf_l}^* , $(G^* = \{G_{Sf_k, Sf_l}^* \mid k \in 1, \dots, K, l \in 1, \dots, L_k\})$, композиции которых с матрицами нечетких значений ситуационных признаков исходных фактических ситуаций Sf_k приводят к соответствующим результирующим фактическим ситуациям Sf_l ($\forall Sf_k, Sf_l \in Sf$).

Шаг 3. Формирование ситуационных переходов ситуационной сетевой структуры МСКМ на основе сформированных фактических нечетких ситуаций и СР. Ситуационные переходы $U_{Sf_{beg}, Sf_{end}}$ представляются в виде дуг между исходными Sf_{beg} и конечными Sf_{end} фактическими нечеткими ситуациями, соответствующих всем фактическим СР G_{Sf_k, Sf_i}^* , $\forall G_{Sf_k, Sf_i}^* \in G^*$.

Множество фактических ситуационных переходов:

$$U = \left\{ U_{Sf_{beg}, Sf_{end}} \mid Sf_{beg}, Sf_{end} \in Sf \right\}.$$

Шаг 4. Определение стратегий СР для достижения целевых ситуаций В зависимости от критериев и ограничений: «качественная» – минимизация числа ситуационных переходов для перехода из текущей в целевую ситуацию; «экономичная» – минимизация затратности ресурсов при переходе из текущей в целевую ситуацию; «безопасная» – минимизация рисков негативных последствий при переходе из текущей в целевую ситуацию; «сбалансированная» – максимизация среднего веса ситуационных переходов [Борисов и др., 2021].

Шаг 5. Задание нечетких весов фактических ситуационных переходов в зависимости от стратегии СР для достижения целевых ситуаций.

Таким образом, формируется фактическая ситуационная сетевая структура МСКМ, в которой зависимость ситуационных признаков «купируется» формированием фактических ситуаций и фактических СР в сетевой ситуационной структуре МСКМ в результате сценарного моделирования с использованием когнитивной сетевой структуры МСКМ.

2. Пример использования мягких ситуационно-когнитивных моделей для интеллектуальной поддержки принятия решений

Рассмотрим задачу поддержки принятия решений при управлении центробежным компрессором. Пусть заданы ситуационные признаки: P_1 – давление сжатого газа после компрессора; P_2 – температура сжатого газа после компрессора; P_3 – расход сжатого газа. Для типизации описания зададим одни те же термы для всех ситуационных признаков $T_i = \{T_1^{(i)} - \text{малый}, T_2^{(i)} - \text{средний}, T_3^{(i)} - \text{большой}\}$ на соответствующих базовых множествах D_i , $i = 1, \dots, 3$.

На изменение P_i могут воздействовать следующие СР: $G_{P_i,1}$ – «Слабо уменьшить значение P_i »; $G_{P_i,2}$ – «Средне уменьшить значение P_i »; $G_{P_i,3}$ – «Сильно уменьшить значение P_i »; $G_{P_i,4}$ – «Не изменять значение P_i »; $G_{P_i,5}$ – «Слабо увеличить значение P_i »; $G_{P_i,6}$ – «Средне увеличить значение P_i »; $G_{P_i,7}$ – «Сильно увеличить значение P_i » (см. напр., [Мелихов и др., 1990]).

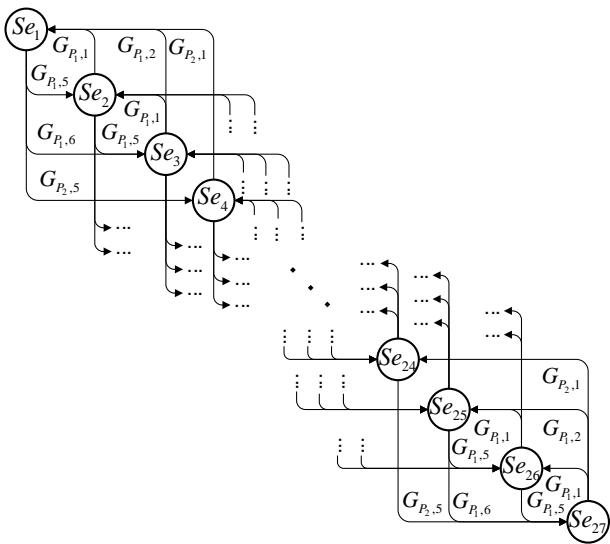


Рис. 1. Эталонная ситуационная сетевая структура МСКМ

Построим когнитивную сетевую структуру МСКМ: задав множество концептов (C_1 – давление сжатого газа после компрессора; C_2 – температура сжатого газа после компрессора; C_3 – расход сжатого газа), соответствующее множеству ситуационных признаков (рис. 2); представив значения отношений влияния концептов друг на друга с учетом временных лагов (табл. 1).

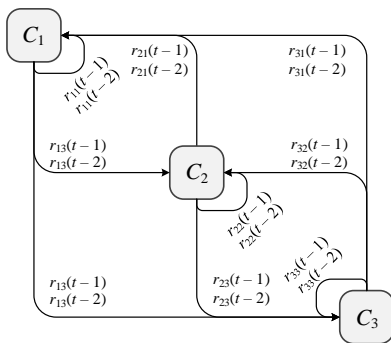


Рис. 2. Пример когнитивной сетевой структуры МСКМ

Таблица 1

R		C_1	C_2	C_3
C_1	$(t-1)$	0,9	0,8	0,7
	$(t-2)$	0,7	0,5	0,4
C_2	$(t-1)$	-0,6	0,9	0,8
	$(t-2)$	-0,3	0,8	0,3
C_3	$(t-1)$	0,6	-0,5	0,9
	$(t-2)$	0,3	-0,2	0,7

Зададим воздействие СР и затем выполним сценарное моделирование с использованием моделей сценарной динамики в когнитивной сетевой структуре МСКМ. Например, при воздействии СР $G_{P_1,5}$ на концепт C_1 в момент $(t-2)$ значение этого концепта изменится следующим образом:

$$\tilde{C}'_1(t-2) = \tilde{C}_1(t-2) \circ \tilde{G}_{P_1,5} = \begin{bmatrix} 1,0 & 0,0 & 0,0 \end{bmatrix} \circ \begin{bmatrix} 1,0 & 0,8 & 0,1 \\ 0,0 & 1,0 & 0,8 \\ 0,0 & 0,0 & 1,0 \end{bmatrix} = \begin{bmatrix} 1,0 & 0,8 & 0,1 \end{bmatrix},$$

Отметим, что момент $(t-2)$ выбран с учетом максимально значимого значения временного лага влияния концептов в когнитивной сетевой структуре МСКМ.

В результате последующего сценарного моделирования из начальной эталонной ситуации Se_q с учетом СР $G_{P_1,5}$, воздействующего на концепт C_1 в момент $(t-2)$, значения концептов установились следующим образом:

$$\tilde{C}_1 = \begin{bmatrix} 1,0 & 0,6 & 0,2 \end{bmatrix}, \quad \tilde{C}_2 = \begin{bmatrix} 0,2 & 1,0 & 0,3 \end{bmatrix}, \quad \tilde{C}_3 = \begin{bmatrix} 1,0 & 0,1 & 0,4 \end{bmatrix}.$$

И, таким образом, по установленным значениям концептов когнитивной сетевой структуры МСКМ уточняется k -локальность воздействия СР $G_{P_1,5}$, а также определяется промежуточная ситуация Str_1 :

$$\tilde{str}_1 = \left\{ \left\{ \left(1,0 / T_1^{(P_1)} \right), \left(0,6 / T_2^{(P_1)} \right), \left(0,2 / T_3^{(P_1)} \right) / P_1 \right\}, \left\{ \left(0,2 / T_1^{(P_2)} \right), \left(1,0 / T_2^{(P_2)} \right), \left(0,3 / T_3^{(P_2)} \right) / P_2 \right\}, \left\{ \left(1,0 / T_1^{(P_3)} \right), \left(0,1 / T_2^{(P_3)} \right), \left(0,4 / T_3^{(P_3)} \right) / P_3 \right\} \right\}.$$

С учетом порога нечеткой близости (сходства) нечетких ситуаций выполним кластеризацию эталонных и промежуточных ситуаций для формирования множества фактических ситуаций.

Далее, решая систему нечетких уравнений ([Гавалец и др., 2011]), определим СР, применение которых приводит к фактическим ситуациям в фактической ситуационной сетевой структуре МСКМ.

Пусть в результате построения МСКМ: сформирована ее фактическая ситуационная сетевая структура; ситуация Sf_1 идентифицирована как наиболее близкая к текущей ситуации S_{cur} ; целевой является ситуация Sf_k ; в качестве стратегии принятия СР выбрана стратегия «сбалансированная», заключающаяся в максимизации среднего веса ситуационных переходов:

$$\forall U_{Sf_{beg}, Sf_{end}} \in U : \tilde{W}_{Sf_{beg}, Sf_{end}}^{(C)}.$$

На рис. 3 приведена фактическая ситуационная сетевая структура МСКМ

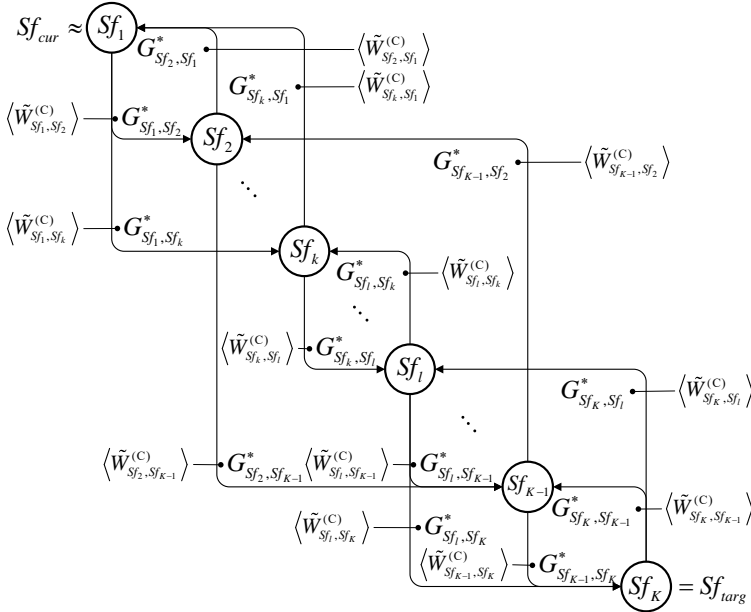


Рис. 3. Фрагмент фактической ситуационной сетевой структуры МСКМ

Для достижения целевой ситуации Sf_k из текущей ситуации Sf_1 возможны следующие траектории:

$$\begin{aligned} d_1(Sf_1, Sf_k) : Sf_1 &\xrightarrow{G^*_{Sf_1, Sf_2} \langle \tilde{W}^{(C)}_{Sf_1, Sf_2} \rangle} Sf_2 \xrightarrow{G^*_{Sf_2, Sf_{K-1}} \langle \tilde{W}^{(C)}_{Sf_2, Sf_{K-1}} \rangle} Sf_{K-1} \xrightarrow{G^*_{Sf_{K-1}, Sf_k} \langle \tilde{W}^{(C)}_{Sf_{K-1}, Sf_k} \rangle} Sf_k; \\ d_2(Sf_1, Sf_k) : Sf_1 &\xrightarrow{G^*_{Sf_1, Sf_k} \langle \tilde{W}^{(C)}_{Sf_1, Sf_k} \rangle} Sf_k \xrightarrow{G^*_{Sf_k, Sf_l} \langle \tilde{W}^{(C)}_{Sf_k, Sf_l} \rangle} Sf_l \xrightarrow{G^*_{Sf_l, Sf_{K-1}} \langle \tilde{W}^{(C)}_{Sf_l, Sf_{K-1}} \rangle} Sf_{K-1} \xrightarrow{G^*_{Sf_{K-1}, Sf_k} \langle \tilde{W}^{(C)}_{Sf_{K-1}, Sf_k} \rangle} Sf_k; \\ d_3(Sf_1, Sf_k) : Sf_1 &\xrightarrow{G^*_{Sf_1, Sf_k} \langle \tilde{W}^{(C)}_{Sf_1, Sf_k} \rangle} Sf_k \xrightarrow{G^*_{Sf_k, Sf_l} \langle \tilde{W}^{(C)}_{Sf_k, Sf_l} \rangle} Sf_l \xrightarrow{G^*_{Sf_l, Sf_k} \langle \tilde{W}^{(C)}_{Sf_l, Sf_k} \rangle} Sf_k, \end{aligned}$$

$$\begin{aligned}\tilde{W}_{sf_1, sf_2}^{(C)} &= 0,8; \tilde{W}_{sf_1, sf_k}^{(C)} = 0,6; \tilde{W}_{sf_2, sf_{k-1}}^{(C)} = 0,5; \tilde{W}_{sf_k, sf_l}^{(C)} = 0,9; \tilde{W}_{sf_l, sf_{k-1}}^{(C)} = 0,7; \\ \tilde{W}_{sf_l, sf_k}^{(C)} &= 0,5; \tilde{W}_{sf_{k-1}, sf_k}^{(C)} = 0,8.\end{aligned}$$

В итоге, наилучшей является траектория $d_2^{(sf_1, sf_k)}$.

Далее с использованием когнитивной сетевой структуры МСКМ выполним сценарное моделирование и оптимизацию применения СР (рис. 4).

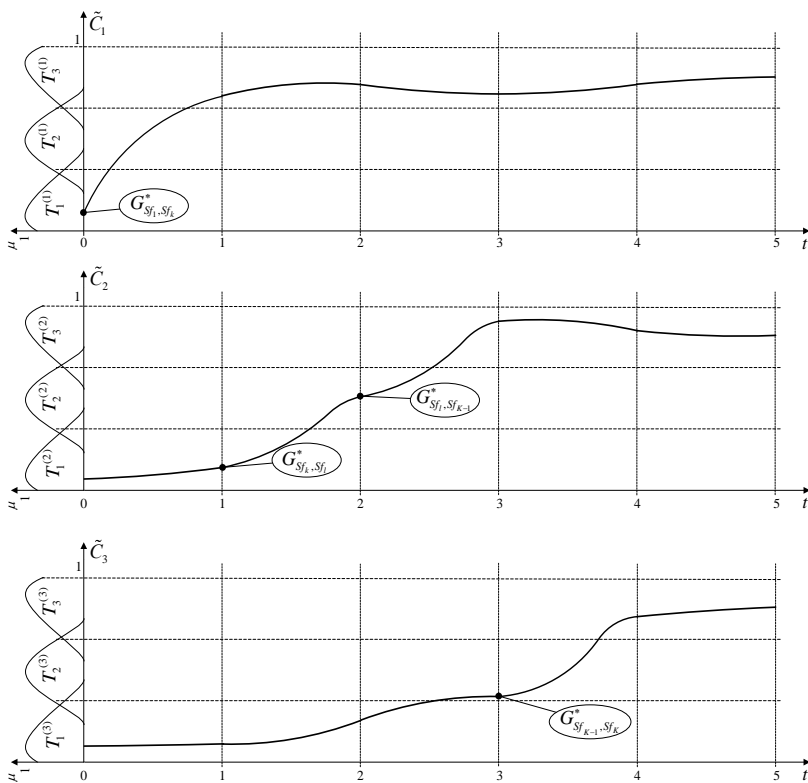


Рис. 4. Результаты сценарного моделирования с использованием когнитивной сетевой структуры МСКМ

Оптимизация применения СР из выбранной последовательности заключается в изменении моментов времени применения СР при сохранении общей их очередности для достижения (по итогам сценарного моделирования) максимальной степени близости к целевой ситуации.

Таким образом, при совместном использовании ситуационной и когнитивной структур МКСМ формируется наилучшая, в зависимости от выбранной стратегии, последовательность ситуативных решений, и обосновывается (по результатам оптимизации) время их применения с учетом реакции ситуационных признаков на их воздействие.

Заключение

Предлагается новая разновидность МКСМ, основанных на полиморфной гибридизации в нечетких ситуационных и когнитивных моделей, обеспечивающей: их структурно-параметрическое соответствие; конвергентность и взаимообусловленность процедур построения структур и структурно-параметрической настройки этих моделей в рамках единой полиморфной модели; их совместное взаимозависимое использование для интеллектуальной поддержки принятия ситуативных решений.

Рассмотрен разработанный метод построения и настройки МКСМ, в рамках которого для оценки воздействия ситуативных решений на зависимые признаки, а также для учета распределенного во времени их воздействия предлагается использовать результаты сценарного моделирования в когнитивной сетевой структуре МКСМ, являющейся двойственной к ситуационной сетевой структуре МКСМ.

Рассмотрен пример интеллектуальной поддержки принятия решений при совместном использовании ситуационной и когнитивной структур МКСМ, иллюстрирующий формирование наилучшей, в зависимости от выбранной стратегии, последовательности ситуативных решений, и обоснование (по результатам оптимизации) времени их применения с учетом реакции ситуационных признаков на их воздействие.

Список литературы

- [Борисов и др., 2009] Борисов В.В., Зернов М.М. Реализация ситуационного подхода на основе нечеткой иерархической ситуационно-событийной сети // Искусственный интеллект и принятие решений. – 2009. – № 1. – С. 17-30.
- [Борисов и др., 2016] Борисов В.В., Денисенков М.А., Федулов А.С. Нечеткие ситуационные сети на основе когнитивных карт // Естественные и технические науки. – 2016. – № 2(92). – С. 118-123.
- [Борисов и др., 2021] Борисов В.В., Авраменко Д.Ю. Нечеткое ситуационное управление сложными системами на основе их композиционного гибридного моделирования // Системы управления, связи и безопасности. – 2021. – № 3. – С. 207-237.
- [Борисов и др., 2024] Борисов В.В., Федулов А.С., Федулова С.А. Интеллектуальная поддержка принятия решений на основе мягких ситуационно-когнитивных моделей // Сб. трудов XXVII Российской научной конференции «Инжиниринг предприятий и управление знаниями» (ИПУЗ-2024, Москва, РЭУ им Г.В. Плеханова, 2024). В 2-х т. Т. 1. – С. 51-56.

- [Гавалец и др., 2011] Гавалец М., Гад М., Циммерман К. Задачи оптимизации при (max, min)-линейных ограничениях в виде равенств и/или неравенств // Фундаментальная и прикладная математика. – 2011/2012. – Т. 17, № 6. – С. 3-21.
- [Захарова и др., 2020] Захарова А.А., Подвесовский А.Г., Исаев Р.А. Нечеткие когнитивные модели в управлении слабоструктурированными социально-экономическими системами // Информационные и математические технологии в науке и управлении. – 2020. – № 4(20). – С. 5-23.
- [Мелихов и др., 1990] Мелихов А.Н., Берштейн Л.С., Коровин С.Я. Ситуационные советующие системы с нечеткой логикой. – М.: Наука, 1990. – 272 с.
- [Поспелов, 2021] Поспелов Д.А. Ситуационное управление: Теория и практика. – 2-е изд. стереотип. – М.: URSS, 2021. – 288 с.
- [Силов, 1995] Силов В.Б. Принятие стратегических решений в нечеткой обстановке. – М.: ИНПРО-РЕС, 1995. – 228 с.
- [Федулов, 2005] Федулов А.С. Нечеткие реляционные когнитивные карты // Известия РАН. Теория и системы управления. – 2005. – № 1. – С. 120-133.
- [Федунов, 2002] Федунов Б.Е. Механизмы вывода в базе знаний бортовых оперативно советующих экспертных систем // Известия РАН. ТИСУ. – 2002. – № 4. – С. 42-52.
- [Borisov et al., 2004] Borisov V.V., Fedulov A.S. Generalized Rule-Based Fuzzy Cognitive Maps: Structure and Dynamics Model // Lecture Notes in Computer Science. – 2004. – Vol. 3316. – P. 918-922.
- [Borisov et al., 2024] Borisov V.V., Zharkov A.P., Luferov V.S. Temporal Cognitive Models Scenario Modeling Based on Fuzzy Relational Temporal Cognitive Models // Pattern Recognition and Image Analysis. – 2024. – Vol. 34, No. 3. – P. 626-633.
- [Borisov, 2025] Borisov V. Soft Relational Temporal Cognitive Models for Scenario Modeling of Electrical Load Distribution / In: Ilin, I., Youzhong, M. (eds) // Digital Systems and Information Technologies in the Energy Sector. Lecture Notes in Networks and Systems. – 2025. – Vol. 1244. – P. 391-400. – Springer, Cham.
- [Carvalho et al., 2000] Carvalho J.P., Tomé J.A. Rule Based Fuzzy Cognitive Maps Qualitative Systems Dynamics // Proc. sci. conf. of the North American Fuzzy Information Processing Society, NAFIPS 2000. Atlanta.
- [Chon et al., 2023] Chon M.-G., Tam L., Lee H., Kim J.-N. Situational Theory of Problem Solving (STOPS). A Foundational Theory of Publics and Its Behavioral Nature in Problem Solving / C. Botan, E. Sommerfeldt (Eds). Public Relations Theory III: In the Age of Publics (1st ed.). Routledge. 2023. – 556 p.
- [Kim et al., 2021] Kim H.J., Hong H. Predicting Information Behaviors in the COVID-19 Pandemic: Integrating the Role of Emotions and Subjective Norms into the Situational Theory of Problem Solving (STOPS) Framework // Health Communication. – 2021. – Vol. 37(13). – P. 1640-1649.
- [Kosko, 1986] Kosko B. Fuzzy Cognitive Maps // International Journal of Man-Machine Studies. – 1986. – Vol. 24. – P. 65-75.
- [Lin, 2007] Lin F. Situation Calculus / F. van Harmelen, V. Lifschitz and B. Porter (Eds). Handbook of Knowledge Representation. – Elsevier, 2007. – P. 649-669.
- [McCarthy, 2002] McCarthy J. Actions and other events in situation calculus // In Proc. of the 8th International Conference on Principles of Knowledge Representation and Reasoning (KR-2002). – 2002. – P. 615-628.
- [Thulukkanam et al., 2015] Thulukkanam K., Vasuki R. Two New Fuzzy Models Using Fuzzy Cognitive Maps Model and Kosko Hamming Distance // Ultra Scientist. – 2015. – Vol. 27(1)B. – P. 43-55.

УДК 004.92

doi: 10.15622/rcai.2025.034

СТРУКТУРНО-ЧУВСТВИТЕЛЬНЫЙ ПОДХОД К ОРГАНИЗАЦИИ ИНТЕЛЛЕКТУАЛЬНОГО МОНИТОРИНГА НА ОСНОВЕ НЕЧЕТКИХ ЭВОЛЮЦИОНИРУЮЩИХ МОДЕЛЕЙ

А.И. Долгий (*a.dolgiy@vniias.ru*)

С.М. Ковалев (*ksm.1954@yandex.ru*)

АО «НИИАС», Москва

В статье представлен новый принцип организации структурно-чувствительного механизма функционирования нечетких эволюционирующих моделей (efTS), используемых в системах интеллектуального мониторинга. Предложенный механизм обеспечивает синхронизацию динамики системной модели с ходом технологического процесса (ТП) и позволяет на основе анализа происходящих в модели структурных изменений обнаруживать характерные особенности в поведении ТП. Достоинством “структурно-чувствительного” подхода к анализу информации является то, что структурные изменения, происходящие в efTS-модели под воздействием внешних изменений во входных данных, являются более выраженными и более четко проявляются в потоке данных, нежели плавные параметрические регуляции. Это существенно упрощает обнаружение целевых событий в потоке данных мониторинга. Для практического использования структурно-чувствительного подхода в алгоритмах мониторинга предложен формальный критерий структурной чувствительности efTS-модели. На основе приведенного критерия разработана алгоритмически простая процедура анализа структурной чувствительности системной модели с целью ее использования в алгоритмах аналитики данных и машинного обучения.

Ключевые слова: инкрементное обучение, структурная чувствительность, нечеткая эволюционирующая модель, интеллектуальный мониторинг.

Введение

Интеллектуальный мониторинг (ИМ) технологических процессов (ТП) отличается от традиционных контрольно-диагностических и мониторинговых систем наличием компонента знаний, поддерживающего важную

функцию в системе ИМ, обеспечивающую возможность извлечения знаний об объекте мониторинга, их аккумуляцию и постоянное обновление в базе знаний (БЗ) мониторинговой системы. Наличие данного компонента в системе ИМ обеспечивает возможность решения широкого круга задач по прогнозированию поведения ТП при изменяющихся условиях функционирования, своевременному, обнаружению отклонений и нарушений в поведении ТП, предупреждению негативных вариантов его развития. Благодаря знаниям, сформированным в БЗ систем ИМ, удастся оптимизировать решение многих оперативно-диспетчерских задач по управлению ТП.

В рамках ИМ критически важным является решение специального класса FDD-задач [Zhao, 2020], составляющих основу характерного для современных систем управления функционала контроля и управления аномалиями (АЕМ-функционала). Реализация АЕМ-функционала позволяет выйти на создание принципиально нового класса высокоавтоматизированных систем управления без привлечения или с минимальным участием человеческого фактора. Однако, реализация АЕМ-функционала в системе ИМ сопряжена с рядом особенностей, которые выдвигают соответствующие требования к разработке базовых математических моделей для систем ИМ, среди которых ключевую роль играет модель нечеткой эволюционирующей системы.

1. Эволюционирующие нечеткие системы

Эволюционирующие нечеткие системы берут свое начало с новаторской работы Сугено и его коллег [Takagi, 1985], в которых авторами была представлена нечеткая модель Такаги-Сугено (TS-модель). Данный класс моделей основан на нечетких правилах вида:

$$r_i: \text{IF } x_1 = \alpha_{i1} \text{ AND } \dots \text{AND } x_m = \alpha_{im} \text{ THEN } C_i = a_{i0} + \sum_{j=1}^m a_{ij}x_j, \quad (1.1)$$

где $\mathbf{x}_t = (x_1(t), \dots, x_m(t))$ – m-мерный вектор входных переменных; α_{ij} – лингвистические значения (нечеткие термы) переменных x_j ; $\mu_{ij}(x_j)$ – функции принадлежности (ФП) нечетких термов α_{ij} ; C_i – целевая переменная состояния процесса.

Распространенной формой представления нечеткого правила (1.1) является выражение в форме нечеткой продукции:

$$r_i: \mathcal{A}_i \Rightarrow C_i, \quad (1.2)$$

где $\mathcal{A}_i = \&_{j=1}^m \mu_{ij}(x_j)$ – нечетко-логическая формула антецедента; C_i – целевая переменная, в качестве которой могут выступать класс состояний ТП или прогнозируемое значение параметра ТП, вычисляемое на основании линейной функции консеквента $C_i = a_{i0} + \sum_{j=1}^m a_{ij}x_j$; $\&$ – оператор нечеткой конъюнкции; \Rightarrow – оператор нечеткой импликации.

В практических задачах нечеткое правило (1.1) выступает в качестве формализованной модели эмпирического знания о связи ситуации принятия решения, представленной в антецеденте нечеткого правила в виде лингвистического описания, с решением, представленным в консеквенте нечеткого правила в виде значения целевой переменной C_i .

Обобщенное описание ситуации принятия решений или состояния ТП, характеризует бесконечное множество схожих между собой ситуаций, имеющих одинаковые или близкие значения параметров $x_j \in \mathbf{x}$. Для каждой конкретной ситуации, представленной в виде вектора параметров \mathbf{x} , степень ее схожести с обобщенной ситуацией \mathcal{A}_i определяется на основе ФП антецедента:

$$\rho(\mathcal{A}_i|\mathbf{x}) = \&_{j=1}^m \mu_{ij}(x_j) \quad (x_j \in \mathbf{x}). \quad (1.3)$$

Величина $\rho(\mathcal{A}_i|\mathbf{x})$, называемая степенью активации нечеткого правила r_i и обозначаемая в дальнейшем через $\rho(r_i)$ либо ρ_i , используется в механизме принятия решений TS-модели, основными элементами которого является БЗ и нечеткая схема вывода. Базу знаний TS-модели образует согласованная совокупность нечетких правил $\{r_i\}$ ($i = 1, \dots, q$).

Нечеткая схема вывода предназначена для вычисления значений целевой переменной C в ситуации, представленной вектором параметров \mathbf{x} , с использованием всех нечетких правил БЗ:

$$C_\Sigma = \sum_{i=1}^q \Psi_i \cdot C_i(\mathbf{x}), \quad (1.4)$$

где q – число нечетких правил в БЗ TS-модели; $C_i(\mathbf{x})$ – значение функции консеквента нечеткого правила r_i , вычисленное для вектора \mathbf{x} ; Ψ_i – вклад нечеткого правила r_i в результирующую оценку C_i целевой переменной, вычисляемый на основании формулы:

$$\Psi_i = \frac{\rho_i(\mathbf{x})}{\sum_{j=1}^q \rho_j(\mathbf{x})} \quad (1.5)$$

$\rho_i(\mathbf{x})$ – степень активации нечеткого правила r_i в ситуации \mathbf{x} , вычисляемая на основании (1.3)

В системе ИМ эволюционирующая efTS-модель базируется на TS-модели, элементы которой применительно к efTS приобретают следующую интерпретацию [Kovalev et al., 2020]:

- нечеткие правила r_i : $\mathcal{A}_i \Rightarrow C_i$, в БЗ efTS ассоциируются с состояниями ТП, при этом степень активации $\rho(\mathcal{A}_i|\mathbf{x}_t)$ нечеткого правила r_i является мерой соответствия текущего состояния ТП, представленного в потоке данных вектором параметров \mathbf{x}_t , ее лингвистическому описанию \mathcal{A}_i , представленному в антецеденте нечеткого правила;

- efTS выступает в качестве предиктивной модели, предназначенной для on-line прогнозирования целевой переменной C_t во входном потоке данных $(\mathbf{u}_t | \mathbf{u}_t = \langle \mathbf{x}_t, C_t \rangle; \mathbf{x}_t \in X^m, t = 1, 2, \dots)$, где первая компонента

$\mathbf{x}_t = (x_{1t}, x_{2t}, \dots, x_{mt})$ представляет набор текущих значений параметров ТП, а вторая – значение целевой переменной $C_t = \mathbf{x}_{t+1}$, в качестве которой выступает вектор прогнозируемых оценок параметров $\mathbf{x}_{t+1} = (x_{1t+1}, x_{2t+1}, \dots, x_{mt+1})$, которые eFTS стремится на каждом шаге предсказать;

- eFTS является динамической системой, функционирование (эволюционирование) которой осуществляется в инкрементном (пошаговом) режиме под управлением потока данных и инкрементного механизма обучения путем реализации для каждого вновь поступившего образца данных \mathbf{x}_t схемы нечеткого вывода (1.3)-(1.5).

В процессе вывода вычисляются значения активаций $\rho_i(\mathbf{x}_t)$ нечетких правил $r_i \in KB_t$. Поскольку, нечеткие правила r_i ассоциируются с состояниями ТП, то очевидно, нечеткие правила, имеющие большие значения активаций $\rho_i(\mathbf{x}_t)$, в большей мере соответствуют реальным состояниям, в которых находится ТП в текущий момент времени t . Поэтому, естественно связать состояние ТП с обобщенной ситуацией, указанной в антецеденте наиболее активного нечеткого правила, то есть:

$$St(\mathcal{A}_i) \Leftrightarrow \mathcal{A}_i = \arg \max_{r_i \in KB} |r_i: \mathcal{A}_i \Rightarrow C_i| (\rho(\mathcal{A}_i | \mathbf{x})). \quad (r_i: \mathcal{A}_i \Rightarrow C_i), \quad (1.6)$$

где $St(\mathcal{A}_i)$ – состояние ТП, представленное лингвистическим описанием \mathcal{A}_i .

Соответствующее формальное определение состоянию ТП через активации нечетких правил будет дано несколько ниже.

2. Организация механизма online обучения нечеткой эволюционирующей модели

Как отмечалось выше, эволюционирующая eFTS-модель, является динамической системой, функционирование которой осуществляется в пошаговом режиме под управлением потоковых данных. Это означает, что формирование, обучение и адаптация модели осуществляются синхронно с непрерывно поступающими на вход системной модели данными. Ключевую роль в этом процессе играет механизм online обучения на основе инкрементной субтрактивной кластеризации [Chiu, 1994].

Основная идея инкрементного обучения на основе субтрактивной кластеризации заключается в поэтапном формировании в пространстве параметров ТП X^m групп (кластеров) близко расположенных точек и последующего извлечения из полученных кластеров нечетких правил для БЗ eFTS.

Информация для обучения eFTS накапливается в режиме online по мере поступления новых данных. Каждый новый образец данных характеризует новую информацию о текущем состоянии ТП, представленную точкой \mathbf{x}_t в пространстве числовых параметров X^m . Предположим, что в течение интервала времени Δt исследуемый ТП находится в одном из стабильных состояний St_t . В этом случае накопленные за время Δt данные образуют в

пространстве X^m кластер \mathcal{K}_t схожих друг с другом точек $\mathbf{x} \in \mathcal{K}_t$, и все последующие образцы входных данных \mathbf{x}_{t+1}, \dots будут группироваться в этом же кластере, пока ТП будет находиться в прежнем состоянии St_t . Изменение состояния ТП в момент t^* сопровождается изменением числовых параметров вектора \mathbf{x}_{t^*} и появлением во входном потоке данных дрейфа концепции [Lu et. al., 2018]. В результате чего последующие данные $\mathbf{x}_{t+1}, \mathbf{x}_{t+2}, \dots (t > t^*)$, описывающие дальнейшее развитие ТП, начнут концентрироваться в другой области признакового пространства X^m , т.е. образовывать новый кластер $\mathcal{K}_{t+1} \subset Z$, характеризующий новое состояние ТП. Таким образом, по мере своего эволюционирования efTS-модель формирует в пространстве X^m последовательность кластеров $\mathcal{K}_1, \mathcal{K}_2, \dots, \mathcal{K}_q$ ($\mathcal{K}_i \subset X^m$), которая, в свою очередь, индуцирует последовательность извлекаемых из них нечетких правил $r_{i1}, r_{i2}, \dots, r_{iq}$, образующих динамическую базу знаний $KB_t = \{r_i | i = 1, \dots, q\}$ эволюционирующей efTS-модели.

Поскольку активации $\rho_i(\mathbf{x}_t)$ нечетких правил $r_i \in KB_t$ характеризуют меру соответствия текущих параметров ТП \mathbf{x}_t состояниям ТП, указанным в антецедентах нечетких правил, представляется естественным связать состояния ТП с активациями нечетких правил динамической БЗ, формируемой efTS-моделью под управлением потока данных $(\mathbf{u}_t | \mathbf{u}_t = \langle \mathbf{x}_t, C_t \rangle; t \in \mathbb{N})$. Для увязки активаций нечетких правил с состояниями контролируемого процесса введем в рассмотрение понятие нечеткого множества “активных” нечетких правил.

Определение 2.1. Пусть KB_t – динамическая БЗ, сформированная к текущему моменту времени t в процессе эволюционирования efTS-модели под управлением потока данных $(\mathbf{u}_t | \mathbf{u}_t = \langle \mathbf{x}_t, C_t \rangle; t \in \mathbb{N})$. Нечетким множеством \tilde{R}_t “активных” нечетких правил в момент t называется нечеткое множество:

$$\tilde{R}_t = \{ \langle r_i, \mu_{\tilde{R}_t}(r_i) \rangle | r_i \in KB_t, \mu_{\tilde{R}_t}(r_i) = \rho_i(\mathbf{x}_t) \}, \quad (2.1)$$

где $\rho_i(\mathbf{x}_t)$ – активации нечетких правил, вычисляемые на основании (1.3). Понятие нечеткого множества активных правил положено в основу определения состояния ТП.

Определение 2.2. Пусть $(\mathbf{u}_t | \mathbf{u}_t = \langle \mathbf{x}_t, C_t \rangle; t \in \mathbb{N})$ – поток входных данных, сопровождающий течение ТП, а KB_t – динамическая БЗ, сформированная в процессе эволюционирования efTS под управлением входного потока данных. Обобщенным состоянием ТП St_t в текущий момент времени t называется нечеткое множество \tilde{R}_t “активных” нечетких правил динамической базы знаний KB_t :

$$St_t = \{ \langle r_i, \mu_{\tilde{R}_t}(r_i) \rangle | r_i \in KB_t, \mu_{\tilde{R}_t}(r_i) = \rho_i(\mathbf{x}_t) \}, \quad (2.2)$$

где $\rho_i(\mathbf{x}_t)$ – активации нечетких правил r_i .

В контексте ИМ важное значение приобретают задачи выявления знаний о ходе ТП, которые тесно связаны с оценкой состояний и обнаружением фактов смены состояний. Здесь ключевую роль играет понятие пространства активаций нечетких правил.

Определение 2.3. Пространством активаций \mathfrak{Z} динамической базы знаний KB_t называется нечеткое множество активаций $\rho_i \in [0,1]$ нечетких правил $r_i \in KB_t$ в q -мерном пространстве нечетких истинностных значений $\mathfrak{Z} \subset [0,1]^q$, образованном активациями нечетких правил базы знаний efTS-модели:

$$\mathfrak{Z} = \{ \langle \rho(r_1|\mathbf{x}_t), \rho(r_2|\mathbf{x}_t), \dots, \rho(r_q|\mathbf{x}_t) \rangle \mid r_i \in KB_t; \rho(r_i|\mathbf{x}_t) \in [0,1] \}, \quad (2.3)$$

где q – общее количество нечетких правил в БЗ efTS-модели, сформированной на момент времени t ;

3. Структурная чувствительность системной модели и принцип стабильной пластичности

В рамках ИМ решается комплекс задач, связанных с оценкой состояний, идентификацией нарушений и прогнозированием поведения ТП. С этой целью efTS должна не только осуществлять прогнозирование целевых переменных, но и адекватно описывать динамику контролируемого процесса, проявляемую в смене его состояний.

При разработке такого класса моделей специалисты сталкиваются с рядом проблем, одной из которых является поиск баланса между стабильностью разрабатываемой модели, обеспечивающей ее оптимальность на достаточно длительном временном интервале, и пластичностью, обеспечивающей способность модели динамично реагировать на входные данные. Такой баланс в ряде работ назван дилеммой стабильной пластичности (stability-plasticity dilemma) [Abraham et al., 2005]. Для решения этой проблемы предлагается придать efTS принципиально новое качество, называемое структурной чувствительностью [Долгий и др., 2018].

Структурная чувствительность характеризует способность efTS-модели на основе анализа происходящих в ней структурных изменений обнаруживать характерные особенности в поведении исследуемого процесса, вызванные сменой его состояний, появлением особых классов нештатных и/или целевых состояний, или появлением предшествующих им предикторных состояний.

Структурная чувствительность efTS-модели является важным с практической точки зрения свойством эволюционирующей модели. Оно обуславливает возможность на основе анализа структурных изменений в БЗ efTS, выявлять существенные изменения в потоке данных и на этой основе обнаруживать характерные особенности в поведении

исследуемого ТП. Достоинством “структурно-чувствительного” подхода к анализу информации является то, что структурные изменения, происходящие в efTS-модели под воздействием внешних изменений во входных данных, являются более выраженными нежели плавные параметрические регуляции, вызванные воздействием тех же данных, что упрощает реализацию механизма принятия решений.

Для обеспечения структурной чувствительности необходимо организовать инкрементный механизм обучения так, чтобы efTS-модель была способна в пошаговом режиме соответствующим образом реагировать на изменения во входном потоке данных $(\mathbf{u}_t | \mathbf{u}_t = \langle \mathbf{x}_t, \mathbf{x}_{t+1} \rangle; t \in \mathbb{N})$. То есть, при каждом поступлении нового образца данных, сопровождающего смену состояний ТП, база знаний efTS-модели должна обновляться путем появления новых нечетких правил или замены одной группы активных нечетких правил другой группой активных правил. Обновление БЗ efTS-модели осуществляется на основе инкрементного механизма структурной адаптации.

Структурная чувствительность efTS-модели обеспечивается инкрементным механизмом обучения [Dolgiy et. al., 2022]. На основании Определения 2.2 он должен быть организован таким образом, чтобы каждый новый образец данных \mathbf{x}_t , сопровождающий смену состояний ТП, одновременно приводил и к смене “активных” групп нечетких правил в БЗ efTS-модели. Поэтому, в основу анализа и формализации условия структурной чувствительности можно положить анализ условий нарушения равенства между нечеткими множествами “активных правил” [Бершт] в БЗ efTS-модели в переходные моменты времени.

Критерий структурной чувствительности. Пусть $(\mathbf{u}_t | \mathbf{u}_t = \langle \mathbf{x}_t, C_t \rangle; t \in \mathbb{N})$ – входной поток данных, сопровождающий течение ТП, а KB_t – динамическая БЗ efTS-модели, сформированная к моменту времени t в процессе эволюционирования efTS под управлением входных данных. Тогда, efTS-модель является структурно чувствительной к входному потоку данных, если и только если выполняется условие:

$$\forall t \in T \quad St_t = St_{t+1} \quad \Leftrightarrow \quad (\tilde{R}_{t+1} \subseteq \tilde{R}_t), \quad (3.1)$$

где St_t, St_{t+1} – состояния ТП в смежные моменты времени $t, t+1$; $\tilde{R}_t, \tilde{R}_{t+1}$ – нечеткие множества “активных” правил в динамической БЗ efTS-модели в моменты времени $t, t+1$.

Критерий структурной чувствительности означает, что смена состояний ТП быть обнаружена на основе анализа условия включения одного нечеткого множества в другое. Для этого воспользуемся эквивалентной формой представления критерия, вытекающей из свойств нечетких множеств []. Оно базируется на следующем Утверждении.

Утверждение 3.1. Эволюционирующая efTS-модель обладает структурной чувствительностью если и только если выполняется условие:

$$\forall t \in T \quad S_t \neq S_{t+1} \quad \Leftrightarrow \quad \exists r^* \in KB_t \quad (\rho(r^*|\mathbf{x}_t) < 0.5) \ \& \ (\rho(r^*|\mathbf{x}_{t+1}) > 0.5).$$

На основе Утверждения 3.1 можно предложить алгоритмически простую процедуру анализа структурной чувствительности efTS-модели. Она заключается в проверке степени активаций нечетких правил, входящих в БЗ efTS-модели, до и после текущего момента времени t . Если хотя бы для одного из правил $r^* \in KB_t$ степень его активации $\rho(r^*|\mathbf{x}_t)$ в момент t меняется с меньшего значения $\rho(r^*|\mathbf{x}_t) < 0.5$ на большее $\rho(r^*|\mathbf{x}_{t+1}) > 0.5$, то фиксируется смена состояний ТП в момент времени t .

4. Организация механизма извлечения знаний из потоковых данных

Новые знания в системе ИМ генерируются в моменты перехода ТП из одних состояний в другие. Для этих целей предлагается энтропийный подход и критерий выявления соответствующих паттернов в потоке входных данных, сопровождающих течение ТП.

При нахождении ТП в момент времени t в одном определенном состоянии активации $\rho_{it} \in \mathbf{p}_t$ нечетких правил, характеризующие данное состояние, принимают либо близкие к единице, либо близкие к нулю значения, уменьшая тем самым энтропию их распределения. В процессе смены или в преддверии новых состояний, происходит уменьшение значений активаций от $\rho_{it} \approx 1$ до $\rho_{it} \approx 0$ для одной группы нечетких правил $\rho_{it} \in \mathbf{p}_t$ и одновременно увеличение от $\rho_{jt} \approx 0$ до $\rho_{jt} \approx 1$ для другой группы правил $\rho_{jt} \in \mathbf{p}_t$, описывающих соответственно исчезающее и вновь появляющееся состояние ТП. В результате в переходный момент t^* происходит выравнивание значений активаций нечетких правил в обеих группах. Это приводит к резкому повышению энтропии распределения значений системных переменных, которую можно выразить в виде критерия:

$$S_{t^*} = \sum_{i=1}^q \rho_{it^*} \ln(\rho_{it^*}), \quad (4.1)$$

где q – число нечетких правил в текущей БЗ efTS-модели.

Полученное выражение выступает в качестве критерия смены состояний и в потоке мониторинговых данных. Пороговое значение для статистики S_{t^*} определяется на основе обучения с использованием массива данных, описывающих сценарии нахождения ТП в стабильных состояниях.

Как упоминалось, нечеткие правила $r_i: \mathcal{A}_i \Rightarrow C_i$ эволюционирующей efTS-модели выступают в качестве элементарных единиц знаний о связи параметров ТП с классами состояний. Поэтому стратегия извлечения знаний, реализуемая в системе ИМ, сводится к выявлению фактов смены со-

стояний на основе критерия (3.1) и интерпретации антецедентов нечетких правил, активации которых претерпевают изменения в переходный моменты времени t^* . Измененные антецеденты нечетких правил образуют новые единицы элементарных знаний, заносимые в БЗ efTS-модели.

Следует отметить, что изначально БЗ efTS-модели формируется экспертами и в последствии по мере поступления новых данных эволюционирует под управлением инкрементного механизма online-обучения путем включения/исключения из нее нечетких правил и/или изменения их параметров.

Благодаря интерпретируемости efTS-модели и ее адаптационным свойствам, а также способности efTS извлекать знания из потока данных в online режиме данная модель выбрана в качестве базовой модели для системы ИМ.

Заключение

В статье рассмотрены некоторые важные аспекты организации интеллектуального мониторинга ТП на основе использования нового класса нечетких эволюционирующих моделей, efTS. Отличительными особенностями efTS-модели по отношению к традиционным нечетким моделям являются возможность online обработки потоковых данных на основе заложенного в модель механизма инкрементного обучения, способность извлекать “на лету” из потоковых данных знания в виде нечетких “IF-THEN”-правил, а также возможность интерпретировать состояния ТП путем сопоставления текущих параметров ТП с лингвистическими описаниями состояний ТП, указанными в предусловиях нечетких правил.

В статье также предложен новый принцип организации структурно-чувствительного механизма обучения efTS-модели, обеспечивающего возможность системной модели синхронизировать свое поведение с ходом ТП и обнаруживать на основе анализа происходящих в модели структурных изменений характерные особенности в поведении ТП. Достоинством “структурно-чувствительного” подхода к анализу информации является то, что структурные изменения, происходящие в efTS-модели под воздействием внешних изменений во входных данных, являются более выраженными и более четко проявляются в потоке данных, нежели плавные параметрические регуляции, что упрощает обнаружение соответствующих событий в данных мониторинга. Для практического использования структурно-чувствительного подхода в алгоритмах мониторинга предложен формальный критерий структурной чувствительности efTS-модели. На основе приведенного критерия возможна разработка алгоритмически простой процедуры анализа структурной чувствительности системной модели для ее использования в алгоритмах аналитики данных и машинного обучения.

Список литературы

- [Zhao et al., 2020] Zhao Yang, et al. A review of data mining technologies in building energy systems: Load prediction, pattern identification, fault detection and diagnosis // *Energy and Built Environment*. – 2020. – 1.2. – P. 149-164.
- [Takagi et al., 1985] Takagi T., Sugeno M. Fuzzy identification of systems and its applications to modeling and control // *IEEE Transactions on Systems, Man and Cybernetics*. – 1985. – 15(1). – P. 116-132.
- [Kovalev et al., 2020] Kovalev S., Kolodenkova A., Sukhanov A. Incremental Structure-Evolving Intelligent Systems with Advanced Interpretational Properties / In: Kuznetsov S.O., Panov A.I., Yakovlev K.S. (eds) // *Artificial Intelligence. RCAI 2020. Lecture Notes in Computer Science*. – Vol. 12412. – Springer, Cham, 2020.
- [Chiu et al., 1994] Chiu S. Fuzzy Model Identification Based on Cluster Estimation // *Journal of Intelligent and Fuzzy Systems*. – 1994. – Vol. 2(3). – P. 267-278.
- [Abraham et al., 2005] Abraham W.C., Robins A. Memory retention – the synaptic stability versus plasticity dilemma // *Trends in Neurosciences*. – 2005. – 28(2). – P. 73-78.
- [Долгий и др., 2018] Долгий А.И., Ковалев С.М. Диагностирование устройств железнодорожной автоматики и подвижного состава на основе иммунного подхода // *Труды Конгресса по интеллектуальным системам и информационным технологиям "IS&IT'18"*. – 2018. – С. 18-26.
- [Dolgiy et al., 2022] Dolgiy A., Khramtsov A., Kovalev S. Intelligent Models for State Assessment and Behavior Prediction in Railway Processes Based on Descriptive Analytics and Soft Computing // *Proceedings of the Sixth International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'22)*, Istanbul, 2022. – Vol. 566. – P. 358-368.
- [Мелихов и др., 1981] Мелихов А.Н., Берштейн Л.С. Конечные четкие и расплывчатые множества: Ч. 11. Расплывчатые множества. – Таганрог: ТРТИ, 1981.
- [Lu et al., 2018] Lu J., Liu A., Dong F., Gu F., Gama J., and Zhang G. Learning under concept drift: A review // *IEEE Transactions on Knowledge and Data Engineering*, – 2018. – 31(12). – P. 2346-2363.

УДК 004.89

doi: 10.15622/rcai.2025.035

МНОГОУРОВНЕВЫЙ ИЕРАРХИЧЕСКИЙ ЭВОЛЮЦИОННЫЙ АЛГОРИТМ РАЦИОНАЛЬНОГО ВЫБОРА ПАРАМЕТРОВ СИСТЕМЫ УПРАВЛЕНИЯ ПРОДВИЖЕНИЕМ В СОЦИАЛЬНЫХ СЕТЯХ

К.В. Захарченков (*zaharchenkovkv@mail.ru*)

А.Е. Мисник (*anton@misnik.by*)

Е.Н. Шеробурко (*sheren@tut.by*)

Белорусско-Российский университет,
Республика Беларусь, Могилев

В работе описан многоуровневый иерархический эволюционный алгоритм рационального выбора параметров системы управления продвижением в социальных сетях. В рамках эволюционного моделирования реализован иерархический генетический алгоритм, позволяющий формировать оптимальные наборы характеристик публикаций и выбирать рациональные решения на основе целевой функции. Механизмы генетических операторов (селекция, скрещивание, мутация, элитарный отбор) обеспечивают поиск глобального оптимума. Отличительной особенностью предложенного алгоритма является рациональный выбор значений характеристик публикаций на первом уровне и рациональный выбор набора публикаций контент-плана на втором уровне. Предложенный алгоритм может использоваться для повышения эффективности продвижения товаров, работ и услуг в социальных сетях.

Ключевые слова: продвижение в социальных сетях, характеристики публикаций, формирование контент-плана, рациональные решения, эволюционное моделирование.

Введение

Интеллектуальная поддержка принятия управленческих решений при формировании стратегии продвижения в социальных сетях позволяет повысить эффективность маркетинговой деятельности предприятия на основе использования эволюционного моделирования для формирования контент-плана.

Предложенный многоуровневый иерархический эволюционный алгоритм рационального выбора параметров системы управления продвижением в социальных сетях позволяет ускорить и повысить качество формирования публикаций в социальных сетях за счет определения рационального набора характеристик публикаций на первом уровне и определении рационального набора публикаций контент-плана на втором уровне.

В результате сокращается время рационального выбора значений характеристик публикаций, формирования контент-плана, соответствующего требованиям целевого сегмента рынка, что ведет к повышению узнаваемости организации и формированию ее положительного имиджа, усилению лояльности потребителей, увеличению объема продаж, повышению прибыли, рентабельности и улучшению иных финансовых показателей.

1. Продвижение в социальных медиа

Маркетинг в социальных медиа (SMM) – это деятельность, которая представляет собой комплекс мероприятий, направленных на поддержание взаимоотношений с потребителями в социальных медиа [Скоробогатых, 2025].

В последние годы социальные сети стали играть значительную роль в маркетинговой деятельности предприятий и организаций. Они оказывают влияние на жизнедеятельность как отдельного человека, так и организаций и общества в целом. Маркетинг социальных медиа является эффективным инструментом, с помощью которого предприятие взаимодействует с потребителями.

Формирование контента является ответственным моментом в формировании стратегии продвижения в социальных сетях. Публикации должны носить регулярный характер, чтобы потребитель не утратил интерес к продукции компании. Вместе с тем, они не должны быть избыточными, чтобы не вызвать негативной реакции, и, как следствие, уход подписчиков. Они должны публиковаться во время наибольшей активности целевой аудитории. Данные об активности можно получить, используя инструменты и сервисы статистики и анализа социальных сетей. Публикации должны быть своевременными и соответствовать настроениям и восприятию целевой аудитории [Смолина, 2024].

Ведение социальных сетей является достаточно трудозатратным процессом, который, вместе с тем, может быть в определенной степени формализовано и автоматизировано с использованием технологий интеллектуальной поддержки принятия решений.

В соответствии с аналитическим докладом Национального центра развития искусственного интеллекта при Правительстве Российской Федерации, 97% организаций, использующих искусственный интеллект, получили тот или иной финансовый или нефинансовый эффект. При этом 36% организаций получили существенный или многократный экономический

эффект от его использования. Наибольший эффект достигается с точки зрения скорости и качества деловых процессов (отмечен 44% организаций) [Индекс, 2024].

Вопросы использования искусственного интеллекта в рамках маркетинговой деятельности в последние годы все больше привлекают внимание отечественных зарубежных ученых.

Будко С.А. и Окрестина О.Р. рассматривают управление технологиями в сфере маркетинга [Будко и др., 2024].

Барина Н.В. и Барин В.Р. изучают вопросы поискового маркетинга, технологии социального медиамаркетинга и краудтехнологии [Барина, 2020].

Казыханов А., Давлетшин, Р.Р., Кузнецова Е.В. изучают возможности использования искусственного интеллекта в маркетинговой деятельности организаций, его актуальности и влиянии на разработку современных бизнес-процессов [Казыханов и др., 2024].

Вопросам анализа и оценки текстового контента посвящены работы А.В. Благова и И.А. Рыцарева. Они рассматривают подходы к классификации текстовых данных в социальных сетях методами k-means с использованием метрики TF-IDF, а также LDA-алгоритма.

K-means – это один из наиболее распространённых алгоритмов кластеризации, который относится к неиерархическим итеративным алгоритмам.

TF-IDF – статистическая мера, используемая для оценки важности слова в контексте документа, являющегося частью коллекции документов или корпуса. Она часто используется в задачах анализа текстов и информационного поиска, например, как один из критериев релевантности документа поисковому запросу, при расчёте меры близости документов при кластеризации.

LDA алгоритм основан на определении наиболее употребляемых тем, которые могут образовывать кластеры. Эта позволяет создать вероятностную модель большой коллекции текстов [Благов, 2020].

Вместе с тем возможности использования эволюционного моделирования в сфере маркетинга социальных сетей в литературе не раскрыты достаточно полно.

Эволюционное моделирование позволит решить важнейшую задачу планирования продвижения в социальных сетях – рациональное распределение временных и финансовых ресурсов для своевременного формирования публикаций и эффективного продвижения организации в социальных сетях [Карпенко, 2017].

2. Постановка задачи рационального выбора параметров системы управления продвижением в социальных сетях

Пусть лицо, принимающее решения (ЛПР) выбирает состав публикаций, из которых состоит контент-план (kp) из множества допустимых составов публикаций ($\{publ_{kpi}\}$). В результате выбора состава публикаций контент-плана, исходя из целей системы управления продвижением, реализуется контент-план, содержащий множество публикаций, необходимых для реализации контент-плана. Каждая публикация характеризуется тематикой $Topics_{kpi}$, привязкой к датам и событиям $\{DE_{publ_{kpi}}\}$ и сроком $\{T_{publ_{kpi}}\}$.

$$kp = \{publ_{kpi} | publ_{kpi} = \langle Topics_{kpi}, DE_{publ_{kpi}}, T_{publ_{kpi}} \rangle\}.$$

Допустимым является состав публикаций, в котором выполнены все условия по темам, привязке к датам и событиям и срокам публикаций.

Задача рационального выбора параметров системы управления продвижением в социальных сетях включает два этапа:

1. Определение рационального набора характеристик публикации, подлежащей размещению.

2. Определение рационального набора публикаций контент-плана.

Пусть $НП = \{\Pi ik\}$ – набор i -х публикаций, $k=1..Ki$, где Ki – количество разновидностей i -ой публикации, подлежащей размещению. $РП = \{P\Pi ik\}$ – стоимость i -й публикации, $i=1..N$, где N – количество публикаций, подлежащих размещению. $QП = \{Q\Pi ik\}$ – количество публикаций, подлежащих размещению.

Пусть $spcj(НПik)$ – нормированная j -я характеристика k -ой позиции набора i -й публикации $НПik$; aij – значимость j -й характеристики i -й публикации для ЛПР, aij – определяются на основе матриц парных сравнений метода Т. Саати.

Целевая функция решения задачи определения рационального набора характеристик публикации, подлежащей размещению, определяется следующим образом:

$$F_{publ} = \sum j aij \cdot spcj(НПik) \rightarrow \max, \quad (2.1)$$

$$QL(\Pi ik) \in [QL\Piimin; QL\Piimax], T(\Pi ik) \in [T\Piimin; T\Piimax].$$

Целевая функция решения задачи определения рационального набора публикаций контент-плана имеет вид:

$$F_{kp} = \sum i P\Pi ik \cdot Q\Pi ik \rightarrow \min. \quad (2.2)$$

3. Алгоритм определения рационального набора значений характеристик публикации

Опишем основные шаги алгоритма [Заровчатская, 2025].

Подготовительный шаг. Задаются параметры генетического алгоритма:

- количество публикаций N_{publ} ;
- количество итераций N_{step} ;
- номер итерации $i=1$.

Особь в популяции (генотип) представляет собой массив, содержащий значения характеристик публикации.

ЛПР принимает решение по первоначальному формированию значений характеристик публикаций.



Рис. 1. Алгоритм определения характеристик публикации

Шаг 1. Вычисляется целевая функция для каждого варианта значений характеристик.

Шаг 2. Операция селекции. Используется турнирный отбор. Проводится два раунда турнирного отбора. В каждом раунде случайным образом отбирается заданное количество вариантов значений характеристик публикаций. Родителем становится вариант распределения с наилучшей целевой функцией.

Шаг 3. Операция скрещивания. Используется кроссинговер порядка. Случайным образом выбирается точка разрыва, в соответствии с которой генотип набора значений характеристик публикации (родителя) делится на две секции, и родители обмениваются ими.

Шаг 4. Операция мутации. Используется мутация, которая заключается в замене значения характеристики публикации другим значением из множества возможных вариантов.

Результатом шага 4 являются новые варианты наборов значений характеристик публикаций.

Шаг 5. Сокращение количества вариантов наборов значений характеристик публикаций. Применяется стратегия элитарного отбора. Промежуточная популяция включает исходные варианты значений характеристик публикации, варианты значений характеристик публикации, полученные в результате скрещивания и мутации. Для дальнейшей работы выбирается подмножество вариантов значений характеристик публикаций численностью N_{publ} с наилучшими значениями целевой функции.

Проверяются условия:

- если $i < N_{step}$, то $i = i + 1$, и выполняется переход к шагу 2;
- если $i = N_{step}$, то выполняется переход к шагу 6;
- если наилучшее значение целевой функции не улучшилось, то выполняется переход к шагу 6.

Шаг 6. Завершение работы алгоритма. В качестве решения задачи выбирается вариант значений характеристик публикации с наилучшим значением целевой функции из последней популяции.

4. Алгоритм определения рационального набора публикаций контент-плана

На втором этапе формируется контент-план с набором публикаций, рациональные значения характеристик которых определены на предыдущем этапе (рис. 2).

Опишем основные шаги алгоритма [Заровчатская, 2025].

Подготовительный шаг. Задаются параметры генетического алгоритма:

- количество вариантов контент-плана $N_{кр}$;
- количество итераций N_{step} ;
- номер итерации $i = 1$.

Особь в популяции (генотип) представляет собой массив, содержащий набор публикаций, входящих в контент-план.

ЛПР принимает решение по первоначальному формированию набора публикаций, входящих в контент-план.

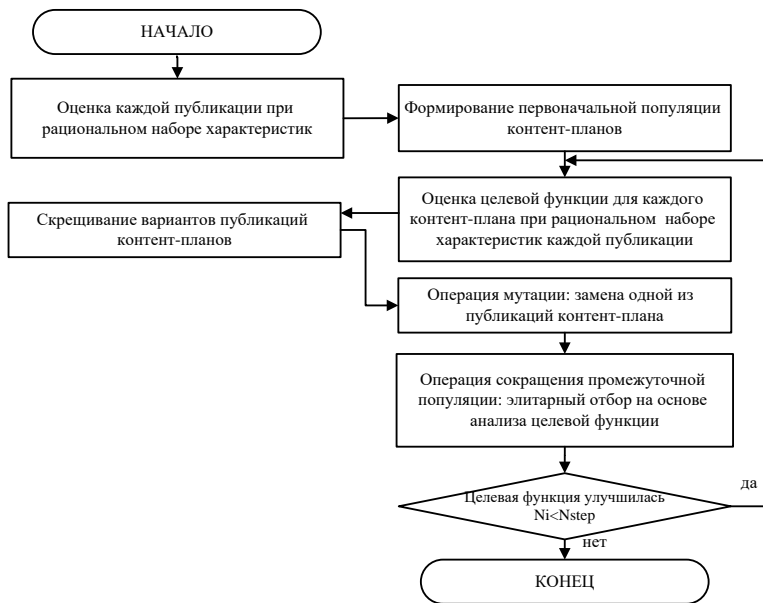


Рис. 2. Схема иерархического эволюционного алгоритма формирования контент-плана

Шаг 1. Вычисляется целевая функция для каждого варианта контент-плана.

Шаг 2. Операция селекции. Используется турнирный отбор. Проводится два раунда турнирного отбора. В каждом раунде случайным образом отбирается заданное количество вариантов контент-плана. Родителем становится вариант распределения с наилучшей целевой функцией.

Шаг 3. Операция скрещивания. Используется кроссинговер порядка. Случайным образом выбирается точка разрыва, в соответствии с которой генотип контент-плана родителей делится на две секции, и родители обмениваются публикациями, содержащимися в этих секциях.

Шаг 4. Операция мутации. Используется мутация, которая заключается в замене публикации в контент-плане другой публикацией из множества публикаций.

Результатом шага 4 являются новые варианты контент-планов.

Шаг 5. Сокращение количества вариантов контент-планов. Применяется стратегия элитарного отбора. Промежуточная популяция включает исходные варианты контент-планов, варианты контент-планов, полученные в результате скрещивания и мутации. Для дальнейшей работы выбирается подмножество вариантов распределения численностью N_{kr} с наилучшими значениями целевой функции.

Проверяются условия:

- если $i < N_{step}$, то $i=i+1$, и выполняется переход к шагу 2;
- если $i=N_{step}$, то выполняется переход к шагу 6;
- если наилучшее значение целевой функции не улучшилось, то выполняется переход к шагу 6.

Шаг 6. Завершение работы алгоритма. В качестве решения задачи выбирается вариант контент-плана с наилучшим значением целевой функции из последней популяции.

5. Экспериментальная проверка алгоритма

Экспериментальная проверка алгоритма проводилась в рамках формирования контент-плана публикаций в социальных сетях для продвижения мобильных регистраторов выполнения сварочных работ. Число публикаций в контент-плане варьировалось от 10 до 30, а число вариантов контент-планов – от 10 до 20.

Для оценки качества публикаций использовались стандартные KPI-метрики – охват, коэффициент вовлеченности (количество оценок «мне нравится», комментарии, репосты), коэффициент конверсии, увеличение узнаваемости. Для оценки качества контент-плана – минимизация общего объема инвестиций в продвижение.

Результаты работы алгоритма сравнивались с результатами формирования контент-плана, выполняемого SMM-специалистом без использования технологий искусственного интеллекта.

Вероятность скрещивания выбиралась равной 0,8, а вероятность мутации равной 0,05. Значение целевой функции решения задачи определения рационального набора характеристик публикации, подлежащей размещению F_{publ} , рассчитывалось по формуле (2.1). Значение целевой функции решения задачи определения рационального набора публикаций контент-плана F_{cp} – по формуле (2.2).

Показатели работы алгоритма формирования характеристик публикаций, равного 30, представлены в табл. 1. Для сравнения выбирались два варианта формирования характеристик публикации, выполняемого SMM-специалистом: между характеристиками публикаций с номерами 1-15 и между характеристиками публикаций с номерами 16-30. Из табл. 1 видно, что в первом случае применение эволюционного алгоритма распределения привело к улучшению значения целевой функции на 28,21%, а во втором случае – на 22,31%.

Таблица 1

	Формирование SMM- специалистом		Итерация генетического алгоритма				
	№ 1-15	№ 16-30	1	3	7	9	итоговый
<i>Fpubl</i>	9,286	9,734	9,338	9,764	10,467	11,479	11,906

В табл. 2 представлены показатели работы алгоритма формирования контент-плана из 20 вариантов наборов, состоящих из 20 публикаций. Для сравнения выбирались два варианта формирования контент-плана, выполняемого SMM-специалистом: между контент-планами с номерами 1-10 и между контент-планами с номерами 11-20. Из табл. 2 видно, что в первом случае применение эволюционного алгоритма распределения привело к улучшению значения целевой функции на 13,95%, а во втором случае – на 20,43%.

Таблица 2

	Формирование SMM- специалистом		Итерация генетического алгоритма				
	№ 1-10	№ 11-20	1	3	7	9	итоговый
<i>Fkp</i>	11,748	12,705	13,705	13,476	12,873	11,205	10,109

Для остальных задач формирования контент-плана были получены схожие результаты. В целом результаты экспериментальной проверки подтвердили эффективность применения разработанного эволюционного алгоритма для разработки контент-плана продвижения мобильных регистраторов сварочных работ по сравнению с распределением SMM-специалистом без использования технологий искусственного интеллекта.

Заключение

В статье представлен многоуровневый иерархический эволюционный алгоритм рационального выбора параметров системы управления продвижением в социальных сетях.

Предложенный алгоритм формирования контент-плана с использованием эволюционного моделирования с реализацией рационального выбора характеристик публикаций позволяет улучшить эффективность маркетинга за счёт автоматизации и персонализации контента.

Основные преимущества предложенного алгоритма:

- автоматизация процессов подготовки публикаций и принятия решений;
- повышение скорости и качества обработки данных;
- снижение трудозатрат и повышение эффективности управления продвижением в социальных сетях.

Список литературы

- [Баринова, 2020]. Баринова Н.В., Баринов В.Р. Трансформация экономического поведения потребителей в цифровом мире // Вестник РЭА им. Г.В. Плеханова. – 2020. – № 5(113).
- [Благов, 2020] Благов А.В., Рыцарев И.А. Анализ социальных сетей: учебное пособие. – Самара: Изд-во Самарского университета, 2020. – 104 с.
- [Будко и др., 2024] Будко С.А., Окрестина О.Р. Управление технологиями в сфере маркетинга // Вестник науки. – 2024. – № 11(80). – URL: <https://cyberleninka.ru/article/n/upravlenie-tehnologiyami-v-sfere-marketinga> (дата обращения: 22.07.2025).
- [Борисов, 2024] Борисов В.В., Шеробурко Е.Н., Мрочек Т.В. Планирование закупок в бюджетных организациях с использованием эволюционного моделирования // Интегрированные модели и мягкие вычисления в искусственном интеллекте: Сборник научных трудов XII Международной научно-практической конференции (ИММВ-2024, Коломна, 14-17 мая 2024 г.). В 2-х т. Т. 1. – Смоленск: Универсум, 2024. – С. 329-340.
- [Гладков, 2019] Гладков Л.А., Гладкова Н.В. Интегрированный подход к решению задач оптимизации на основе методов эволюционного проектирования и мультиагентных технологий // Информатика, вычислительная техника и инженерное образование. – 2019. – № 1(34). – С. 1-8.
- [Заровчатская, 2025] Заровчатская Е.В., Захарченков К.В., Подвесовский А.Г. Алгоритм интеллектуальной поддержки принятия решений при управлении распределением сварочных работ между исполнителями // Эргодизайн. – 2025. – № 2(28). – С. 117-125.
- [Захарченков, 2018] Захарченков К.В., Вайнилович Ю.В. Методика многоуровневого управления учебными IT-проектами // Энергетика, информатика, инновации – 2018 (инновационные технологии и оборудование в промышленности, управление инновациями, экономика и менеджмент, научные исследования в области физической культуры, спорта и общественных наук): Сб. трудов VIII междунар. науч.-техн. конф. – Смоленск: Универсум, 2018. – Т. 3. – С. 18-21.
- [Индекс, 2024] Индекс готовности приоритетных отраслей экономики Российской Федерации к внедрению искусственного интеллекта. Аналитический доклад. – М.: Национальный центр развития искусственного интеллекта при Правительстве Российской Федерации, 2024. – IV + 85 с.
- [Казыханов и др., 2024] Казыханов А., Давлетшин Р.Р., Кузнецова Е.В. Использование искусственного интеллекта в маркетинговой деятельности организаций [Электронный ресурс] // Экономика и бизнес: теория и практика. – 2024. – № 12-1(118). – URL: <https://cyberleninka.ru/article/n/ispolzovanie-iskusstvennogo-intellekta-v-marketingovoy-deyatelnosti-organizatsiy> (дата обращения: 22.07.2025).
- [Карпенко, 2017] Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учебное пособие. – 2-е изд. – М.: МГТУ им. Баумана, 2017. – 448 с.
- [Скоробогатых, 2025] Скоробогатых И.И., Сидорчук Р.Р., Андреева С.Н. Маркетинг: создание и донесение потребительской ценности: учебник. – М.: ИНФРА-М, 2025. – 589 с.
- [Смолина, 2024] Смолина В.А. SMM с нуля. Секреты продвижения в социальных сетях: практическое пособие. – М.; Вологда: Инфра-Инженерия, 2024. – 224 с.

УДК 004.827

doi: 10.15622/rcai.2025.036

СПОСОБ И ПРОГРАММНАЯ БИБЛИОТЕКА УЧЕТА СТРАТЕГИИ ОГРАНИЧЕНИЙ БАЗОВОГО ДИАПАЗОНА НЕЧЕТКИХ ЧИСЕЛ

М.М. Зернов (zmmioml@yandex.ru)^{A,B}

Н.А. Макеев (n.a.makeenkov@yandex.ru)^A

^A АО «Радиозавод», Пенза

^B Филиал Национального исследовательского университета «МЭИ»
в г. Смоленске, Смоленск

В работе предлагается способ учета стратегии ограничений базового диапазона нечетких чисел на основе функций высшего порядка, позволяющий упорядочить интерпретацию нечеткого числа с учетом ограничений. Способ нашел применение в разрабатываемой программной библиотеке нечетких вычислений на языке C++. Структура программных классов библиотеки построена с применением шаблонов проектирования «стратегия», CRTP, «посетитель». Использован дополнительный прием отсеечения элементов, выходящих за пределы носителя, при расчете функции принадлежности для упорядоченного массива входных чисел. Проведена экспериментальная оценка быстродействия, подтвердившая эффективность реализации способа.

Ключевые слова: функции высшего порядка, нечеткая математика, нечеткие числа.

Введение

Область определения нечеткого числа в ситуациях, когда оно определено не на всей числовой оси, а на ограниченном базовом диапазоне (интервале), редко учитывается при проведении арифметических операций. Можно задать ограничение на множество возможных пар элементов базового множества при определении операций по принципу расширения Заде, т.е. использовать совместное распределение возможностей [Fuller et al., 2004], [Carlsson et al., 2004]. Однако, гораздо эффективнее использовать совместные распределения возможностей для других целей, например, уменьшения размытия результата нечетких операций при итерацион-

ных вычислениях [Федулов, 2006]. Всевозможные способы выполнения нечетких арифметических операций на основе L-R чисел, как и большинство способов на основе α -уровневых множеств не предусматривают учета границ базового диапазона. Фактически, единственной областью, в которой такой контроль осуществляется, является нечеткий логический вывод, в котором, в принципе, границы области определения лингвистических переменных используются лишь при вычислении значений функций принадлежности. При этом, по сути, используется стратегия отсечения части нечеткой переменной (заданной нечетким числом) не попавшей в область определения. В работе [Зернов, 2008] было предложено преобразование интерпретации результата нечетких арифметических операций на ограниченном базовом диапазоне, использующее другую стратегию – агрегации посредством S-нормы. Смысл преобразования заключается в том, что, чем более значимая часть результата нечеткой арифметической операции выходит за границу базового диапазона, тем выше принадлежность соответствующей границы диапазона результату операции. Иначе говоря, утверждения “число X меньше a ” и “число X больше b ” лишены физического смысла и должны трактоваться как “Число X равно a ” и “Число X равно b ” соответственно.

Однако его применение можно расширить и на случай любого нечеткого числа, в т.ч. заданного параметрически. Пусть имеем нечеткое число \tilde{X} с функцией принадлежности μ , заданное на интервале действительных чисел $[a, b]$. Тогда скорректированное значение принадлежности μ' можно представить как:

$$\mu'(x) = \begin{cases} \max_{x \leq a} \mu(x), & x = a, \\ \mu(x), & x \in (a, b), \\ \max_{x \geq b} \mu(x), & x = b, \\ 0, & x \notin [a, b]. \end{cases} \quad (1.1)$$

Естественно, такое преобразование является лишь одним из вариантов интерпретации нечеткого числа в границах базового диапазона. Помимо этого, могут использоваться другие S-нормы, могут вовсе не накладываться ограничения. Наконец, может быть применено и простое отсечение:

$$\mu'(x) = \begin{cases} \mu(x), & x \in [a, b], \\ 0, & x \notin [a, b]. \end{cases} \quad (1.2)$$

Далее в работе будет предложен формализм способа учета стратегии ограничений и представлена программная библиотека на языке C++, в которой реализован предложенный способ.

1. Обзор библиотек нечетких вычислений

В настоящее время имеется большое количество программных библиотек в той или иной степени, ориентирующихся на флагманское решение – пакет Fuzzy Logic Toolbox для работы с системами нечеткого логического вывода [Alcal'a-Fdez et al., 2016]. Основной функционал построен вокруг трех областей – построения систем нечеткого логического вывода, нечеткой кластеризации, обучения систем нечеткого логического вывода на основе гибридных ANFIS-моделей [Jang, 1993]. Позднее к нему были добавлены возможности построения систем нечеткого логического вывода на основе нечетких множеств 2 типа [Liang et al., 2000]. Примечательно, что Fuzzy Logic Toolbox также предоставляет функцию *fuzarith*, позволяющую выполнять арифметические операции над нечеткими числами, заданными на одном и том же базовом диапазоне, с которым соотносится и результат. Вычисления осуществляются методом α -уровней [Dong et al., 1985]. Операнды задаются как массивы нечетких значений, соотносимых с одним универсальным множеством. К нему же относится и результат. Таким образом, реализуется стратегия ограничения результата операции путем отсека. Однако, работа с нечеткими числами является скорее побочной функцией и реализована не самым удобным образом, особенно с учетом не самых плохих возможностей объектно-ориентированного программирования в Matlab. В любом случае, решение является проприетарным и нишевым.

Другой заметной программной библиотекой, ориентированной на нечеткие вычисления и предоставляющей операции над нечеткими числами является *scikit-fuzzy* – пакет (набор библиотек) для языка Python, совместимая с другим известным пакетом *scikit-learn* [Warner et al., 2019], [Zhang et al., 2024]. Здесь вычисления выполняются без ограничений. При этом используется два метода – на основе принципа расширения Заде [Zadeh, 1975] и метод на основе α -уровней. В последнем случае, в отличие от реализации Fuzzy Logic Toolbox, операнды могут иметь разные базовые диапазоны, а базовый диапазон результата вычисляется на их основе.

Несмотря на развитую объектную модель в отношении систем нечеткого логического вывода, нечеткие числа, как таковые, отдельными классами не представлены.

По результату анализа распространенных программных библиотек для реализации систем нечеткого логического вывода и нечетких вычислений, авторами было принято решение о разработке собственной объектно-ориентированной библиотеки на языке C++ 17, позволяющей варьировать стратегии ограничения базового диапазона для разнообразных функций принадлежности нечетких чисел. Помимо этого библиотека должна позволять осуществлять арифметические операции над нечеткими числами с применением различных методов. Сами операции должны, по возможности, позволять естественную запись на основе математических операторов над объектами, представляющими нечеткие числа.

2. Способ учета стратегии ограничений базового диапазона нечетких чисел

Пусть имеется множество стратегий ограничений базового диапазона $S = \{s_0, s_1, \dots, s_k\}$, состоящее минимум из двух элементов – тривиальной стратегии s_0 “без ограничений” и хотя бы одной “ограничивающей”, например на основе агрегации через \max выпадающей части нечеткого числа. Каждая стратегия s_i представлена функционалом (функцией высшего порядка) $F_{s_i}(\mu, [a, b])$ от функции принадлежности и базового диапазона, возвращающим скорректированное значение функции принадлежности. Для стратегии “без ограничений” функционал тривиален:

$$F_{s_0}(\mu, [a, b])(x) = \mu(x).$$

Для ограничивающей стратегии на основе агрегации через \max – определяется в соответствии с выражением (1.1). Для ограничивающей стратегии на основе отсечения – в соответствии с выражением (1.2).

Таким образом, нечеткое число \tilde{X} с функцией принадлежности μ характеризуется тройкой:

$$\langle \mu, [a, b], s \rangle, [a, b] \subseteq \mathbb{R}, s \in S.$$

При этом, разумно наложить ограничение – вся числовая ось в качестве базового диапазона может быть использована только совместно с тривиальной стратегией:

$$[a, b] = [-\infty; +\infty] \Leftrightarrow s = s_0.$$

Иначе говоря, бессмысленно использовать “ограничивающую” стратегию, не зная базового диапазона, равно как и устанавливать базовый диапазон, не задавая нетривиальной стратегии.

Таким образом, способ учета стратегии ограничений базового диапазона нечетких чисел на основе функций высшего порядка выражается в специфике выполнения следующих действий.

1. Создание/инициализация нечетких чисел.

Вначале нужно определить нетривиальную стратегию ограничений, используемую для создания нового “ограниченного” числа. Затем при создании нового нечеткого числа, ответить на вопрос известно ли ограничение на базовый диапазон для него в текущем моменте/месте производимых вычислений.

Если ответ положительный – создаем нечеткое число как совокупность функции принадлежности, “ограничивающей” стратегии и нетривиального базового диапазона.

Иначе – как совокупность функции принадлежности, тривиальной стратегии “без ограничений” и тривиального базового диапазона.

2. Наложение ограничений на уже имеющееся нечеткое число.

Может производиться как над неограниченным числом, так и числом с нетривиальным базовым диапазоном. В обоих случаях следует создать новое нечеткое число в виде совокупности функции принадлежности исходного числа, выбранной в п.1 ограничивающей стратегии и задаваемого диапазона.

3. Вычисление значения функции принадлежности ограниченного числа в заданной точке.

Вычисление делегируется функции высшего порядка, представляющей стратегию. При этом, наивная реализация такой стратегии, как например, “агрегация через \max ” может быть вычислительно затратной. Вместо прямого использования выражения (1.1), предлагается в случае вычисления на границах диапазона сопоставлять взаимное расположение границы и супремума нечеткого числа.

Итак, пусть имеется функция $\sup(\tilde{X})$, возвращающая пару

$$\left(\alpha_{\sup}(\tilde{X}) = \sup_{x \in [-\infty, +\infty]} \mu(x), A_{\alpha_{\sup}}(\tilde{X}) = [xl_{\alpha_{\sup}}(\tilde{X}), xr_{\alpha_{\sup}}(\tilde{X})] \right)$$

из собственно супремума функции принадлежности (который может быть вычислен, например, из ее параметров) на всей числовой оси и соответствующего ему интервала α -уровня. Тогда выражение 1.1 может быть переписано в виде:

$$\mu(x) = \begin{cases} \alpha_{\sup}, x = a \wedge (xl_{\alpha_{\sup}}(\tilde{X}) \leq a), \\ \mu(x), x = a \wedge (xl_{\alpha_{\sup}}(\tilde{X}) > a), \\ \mu(x), x \in (a, b), \\ \mu(x)_{\sup}, x = b \wedge (xr_{\alpha_{\sup}}(\tilde{X}) < b), \\ \alpha_{\sup}, x = b \wedge (xr_{\alpha_{\sup}}(\tilde{X}) \geq b), \\ 0, x \notin [a, b]. \end{cases}$$

Строго говоря, $\alpha_{\sup} = 1$ в силу условия нормированности нечеткого числа, а сам соответствующий интервал α -уровня представляет собой моду (для унимодальных чисел) или интервал модальных значений (для толерантных чисел). Однако, это условие предлагается ослабить и применять предложенное преобразование для всех нечетких множеств, заданных выпуклыми функциями принадлежности.

Отдельно стоит рассмотреть вопрос применения ограничений к нечеткому числу, заданному гистограммой. В этом случае (при выполнении условия выпуклости функции принадлежности), предлагается использовать его аппроксимацию на основе множеств α -уровня.

Если же условие выпуклости функции принадлежности не выполняется, то речь идет уже о просто нечетком множестве, определенном на множестве действительных/целых чисел и для него в принципе целесообразно определить как набор поддерживаемых операций, так и способов их реализации. Предложенный способ и его программная реализация применимы к нечетким множествам с выпуклой функцией принадлежности.

3. Объектная модель библиотеки FuzzyLib++

В данном разделе представлена объектная модель разрабатываемой библиотеки нечетких вычислений на языке C++ – FuzzyLib++. В настоящий момент реализована только часть планируемого функционала библиотеки, а именно, управление стратегией ограничения нечетких чисел.

На рис. 1. представлена диаграмма классов библиотеки. Объектная модель использует три распространенных шаблона проектирования – “стратегия”, CRTP (Curiously Recursive Template Pattern) и “посетитель” [Алексадреску, 2008].

Полиморфизм реализован статически на основе CRTP-шаблона, в котором класс нечеткого множества *FuzzySet* выступает в качестве базы наследования, предоставляя общие методы нечетких множеств – получения типа множества, расчета значения функции принадлежности. Последний подразумевает несколько реализаций – как для передаваемого буфера результатов, так и вновь формируемого, а также для упорядоченных и неупорядоченных входных значений.

Промежуточным звеном иерархии наследования является класс нечеткого числа *FuzzyNumber*, использующий в качестве носителя – числа с плавающей точкой. Нечеткое число является точкой сборки стратегий – функции принадлежности *MF* и шаблонной стратегии ограничения *TLimiter*, смешивая их с функционалом нечеткого множества. Нечеткое число добавляет методы получения интервала носителя, модального интервала/интервала супремума, набора (вектора) α -уровней.

В соответствии с шаблоном CRTP, базовый класс в качестве шаблонного аргумента принимает класс наследника. Таким образом, класс наследника наследует неявно заданному абстрактному интерфейсу. В нашем случае, наследование от *FuzzySet* подразумевает следование интерфейсу *CFuzzySet* (конкретный тип нечеткого множества), определяющему наличие реализаций методов расчета значений функции принадлежности в точке и на массиве входных значений с размещением в готовом буфере. Наследование от шаблонного класса нечеткого числа более сложно. Наследниками являются стратегии ограничений, примененные к функциям принадлежности. Помимо методов конкретного нечеткого множества, они должны предоставлять методы функции принадлежности – частные реализации получения всех интересующих интервалов. Для того, чтобы ме-

58

тоды стратегий ограничения реализовывались с учетом результатов конкретных функций принадлежности – они наследуются от своих шаблонных параметров.

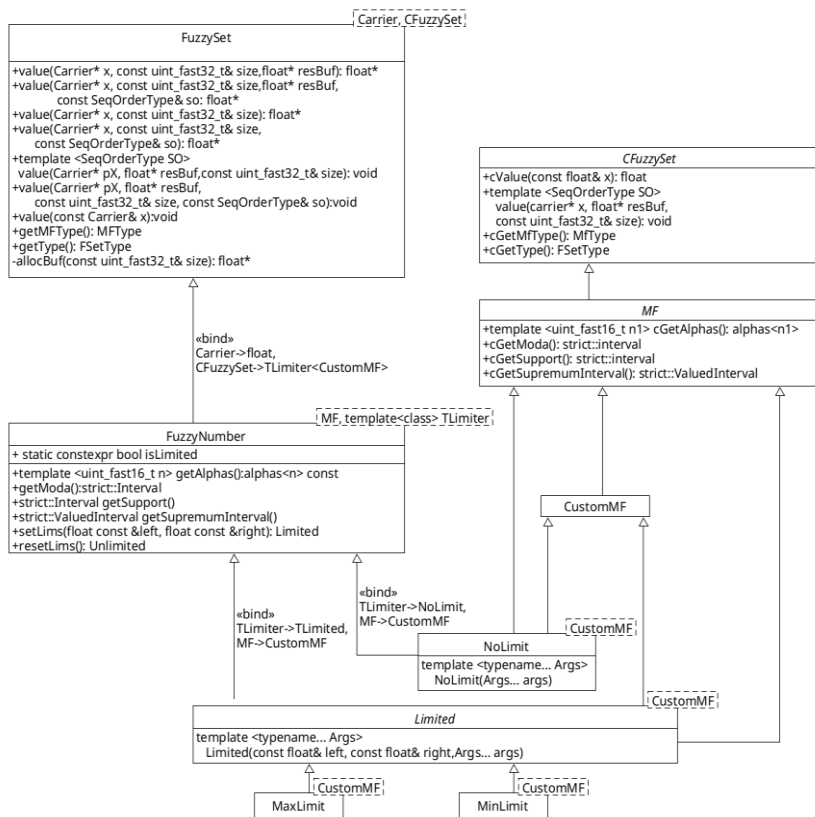


Рис. 1. Общая диаграмма классов библиотеки FuzzyLib++

На рис. 2 представлена диаграмма классов для результата подстановки (инстанцирования) в используемые шаблоны конкретного класса 2-сторонней гауссовой функции принадлежности. Стратегия отсечения в текущей версии не реализована, но подразумевается возможность ее внедрения.

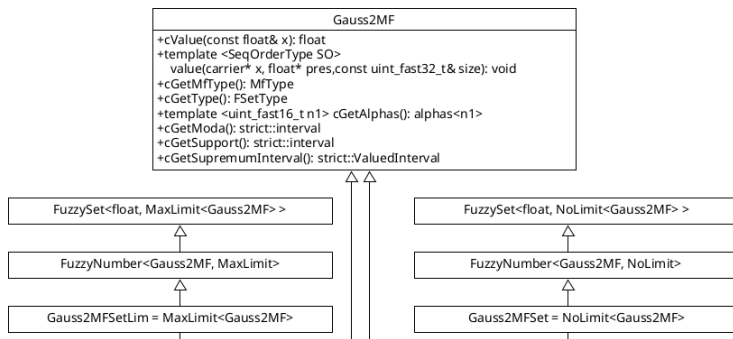


Рис. 2. Диаграмма классов, генерируемых для 2-сторонней гауссовой функции принадлежности

Слабой стороной шаблона проектирования CRTP является то, что для каждого конкретного наследника (например, неограниченного и ограниченного в соответствии со стратегией агрегации по максимуму нечеткого числа с 2-сторонней гауссовой функцией принадлежности), при подстановке строится своя иерархия наследования. Статический полиморфизм обеспечивает большее быстродействие, что особенно критично для встраиваемых систем по сравнению с динамическим (виртуальными методами, как способом реализации абстрактных интерфейсов). С другой стороны, к наследникам уже не обратиться через указатель на абстрактный интерфейс.

В результате нужно или явно использовать каждый такой конкретный класс нечеткого числа или создавать сигма-тип объединяющий наследников. Для последнего необходимо явное указание входящих в него вариантов и определение основных вызовов с использованием шаблона “посетитель”.

Рассмотренная модель реализована в рамках стандарта C++17, предоставляющего средства как для реализации сигма-типов (шаблонный класс `std::variant`), так и для их посещения (шаблонная функция времени компиляции `std::visit`).

4. Результаты оценки производительности вычисления значений функций принадлежности в библиотеке FuzzyLib++

Тестирование проводилось на вычислительной машине с процессором Intel(R) Xeon(R) CPU E5-2680 v4 с базовой частотой 2.40ГГц и максимальной 3.30 ГГц под управлением операционной системы Fedora Linux 41 Server с оперативной памятью 112 ГБайт DDR4 работающей на эффективной частоте 2400 МГц. В качестве компилятора был использован

гсс 13.3. Использован уровень оптимизации компилятора «-O2» и флаг оптимизации под архитектуру процессора «-march=native». Использовалась система сборки *cmake*.

Задавалось различное количество (101, 1001, 10001) входных действительных значений, которые в виде массива подавались на вход метода вычисления значения функции принадлежности нечеткого числа. С целью подсчета чистой производительности без учета аллокации буфера результата в куче, использовался метод класса нечеткого множества, размещающий рассчитанные значения в буфере результата, переданном в виде указателя.

В табл. 1 представлено сравнение времени вычисления нечеткого числа с гауссовой функцией принадлежности с параметрами $M = 5$, $\sigma = 1,67$ для 10001 значения на интервале $[-3,01354, 13,01345]$ без ограничения и с ограничением базового диапазона интервалом $[5,5, 20]$.

Таблица 1

Время вычисления 10001 значения функции принадлежности нечеткого числа гауссового типа

Количество значений 10001	Попадание в носитель, отн.ед.	t возр., нс	t неуп., нс
Неограниченное число	1	93182	92839
Ограниченное число	0,468	64409	140995
Отн. разница, %	-53,2	-30,9	51,9

Использовались два варианта метода вычисления значения функции принадлежности – с учетом порядка входных значений по возрастанию и без. Порядок имеет смысл только для реализации гауссового числа с ограничениями – отсекаются части входного массива, не попадающие в ограничения и носитель нечеткой функции принадлежности. Неограниченная реализация такой оптимизации пока не имеет. Доля входных значений, попавших в носитель нечеткого числа показана в первом столбце таблицы. В каждом случае указано среднее время по результатам 50 измерений.

Как видно, в общем, неупорядоченном, случае, время вычисления значений функции принадлежности в ограниченном варианте почти на 52% превышает время неограниченного варианта. Тем не менее, расчет результата с размещением в готовом буфере, даже с учетом накладных расходов на реализацию стратегии агрегации по максимуму, не превышает в среднем 0,2 мс. Такой результат позволяет надеяться на достаточное быстродействие в режиме реального времени и для встраиваемых систем. Последнее утверждение, конечно, в дальнейшем, предстоит проверить экс-

периментально – скомпилировав библиотеку для распространенных архитектур микроконтроллеров и проведя испытания на соответствующих стендах. Заметим, что оптимизация по пересечению с носителем позволяет частично компенсировать накладные расходы на применение стратегии ограничения.

Аналогичное сравнение проведено и для меньшего количества входных значений – 1001, результаты которого представлены в табл. 2.

Таблица 2

Время вычисления 1001 значения функции принадлежности нечеткого числа гауссового типа

Количество значений 1001	Попадание в носитель, отн.ед.	t возр., нс	t неуп., нс
Неограниченное число	1	9525	9428
Ограниченное число	0,468	7040	15854
Отн. разница, %	-53,2	-26,1	68,2

Как видно, с уменьшением числа рассчитываемых значений, при той же их пропорции между ограниченным и неограниченным случаем, доля накладных расходов на расчет ограничений растет. Тем не менее, общее время расчета не превышает 16 мкс.

В табл. 3. Представлены результаты измерения времени расчета 10001 значения функций принадлежности нечетких чисел с гауссовой, двусторонней гауссовой, треугольной функциями принадлежности, а также, функции принадлежности на основе 101 α -уровневого интервала (получены из двусторонней гауссовой функции принадлежности). Для последней, результат рассчитывается путем линейной интерполяции по границам двух ближайших интервалов α -уровня.

Ограниченные версии чисел рассчитывались с различной пропорцией попадания входных значений в носитель ограниченного числа. Для неограниченных чисел, практически, все значения попадали в носитель.

Таблица 3

Время вычисления 10001 значения функции принадлежности нечетких чисел различных типов

Тип ФП	Попад. в носит. огр., отн.ед.	Отн. разн, %	t неогр., нс	t огр. неуп., нс	Отн. разн, %	t огр. возр., нс	Отн. разн, %
GaussMF	0,468	-53,2	93182	140995	51,3	64409	-30,9
Gauss2MF	0,595	-40,5	94230	152769	62,1	70114	-25,6
TriMF	0,250	-75,0	491517	405236	-17,6	146047	-70,3
Alphas101	0,595	-40,5	2375444	866877	-63,5	1634639	-31,2

Как видно, наиболее сложной в смысле времени вычислений является функция принадлежности на основе α -уровней. Но даже для нее время расчета 10001 значения составляет порядка единиц мс.

Для ограниченных чисел на основе гауссовой и 2-сторонней гауссовой функций принадлежности в общем, неупорядоченном, случае, время вычисления значений функции принадлежности растет.

Упорядоченный вариант позволяет компенсировать часть накладных расходов, пусть и не в той же пропорции, в которой уменьшается доля входных значений, попадающих в носитель.

Для треугольной функции принадлежности и функции принадлежности на основе α -уровней, в силу их интервального характера, даже неупорядоченный вариант вычислений приводит к уменьшению времени вычислений при «выпадании» элементов носителя. Причем, для последней, «улучшенной» вариант, учитывающий порядок значений делает это в среднем хуже. Такие результаты говорят о необходимости дальнейшей оптимизации как процедуры отсечения «лишних значений» так и процедуры интерполяции.

Заключение

Представленная программная библиотека реализует пока только простейший функционал. Тем не менее, в ней уже заложены принципы, отличающие ее от других библиотек нечетких вычислений – представление нечеткого числа в виде объекта, возможность гибкого применения стратегий ограничения нечетких чисел, возможность восстановления исходной формы ограниченного нечеткого числа. Таким образом, реализуется предложенный способ учета стратегии ограничений базового диапазона нечетких чисел на основе функций высшего порядка.

В дальнейшем, планируется добавление в библиотеку нечетких арифметических и логических операций с возможностью задействовать различные комплекты реализации данных операций.

Предполагается публикация библиотеки под свободной лицензией.

Список литературы

- [Александреску, 2008] Александреску А. Современное проектирование на C++: пер. с англ. – М.: Издательский дом «Вильямс», 2008. – 336 с.
- [Зернов, 2008] Зернов М.М. Арифметические вычисления над нечеткими числами на ограниченном базовом диапазоне // Тез. докл. IX Междунар. конф. «Системы компьютерной математики и их приложения» (СКМП-2008). – Смоленск, 2008. – С. 42-44.
- [Федулов, 2006] Федулов А.С. Вид взаимодействия нечетких чисел, ограничивающий возрастание неопределенности при выполнении операций нечеткой арифметики // Вестник МЭИ. – 2006. – № 1. – С. 101-110.

- [Alcal'a-Fdez et al., 2016] Alcal'a-Fdez J., Alonso J.M. A Survey of Fuzzy Systems Software: Taxonomy, Current Research Trends, and Prospects // IEEE Transactions on Fuzzy Systems. Feb. 2016. – Vol. 24, No. 1. – P. 40-56 (статья в журнале на англ. языке).
- [Carlsson et al., 2004] Carlsson C., Fuller R., Majlender P. Addition of completely correlated fuzzy numbers // FUZZ – IEEE 2004. IEEE Int. Conf. on Fuzzy Systems. Budapest, Hungary, 2004.
- [Dong et al., 1985] Dong W., Shah H., Wong F., Fuzzy computations in risk and decision analysis // Civ Eng Syst. – 1985. – 2. – P. 201-208.
- [Fuller et al., 2004] Fuller R., Majlender P. On interactive fuzzy numbers // Fuzzy Sets and Systems. – 2004. – Vol. 143. – P. 355-369.
- [Jang, 1993] Jang J.-S. ANFIS: adaptive-network-based fuzzy inference system // IEEE Transactions on Systems, Man, and Cybernetics. – 1993. – Vol. 23, No. 3. – P. 665-685.
- [Liang et al., 2000] Liang, Q., & Mendel, J. M. Interval type-2 fuzzy logic systems: theory and design // IEEE Transactions on Fuzzy systems. – 2000. – Vol. 8, No. 5. – P. 535-550.
- [Warner et al., 2019] Warner J., Sexauer J., scikit-fuzzy, twmeggs, alexsavio, Unnikrishnan A., Castelaõ, G., Pontes, F.A., Uelwer, T., pd2f, laurazh, Batista, F., alexbuy, Broeck, W.V. den, Song, W., Badger, T.G., Pe'rez, R.A.M., 7002 Power, J.F., Mishra, H., Trullols, G.O., Ho'rteborn, A., 99991, 2019. JDWarner/scikit-fuzzy: Scikit-Fuzzy version 0.4.2. – <https://doi.org/10.5281/zenodo.3541386>.
- [Zadeh, 1975] Zadeh L.A. Concept of a linguistic variable and its application to approximate reasoning // I, II, III, Information Sciences. – 8(1975) 199-249, 301-357; 9(1975). 43-80.
- [Zhang et al., 2024] Zhang D., Chen T. Scikit-ANFIS: A Scikit-Learn Compatible Python Implementation for Adaptive Neuro-Fuzzy Inference System // International Journal of Fuzzy Systems. – 2024. – 26. – 10.1007/s40815-024-01697-0.

РЕАЛИЗАЦИЯ АЛГОРИТМА РАЗДЕЛЕНИЯ НЕЧЕТКОГО ГРАФА ДЛЯ ФОРМИРОВАНИЯ ЗОН ВЛИЯНИЯ ОБЪЕКТОВ СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМ

Е.Р. Зяблова (*zero32f@mail.ru*)

А.О. Ермаков (*alee@sfnedu.ru*)

Э.С. Шевкетова (*shevketova@sfnedu.ru*)

Южный федеральный университет, Таганрог

В работе рассматриваются особенности реализации модифицированного алгоритма разделения нечеткого графа с разнотипными вершинами и множественными разнотипными связями. Обосновано применение такого графа для моделирования сложных технических систем (СТС) с разнотипными информационными потоками. Преимуществом модифицированного алгоритма разделения является сохранение вычислительной сложности оригинального алгоритма с учетом возможности обработки нечеткого графа с разнотипными вершинами и множественными разнотипными связями. Отличительной особенностью реализованного алгоритма является возможность разделения графа на пропорциональные подграфы (по количеству вершин). Описаны подходы формирования зон влияния объектов системы с использованием алгоритма разделения и алгоритма вычисления метрик графа. В качестве примера СТС рассмотрена система охраны протяженного периметра, для которой сформированы зоны влияния агентов в зависимости от метрик подграфов.

Ключевые слова: нечеткий граф, разнотипные вершины, разнотипные множественные связи, разделение графа, система охраны, программная реализация.

Введение

В работе [Цветков, 2017] сложные технические системы (СТС) определены как «информационно-технические системы» и «системы, поведение которых сложно моделировать из-за сложных зависимостей между их частями или из-за сложных взаимодействий между данной системой и окружающей средой», и показано, что они «ближе относятся к области

«больших данных», «больших графов» и «больших сетей». В данной работе в качестве СТС рассматривается группа систем с разнотипными информационными потоками между объектами, например потоками видео- / аудиоинформации, управляющих сигналов и др.

Решению задачи моделирования СТС, в том числе для обеспечения безопасности объектов в системах охраны периметров большой протяженности посвящены множество работ [Особенности организаций систем безопасности объектов с большой протяженностью периметра, 2020], [Збирко, 2021], в том числе актуальной остается задача поиска эффективных подходов для моделирования взаимодействия объектов таких систем на основе графов [Лазарев, 2020] и их обобщений, например, гиперграфов, метаграфов [Меньших и др., 2021], и их программная реализация.

Так, например, в работе [Лазарев, 2020] для оценки эффективности комплекса технических средств системы охраны используется ориентированный граф, заданный множествами вершин и дуг (ориентированных связей). Здесь в качестве вершин представлены показатели интеллектуальных элементов (датчиков, линий связи и др.), а дуги характеризуют иерархические связи элементов комплекса технических средств.

В статье [Меньших и др., 2021] предложено использование ориентированного метаграфа для построения структурно-параметрической модели системы обеспечения безопасности. Здесь ориентированный метаграф задан конечными множествами вершин, метавершин, дуг и соответствий между дугами и вершинами и метавершинами. В качестве метавершин могут выступать множества объектов защиты, множества угроз безопасности, а дуги между метавершинами отражают различные типы взаимоотношений, например «множество объектов защиты находятся в конфликте с множеством угроз безопасности» и др. Структурно-параметрическая модель системы строится на основе метаграфа с учетом вычисленных весов вершин и дуг, соответствующих степеням защищенности и важности объектов. Целевой функцией является максимизация степени защищенности системы.

Подходы, изложенные в работах [Лазарев, 2020] и [Меньших и др., 2021], позволяют отобразить систему в виде графовой или метаграфовой модели, но при этом не используют возможности теории графов в части алгоритмических средств. К тому же в метаграфовой модели для представления отношений различного типа используются однотипные дуги с весовыми коэффициентами.

В то же время развитие теории графов привело к появлению моделей, учитывающих разнотипные связи в графах [Boccaletta et al., 2014]. При помощи таких моделей можно представить разнотипные отношения в системе.

Однако, графы, моделирующие СТС, могут быть достаточно большими, что влечет рост временных затрат на вычисления при помощи алгоритмов [Hamilton, 2020]. Эта проблема частично решена в моделях на основе нечетких графов с множественными и разнотипными связями [Еремеев и др., 2021].

1. Постановка задачи

В данной работе будем рассматривать сложную техническую систему на примере системы охраны протяженного периметра. Такая система содержит ряд разнотипных объектов: ОО – мобильные и/или стационарные объекты охраны (люди, здания, склады и др.), ОПО – опорные точки объектов охраны (вышки и т.д.); агенты – роботизированные устройства, ЛПП – лицо, принимающее решения (компьютер), а также ПН – потенциальные нарушители (люди, роботы и другие объекты, проникающие извне, которые представляют возможную угрозу для безопасности охраняемого периметра). Между объектами существуют разнотипные отношения, например «управлять движением», «удаленность объектов», «наблюдать» и др.

Зона влияния агента может распространяться на некоторые объекты системы, в том числе на объекты охраны, их опорные точки и потенциальных нарушителей. Информация от агентов может поступать на ЛПП через различные беспроводные каналы связи, например, через Wi-fi, Bluetooth или низкочастотный радиоканал, в зависимости от степени удаленности агентов и ЛПП.

К основным задачам ЛПП относятся следующие задачи:

- моделирование системы охраны при помощи подходящей модели, например нечеткого графа с множественными и разнотипными связями [Еремеев и др., 2021] или его модификации с учетом разнотипности вершин, как предложено в работе [Зяблова, 2024,б];
- обмен видеоинформационными и управляющими сигналами с агентами с целью формирования их зон влияния.

Для формирования зон влияния объектов системы предложено использование алгоритма поиска кратчайших путей и алгоритма разделения графа [Зяблова, 2024,в]. После разделения графа в полученных подграфах вычисляются метрики (радиус, диаметр) и сравниваются с метриками конкретных моделей агентов.

Графовая модель должна меняться в режиме реального времени в зависимости от текущего состояния объектов системы. Для реализации такой модели была разработана среда моделирования на основе нечеткого графа с множественными и разнотипными связями [Мунтян, 2021]. Данная среда включала следующие возможности:

- разработка графовой модели с учетом множественных и разнотипных связей вручную;

- использование модифицированного алгоритма поиска кратчайших путей Форда-Беллмана [Зяблова, 2024,в];
- использование модифицированного алгоритма разделения графа [Мунтян, 2023] на основе алгоритма свертки вершин.

Однако требования динамичности к среде моделирования потребовали обновления программного обеспечения (ПО). Коллективом разработчиков принято решение об использовании клиент-серверной архитектуры для разрабатываемого ПО и использовании языка разработки Python. В настоящее время разработана структура программной системы моделирования взаимодействия объектов СТС [Зяблова, 2024,а], определены ее функциональные возможности и реализован ряд модулей, включая модуль динамической визуализации графовой модели [Зяблова и др., 2024].

Принято решение об использовании модифицированной модели на основе нечетких графов с разнотипными вершинами и множественными разнотипными связями, подробно описанной в работе [Зяблова, 2024,б].

Данная работа посвящена программной реализации алгоритма разделения такого графа.

2. Описание нечеткого графа с разнотипными вершинами и множественными разнотипными связями

Формально такой граф задан в виде

$$G = (Gv, Ge), \quad (1)$$

где $Gv = \{gv_i \mid i = 1, 2, \dots, n\}$ – нечеткое конечное множество разнотипных вершин (gv – *graph vertex*), $Ge = \{ge_l \mid l = 1, 2, \dots, m\}$ – нечеткое конечное множество связей (ge – *graph edge*).

Множество $Gv = \{Gv(tpv_k)\}$ позволяет задать группы разнотипных вершин. К атрибутам вершины относятся ее идентификатор, вес и тип. Применительно к системе охраны, вершины графа G могут задавать различные группы объектов системы, например объекты охраны, опорные точки объектов охраны, агенты, ЛПП и потенциальные нарушители.

Множество связей графа G определяется следующим образом:

$$Ge = \{Ge(o), Ge(tp_k), Ge(v_e)\}, \quad (2)$$

где $Ge(o)$ – подмножество однотипных связей, $Ge(tp_k)$ – подмножество разнотипных связей и $Ge(v_e)$ – подмножество множественных связей в виде вектора, позволяющих объединить t разнотипных связей в виде кортежа $\bar{v} = \langle v_1, v_2, \dots, v_t \rangle$. Существующие отношения между объектами системы в графовой модели представлены разнотипными связями (отношения «управлять движением», «удаленность объектов», «наблюдать» и др.) и множественными связями в виде вектора, например отношение «передача информации по различным каналам», в том числе: v_1 – «Wi-fi», v_2 – «Bluetooth», v_3 – «низкочастотный радиоканал».

Весу связи в нечетком графе G может придаваться смысл меры достоверности или степени принадлежности нечеткому множеству. Степени принадлежности могут быть постоянными, либо вычисляемыми посредством фаззификации [Ярушкина, 2004]. В таком графе веса вершин и связей ($\eta_i \in [0,1]$ и $\mu_l \in [0,1]$) можно интерпретировать в зависимости от используемой смысловой нагрузки, например вес разнотипной связи, задающей отношение «удаленность объектов», соответствует степени удаленности объектов системы. Веса вершин определенного типа (например, задающих агентов системы), могут соответствовать значениям диаметров обзора агентов с учетом принятой шкалы.

Для оценки значений показателей весов вершин и связей графа (1) предложено вербальное описание показателей и соответствующий им диапазон численных значений весов вершин и связей графа: 0 – показатель отсутствует; [0,01; 0,40] – слабый показатель; [0,41; 0,70] – умеренный показатель; [0,71; 1,00] – сильный показатель.

Преимущества данной модели отмечаются в работе [Зяблова, 2024,б] и заключаются в возможности представить разнотипные элементы системы и их отношения и одновременно снизить время анализа данных в системах с разнотипными информационными потоками за счет использования множественных связей в виде вектора, которые позволяют объединить ряд разнотипных связей. Например, в работе [Зяблова, 2024,а] показано уменьшение временных затрат на формирование матрицы расстояний для графов размерностью 300 – 1000 вершин не менее, чем в 1,5 раза по сравнению с известными моделями [Boccaletta et al., 2014], допускающими только разнотипные связи.

Это подтверждает эффективность и предпочтительность использования на практике нечетких графов с разнотипными вершинами и множественными разнотипными связями для моделирования взаимодействия объектов сложных систем.

3. Реализация модифицированного алгоритма пропорционального разделения нечеткого графа с разнотипными вершинами и множественными разнотипными связями

Модифицированный алгоритм пропорционального разделения нечеткого графа с разнотипными вершинами и множественными разнотипными связями (далее – алгоритм разделения графа) представлен и подробно описан в работе [Мунтян, 2023], где отмечены его преимущества. Данный алгоритм позволяет обрабатывать нечеткий граф с разнотипными вершинами и множественными разнотипными связями, сохраняет вычислительную сложность оригинального алгоритма и позволяет уменьшить время разделения графа по сравнению с алгоритмами, использующими матрицы

смежности или инцидентности и последовательный перебор вершин в зависимости от полноты графа до 20% [Сергеев и др., 2018]. Модуль, реализующий данный алгоритм, является составной частью программной системы моделирования [Зяблова, 2024,а]. Данный алгоритм позволяет разделить граф на подграфы с заданными пропорциями подмножеств вершин.

Хранение графовой структуры осуществляется в формате JSON. Связи графа определяются парами идентификаторов начальной и конечной вершин с указанием веса. Такой подход к структурированию данных обеспечивает простоту интеграции алгоритма с различными информационными системами и специализированными графовыми базами данных, включая, Neo4j [Еремеев и др., 2022] и др.

Основная логика алгоритма реализована в классе ZonePartitioner. Метод partition отвечает за процесс распределения вершин по подграфам, начиная с выбранных исходных вершин и итерационно присоединяя соседние элементы с учётом установленных соотношений.

Алгоритм разделения графа строится на последовательности ключевых этапов:

- 1) **фильтрация входных данных** (на этом этапе осуществляется отбор вершин и рёбер по заданным критериям, например, по типу вершины и/или связи, по значению веса вершины и/или связи. Это позволяет убрать из рассмотрения несущественные элементы и сфокусировать вычисления на релевантных компонентах графа);
- 2) определения количества частей разделения на подграфы с помощью ввода в консоль необходимых значений;
- 3) выбор исходных вершин (для формирования подграфов определяется набор исходных вершин, которые могут находиться как в центре, так и на периферии графа. В прикладных задачах, таких как охрана периметра, в качестве исходных вершин графа могут выступать агенты системы;
- 4) возможность выбора пропорций разделения графа на подграфы: равные или пропорциональные части (подмножества вершин);
- 5) **параллельное выполнение процедуры свёртки** (основная суть алгоритма заключается в итеративном присоединении соседних вершин, начиная с выбранных исходных вершин. Процесс проводится одновременно для всех подграфов, что обеспечивает равномерное и эффективное распределение вершин между выделенными подграфами. Важной особенностью является независимость метода от типа связей).

Предварительно представляется графовая модель системы охраны, в виде графа $G_1 = (Gv_1, Ge_1)$. Множество вершин $Gv_1 = \{gv_i \mid i = 1, 2, \dots, 20\}$. Соответствие вершин графа G_1 объектам системы отражено в табл. 1.

Здесь веса вершин типов tpv_1 и tpv_2 назначаются экспертом и соответствуют степеням значимости объектов, а веса вершин типа tpv_3 определяются посредством фаззификации в соответствии с техническими характеристиками агентов.

Таблица 1

Соответствие вершин графа G_1 объектам системы

Вершины графа	Типы вершин	Веса вершин	Объекты системы
gv_1-gv_{16}	tpv_1	по 0,5	объекты охраны $ОО_1 - ОО_{16}$
gv_{17}	tpv_2	1,0	ЛПР
$gv_{18}-gv_{21}$	tpv_3	по 0,9	агенты $A_1 - A_4$

Примем шкалу соответствий реального расстояния в системе и веса вершины и/или связи графа. Будем полагать, что расстояние в 2 км соответствует весовому коэффициенту с числовым значением «1». В соответствии с принятой шкалой выбраны агенты $A_1 - A_4$ с диаметром обзора 1,8 км (веса вершин $gv_{18}-gv_{21}$ принимают значение «0,9»).

Множество Ge_1 содержит множественные и разнотипные связи:

- связи ge_1-ge_{20} (тип tp_1), которые соответствуют отношению «удаленность», а веса этих связей – степени удаленности объектов друг от друга;
- связи $ge_{21}-ge_{40}$ (тип tp_2) соответствуют отношению «наблюдать»;
- множественные связи в виде вектора $ge_{41}-ge_{44}$ представляют отношение «Передача информации по каналам». Информация может передаваться по одному из трех каналов в зависимости от степени удаленности агентов и ЛПР: Wi-Fi; Bluetooth; низкочастотный радиоканал.

На этапе 1 выполняется фильтрация множеств вершин и связей графа по их типу. Для данного графа будут оставлены связи типа tp_1 и вершины типа tpv_1 . В результате фильтрации, сформирован граф G_2 (рис. 1).

На этапе 2 определяется количество частей графа, которое зависит от количества агентов в системе. С учетом количества агентов графовая модель (рис.1) будет разделена на 4 части.

На этапе 3 происходит формирование подмножества исходных вершин графа G_2 : $OrVertex = \{gv_3, gv_7, gv_{10}, gv_{12}\}$.

На этапе 4 осуществляется выбор пропорций для деления на подграфы, пусть для описанной выше модели графа пропорции для каждой зоны будут равны 0,25.

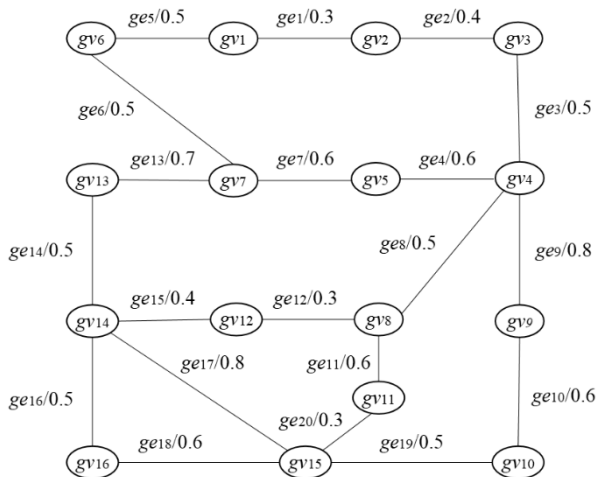


Рис. 1. Изображение графа G_2

На этапе 5 реализуется процедура свёртки и формирование подграфов $G_{2.1} - G_{2.4}$, где $G_{v_{2.1}} = \{gv_1, gv_2, gv_3, gv_4\}$, $G_{v_{2.2}} = \{gv_5, gv_6, gv_7, gv_{13}\}$, $G_{v_{2.3}} = \{gv_9, gv_{10}, gv_{15}, gv_{16}\}$, $G_{v_{2.4}} = \{gv_8, gv_{11}, gv_{12}, gv_{14}\}$.

На этом работа алгоритма разделения графа завершается. На рис. 2 приведены результаты работы алгоритма, здесь подграфы выделены разными цветами.

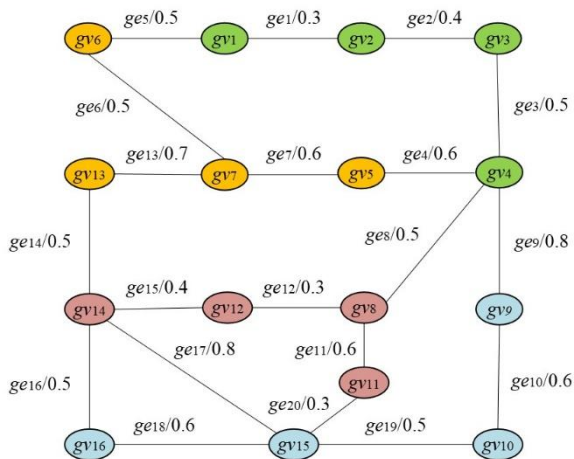


Рис. 2. Результаты работы алгоритма разделения графа G_2 на части

Сформированы зоны влияния объектов системы охраны: $A_1 = \{OO_1, OO_2, OO_3, OO_4\}$; $A_2 = \{OO_5, OO_6, OO_7, OO_{13}\}$; $A_3 = \{OO_9, OO_{10}, OO_{15}, OO_{16}\}$; $A_4 = \{OO_8, OO_{11}, OO_{12}, OO_{14}\}$.

Для каждого подграфа при помощи модифицированного алгоритма поиска кратчайших путей [Зяблова, 2024,в] вычисляются радиус r и диаметр d . Полученные значения ($d_{2,1} = 0.6$, $r_{2,1} = 0.35$; $d_{2,2} = 0.65$, $r_{2,2} = 0.35$; $d_{2,3} = 0.85$, $r_{2,3} = 0.55$; $d_{2,4} = 0.65$, $r_{2,4} = 0.35$) позволяют представить метрики зон влияния агентов $A_1 - A_4$.

Полученные параметры служат основой для выбора моделей агентов с заданными техническими характеристиками, что обеспечивает надёжное и эффективное видеонаблюдение за областью охраняемого периметра. Так, для обеспечения видеонаблюдения за областью охраняемого периметра достаточно четыре агента с диаметром обзора не менее 1,7 км.

Заключение

В работе рассмотрены особенности и пример применения модифицированного алгоритма пропорционального разделения нечеткого графа с разнотипными вершинами и множественными разнотипными связями для формирования зон влияния объектов СТС. Данный алгоритм позволяет обрабатывать нечеткий граф с разнотипными вершинами и множественными разнотипными связями и сохраняет вычислительную сложность оригинального алгоритма. Особенностью алгоритма разделения является возможность фильтрации вершин и связей графа, а также разделения графа на подграфы в заданных пропорциях по подмножествам вершин.

Список литературы

- [Особенности организаций систем безопасности объектов с большой протяженностью периметра, 2020]. Гречаный С.А., Герасименко Е.С., Романов М.С., Таравков М.В. Особенности организаций систем безопасности объектов с большой протяженностью периметра // Охрана, безопасность, связь. – 2020. – № 5-1. – С. 54-58.
- [Еремеев и др., 2021] Еремеев А.П., Мунтян Е.Р. Разработка онтологии на основе графов с множественными и разнотипными связями // Искусственный интеллект и принятие решений. – 2021. – № 3. – С. 3-18. – DOI 10.14357/20718594210301.
- [Еремеев и др., 2022] Еремеев А.П., Панявин Н.А. Унификация модели представления данных и преобразование форматов на основе нереляционной СУБД Neo4j // Программные продукты и системы. – 2022. – Т. 35, № 4. – С. 549-556. – DOI: 10.15827/0236-235X.140.549-556.
- [Збирко, 2021] Збирко А.И. Исследование возможностей беспилотных летательных аппаратов при охране объектов большой протяженности // Современные проблемы радиоэлектроники и телекоммуникаций. – 2021. – № 4. – С. 32.
- [Зяблова, 2024,а] Зяблова Е.Р. Программная система моделирования взаимодействия объектов системы охраны // Труды Международного научно-технического конгресса «Интеллектуальные системы и информационные технологии «IS&IT'24». Т. 2. – Таганрог: Изд-во Ступина С.А., 2024. – С. 355-359.

- [Зяблова, 2024,б] Зяблова Е.Р. Использование графовых подходов для решения задачи прогнозирования в области формирования карьерных траекторий // Труды XII Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте». В 2-х т. Т. 1. – Смоленск: Универсум, 2024. – С. 150-158.
- [Зяблова, 2024,в] Зяблова Е.Р. Модификация алгоритма поиска кратчайших путей ГН-графа для анализа сложных технических систем // Программные продукты и системы. – 2024. – Т. 37, № 3. – С. 354-363. – DOI: 10.15827/0236-235X.147.354-363.
- [Зяблова и др., 2024] Зяблова Е.Р., Боваев А.Б., Литвиненко В.Н. Особенности разработки программного модуля динамической визуализации графовой модели // Системный синтез и прикладная синергетика: Сб. науч. тр. XII Всерос. научн. конф. – Ростов-на-Дону; Таганрог: Изд-во ЮФУ, 2024. – С. 193-196.
- [Лазарев, 2020] Лазарев И.В. Применение графовой модели для оценки эффективности комплекса технических средств систем охраны // Охрана, безопасность, связь. – 2020. – № 5-2. – С. 151-153.
- [Меньших и др., 2021] Меньших В.В., Чиркова Н.Е. Построение структурно-параметрической модели гетерогенной системы обеспечения безопасности с использованием теории метаграфов // Вестник Воронежского института МВД России. – 2021. – № 1. – С. 54-61.
- [Мунтян, 2021] Мунтян Е.Р. Использование нечетких ГН-моделей для представления сложных технических систем // Информатизация и связь. – 2021. – № 3. – С. 55-60. – DOI:1034219/2078-8320-2021-12-3-55-60.
- [Мунтян, 2023] Мунтян Е.Р. Разработка алгоритма пропорционального разделения ГН-графа для формирования зон влияния объектов в сложных технических системах // Программные продукты и системы. – 2023. – Т. 36, № 3. – С. 378-387. – DOI: 10.15827/0236-235X.143.378-387.
- [Цветков, 2017] Цветков В.Я. Сложные технические системы // Образовательные ресурсы и технологии. – 2017. – № 3(20). – С. 86-91.
- [Сергеев и др., 2018] Сергеев Н.Е., Мунтян Е.Р. Применение алгоритма свертки для разделения графа на пропорциональные подграфы // Вестник УГАТУ. – 2018. – Т. 22, № 1(79). – С. 121-130.
- [Ярушкина, 2004] Ярушкина Н.Г. Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004. – 320 с.
- [Boccalettia et al., 2014] Boccalettia S., Bianconic G., Criadod R., del Geniof C.I., Gómez-Gardeñesi J., Romanced M., Sendiña-Nadalj I., Wangk Z., Zaninm M. The structure and dynamics of multilayer networks. Preprint submitted to Physics Reports, 2014. – 157 p.
- [Hamilton, 2020] Hamilton W.L. Graph Representation Learning // Synthesis Lectures on Artificial Intelligence and Machine Learning. – 2020. – Vol. 14, No. 3. – P. 1-159.
- [Eremeev, 2022] Eremeev A.P., Muntyan E.R. Developing an ontology on the basis of graphs with multiple and heterotypic connections // Scientific and Technical Information Processing. – 2022. – Vol. 49, No. 6. – P. 427-438.

ПОДХОДЫ К УЧЕТУ НЕОПРЕДЕЛЕННОСТИ ЗНАЧЕНИЙ ПАРАМЕТРОВ НЕЧЕТКИХ КОГНИТИВНЫХ КАРТ

Р.А. Исаев (*Ruslan-Isaev-32@yandex.ru*)^{A,B}

К.И. Рыченкова (*k.ri4enkova@yandex.ru*)^B

И.А. Копелиович (*igor.copeliiovitch@yandex.ru*)^B

А.Г. Подвесовский (*a.podvesovskiy@gmail.com*)^{A,B}

^A Институт проблем управления им. В.А. Трапезникова РАН, Москва

^B Брянский государственный технический университет, Брянск

В работе описывается проблема, связанная с недостаточной обоснованностью значений параметров нечетких когнитивных моделей на примере нечетких когнитивных карт В.Б. Силова. Рассматриваются причины существования этой проблемы, и на примере демонстрируется, что при использовании когнитивной модели для выработки стратегий управления отсутствие уверенности в значениях параметров может приводить к принятию ненадежных решений. Описываются недостатки применяемых подходов к снижению влияния неопределенности значений параметров когнитивных моделей на результаты моделирования, и предлагаются новые пути решения обозначенной проблемы. Первый из них предполагает модернизацию математического аппарата нечетких когнитивных карт Силова, в рамках которой неуверенность в значениях параметров интерпретируется как субъективная вероятность этих значений, при этом параметры представляются в форме случайных величин. Вторым подход основан на проведении анализа устойчивости параметров модели, под которым понимается исследование зависимости результатов моделирования от возможных ошибок в значениях параметров. Представлены ключевые направления исследований для разработки каждого из предложенных подходов.

Ключевые слова: когнитивное моделирование, нечеткая когнитивная карта, параметрическая идентификация, неопределенность параметров, анализ устойчивости, случайная величина.

Введение

Одним из подходов, применяемых для разработки, анализа и обоснования решений в слабоструктурированных системах, является когнитивный подход. Этот подход ориентирован на структуризацию знаний о слабоструктурированной системе, для понимания основных процессов, протекающих в ней. При этом множество процессов описываются в виде модели экспертных знаний о законах и закономерностях функционирования системы, с преобладанием субъективных оценок и лингвистических значений. В рамках когнитивного подхода способом описания знаний об исследуемой системе и происходящих в ней процессах является когнитивная модель, которая допускает формальное представление в виде когнитивной карты. Когнитивной картой называется причинно-следственная сеть, которая отражает субъективное представление исследователя о системе (индивидуальное или коллективное) в виде множества семантических категорий, называемых факторами или концептами, и множества причинно-следственных связей между ними.

Когнитивные модели не могут использоваться для моделирования и прогнозирования точных количественных характеристик функционирования сложных систем, однако могут служить хорошим инструментом их разведочного анализа [Захарова и др., 2020]. В то же время, принципиально приближенный характер когнитивной модели не отменяет того факта, что результаты моделирования (в том числе выводы качественного характера) определяются структурой и параметрами модели и, таким образом, находятся под влиянием присущей ей неопределенности.

В настоящей работе исследуется проблема, связанная с недостаточной обоснованностью значений параметров когнитивных моделей на примере нечетких когнитивных карт (НKK), предложенных В.Б. Силовым [Силов, 1995].

1. Проблема недостаточной обоснованности значений параметров нечетких когнитивных моделей

В процессе построения когнитивных моделей реальных систем на основе НKK Силова неизбежно возникают неопределенности, которые могут иметь следующие формы:

- неопределенность самого множества концептов НKK (под которой понимается его возможная неполнота или избыточность, наличие в его составе неудачно концептуализированных понятий реальной системы);
- неопределенность множества влияний между концептами (возможное отсутствие необходимых влияний, наличие неадекватных или избыточных влияний);

- неопределенность значений сил влияний между концептами, а также значений прочих параметров НКК (возникающая, в первую очередь, ввиду возможности совершения ошибок и допущения неточностей в процессе оценки значений параметров).

Другими словами, можно говорить о существовании таких форм неопределенности когнитивной модели, как неопределенность ее структуры и неопределенность значений ее параметров. Далее в настоящей работе рассматривается только неопределенность значений параметров НКК, на примере сил влияний между концептами – отметим, что именно эта разновидность параметров модели является наиболее значимой с точки зрения процесса моделирования в целом.

Возникновение ошибок в значениях сил влияний между концептами при построении НКК становится возможным ввиду действия многих факторов, среди которых можно выделить следующие:

- сложность формализации представлений экспертов о предметной области в терминах когнитивной модели и в рамках, задаваемых математическим аппаратом когнитивного моделирования;
- возможный недостаточный уровень компетентности экспертов в тех или иных аспектах сложных систем, вероятность чего повышается при построении когнитивных моделей междисциплинарных систем;
- особенности используемых методов параметрической идентификации [Подвесовский и др., 2019], накладывающие ограничения на точность получаемых значений параметров модели;
- возможность совершения экспертом случайных ошибок при использовании методов параметрической идентификации, в том числе ввиду того, что эти методы допускают некоторое нарушение согласованности экспертных суждений [Yager, 1982];
- несовершенство статистических данных, используемых для построения когнитивной модели с применением статистических методов идентификации [Подвесовский и др., 2019].

Наличие ошибок и неточностей в значениях параметров когнитивной модели обычно приводит к недостоверным результатам моделирования, что в конечном итоге может привести к принятию неэффективных и даже ошибочных управленческих решений.

Продemonстрируем, на примере абстрактной нечеткой когнитивной модели (рис. 1), ситуацию, в которой ошибки, допущенные при оценке значений сил влияний между концептами, способны привести к противоречию с точки зрения результатов моделирования. Представленная модель характеризуется силами влияний, указанными на соответствующих дугах когнитивного графа. Визуальное представление модели построено в разработанной с участием авторов СППР «ИГЛА» [Захарова и др., 2020], в этой же системе получены приведенные далее результаты моделирования.

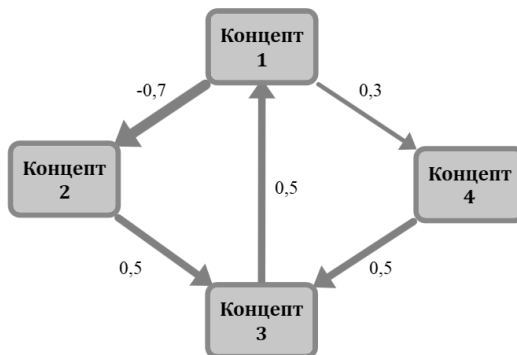


Рис. 1. Абстрактная нечеткая когнитивная модель для демонстрации влияния неопределенности значений параметров на результаты моделирования

Проведем сценарный анализ этой модели, исходя из следующих условий:

- управляемым концептом является «Концепт 1», при этом будут рассмотрены 7 сценариев управления (альтернатив), состоящих в установке различных значений данного концепта;
- целевым концептом является «Концепт 3», при этом целью управления является увеличение значения данного концепта в сравнении с текущим.

В результате моделирования возможных сценариев управления получим динамику значений целевого «Концепта 3» (рис. 2, слева), из которой видно, что оптимальный сценарий описывается Альтернативой 1. Данная альтернатива в качестве управляющего воздействия подразумевает уменьшение значения «Концепта 1» до предельно низкого уровня.

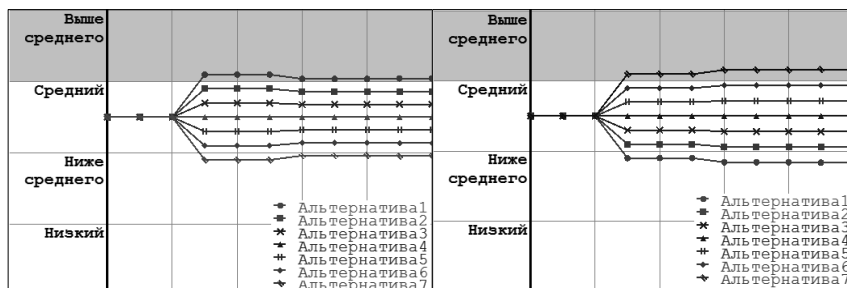


Рис. 2. Результаты сценарного анализа двух вариантов когнитивной модели: демонстрация смены порядка предпочтений между альтернативами

Предположим теперь, что при оценке сил влияний «Концепта 1» на зависящие от него «Концепт 2» и «Концепт 4» были допущены ошибки. Смоделируем это предположение, назначив указанным влияниям измененные значения: соответственно -0,3 и 0,7 (таким образом, знаки влияний остаются теми же, а их интенсивности меняются). Проведем повторный сценарный анализ модели, исходя из прежних условий. Получим (рис. 2, справа), что теперь оптимальный сценарий управления описывается Альтернативой 7, подразумевающей увеличение значения «Концепта 1» до предельно высокого уровня. Кроме того, заметим, что порядок предпочтений на множестве всех рассмотренных альтернатив изменился на противоположный.

Таким образом, допущение при построении когнитивной модели ошибок даже на уровне значений сил влияний (в том числе, при верно заданных знаках влияний) способно привести к искаженным результатам анализа модели, «рекомендующим» выбор такого управляющего воздействия на систему, которое на практике окажется неэффективным.

2. Анализ путей снижения влияния неопределенности значений параметров на результаты моделирования

В практике когнитивного моделирования сложился ряд способов противодействия рассмотренному выше типу неопределенности, имеющему место в когнитивных моделях.

Так, одним из важных этапов когнитивного моделирования является верификация построенной НКК. Этот этап направлен на выявление возможного несоответствия имеющейся когнитивной модели самой моделируемой системе, в контексте всех рассмотренных выше форм неопределенности. Важным аспектом эффективно организованного процесса верификации является применение подходов, направленных на активизацию и использование когнитивных способностей человека, для повышения вероятности обнаружения допущенных при построении модели ошибок. Наиболее естественным из них является подход, связанный с визуализацией НКК и ее визуальным анализом. Так, в работе [Исаев и др., 2020] представлены метафоры визуализации когнитивных карт, которые могут применяться для визуального обнаружения избыточности или неполноты множества влияний в составе когнитивной модели.

В то же время, эффективность верификации как средства противодействия неопределенности значений параметров остается невысокой. Это обусловлено акцентом существующих подходов к верификации на выявление и исправление ошибок в структурных (а не параметрических) аспектах модели. Что касается возможностей поиска ошибок в значениях сил влияний, то верификация может поспособствовать обнаружению

влияний с неверно заданными знаками — однако, как уже демонстрировалось, даже при верных знаках влияний, неточности в их значениях могут стать причиной ненадежных результатов моделирования.

Другим важным направлением работы в области повышения точности и достоверности оценки параметров НКК является совершенствование самих методов идентификации модели. В случае экспертных методов оно может быть направлено на повышение «разрешающей способности» метода (т.е. разнообразия различных значений сил влияний, которые эксперт может обоснованно задать с применением данного метода [Подвесовский и др., 2019]), а также на усовершенствование механизмов оценки согласованности выносимых экспертом суждений [Исаев и др., 2017]. Совершенствование статистических методов может затрагивать вопросы учета низкого качества используемых данных, устранения явления мультиколлинеарности [Подвесовский и др., 2019] и т.д.

Тем не менее, развитие методов параметрической идентификации НКК не позволит полностью устранить присутствие неопределенности в значениях параметров. Так, экспертные методы не способны обеспечить идеальную согласованность суждений эксперта, в то время как любой факт нарушения согласованности уже вносит неопределенность в результаты обработки суждений. Проблема статистических методов состоит в несовершенстве количественных данных о моделируемой системе, которое всегда имеет место на практике (ввиду чего получаемые на основе этих данных величины, такие как регрессионные коэффициенты, являются лишь оценками искомых параметров, а не их достоверными значениями).

Ввиду невозможности обеспечить отсутствие неопределенности значений параметров НКК, на практике прибегают к попыткам снизить влияние этого фактора на результаты анализа когнитивной модели. Наиболее интуитивным способом такого снижения является проведение повторного моделирования с несколько измененными условиями. В частности, при наличии сомнений в корректности определенного аспекта структуры НКК или ее параметров, в модель вносят изменение, касающееся данного аспекта (тем самым моделируя предполагаемую ошибку в нем). Сравнение результатов анализа скорректированной таким образом модели с результатами анализа ее исходного варианта позволяет оценить степень значимости их различий — и, следовательно, степень влияния смоделированной ошибки на получаемые результаты. Далее полученную информацию можно использовать, например, для более тщательной верификации конкретных аспектов когнитивной модели или оценки надежности различных стратегий управления.

Проблема такого подхода заключается в его интуитивном характере и несистемности процесса его реализации, а также в отсутствии адекватных механизмов автоматизации этого процесса в средствах программной под-

держки когнитивных моделей. Следствием этого обычно является моделирование лишь небольшой части возможных ошибок, неэффективный учет либо отсутствие учета различных вероятностей наличия ошибок в разных элементах НКК, затрудненный (ввиду отсутствия специальных средств поддержки) анализ результатов и т.д. В целом, фактор несистемности применения описанного подхода не позволяет раскрыть его потенциально высокую эффективность. Далее в работе будут представлены авторские предложения, связанные с постановкой этого подхода на системную основу.

Наконец, необходимо упомянуть о существовании ряда других типов когнитивных моделей (помимо НКК Силова), особенности математического аппарата которых потенциально могут способствовать решению либо снижению значимости исследуемой проблемы. Так, предложены типы когнитивных карт, параметры которых не являются конкретными числовыми значениями. К таким можно отнести нечеткие продукционные когнитивные карты (НПКК) [Борисов и др., 2012], сочетающие преимущества продукционных правил и нечеткой логики. Другой подобный тип моделей – нечеткие реляционные когнитивные карты (НРКК) – представляют собой расширение классических НКК за счет реляционного представления нечетких соотношений [Борисов и др., 2012]. Позже были предложены темпоральные когнитивные модели, среди которых выделяются нечеткие реляционные [Borisov et al., 2024], мягкие реляционные [Борисов, 2024] и нейро-нечеткие [Borisov et al., 2022]. Однако построение и анализ упомянутых типов когнитивных моделей является более сложным, в частности, требуется задание функций принадлежности для лингвистических переменных и продукционных правил. При этом исследуемая проблема отчасти сохраняется ввиду неизбежной недостаточной обоснованности параметров уже самих функций принадлежности и правил.

Итак, проведенный обзор показал недостаточность существующих путей снижения влияния неопределенности значений параметров НКК на результаты когнитивного моделирования. В этой связи далее в работе представлены два альтернативных взгляда на организацию данного процесса на системной основе.

3. Когнитивные модели с представлением параметров как случайных величин

Предлагается рассмотреть новый тип когнитивных моделей, являющийся обобщением НКК Силова, в котором все параметры модели представлены как случайные величины (СВ), способные принимать значения из соответствующих допустимых диапазонов. При этом вероятность конкретного значения параметра модели следует интерпретировать как степень уверенности в том, что параметр принимает именно это значение.

Такой подход к интерпретации вероятности характерен также для байесовской теории. Отметим, что в ряде работ (например, [Aguilar, 2003], [Cai et al., 2010]) предлагались разновидности когнитивных моделей, основанных на НКК Коско [Kosko, 1986], аппарат которых также включает в себя элементы случайности – однако, используемые в других аспектах (например, случайное срабатывание влияния между концептами).

Идентификацию параметров таких НКК предлагается начинать с построения непрерывных СВ, описывающих силы влияний между концептами. Интуитивным решением является выбор нормального распределения, что обосновано центральной предельной теоремой, учитывающей суммарное воздействие множества факторов в сложных системах. Для построения непрерывной СВ с нормальным распределением необходимо задать математическое ожидание (μ) и среднеквадратическое отклонение (σ). Математическое ожидание силы влияния может быть определено с помощью любого из методов, используемых при задании параметров НКК Силова. Стандартное отклонение, в свою очередь, должно отражать степень нарушения согласованности экспертных оценок или уровень несовершенства статистических данных о системе.

Для упрощения дальнейшей адаптации методов анализа НКК Силова к новому типу моделей необходимо преобразование непрерывных СВ в дискретные путем дискретизации с заданным шагом. В связи с этим возникает задача подбора оптимальной частоты дискретизации, которая обеспечивать баланс между точностью представления данных и вычислительными затратами.

В целом, использование СВ для представления параметров НКК позволит формализовать неопределенность, связанную с оценкой этих параметров. Такой подход, учитывающий потенциальные ошибки и погрешности в ходе оценки, обеспечит повышение точности и надежности результатов моделирования. Разработка специализированного математического аппарата для построения и анализа НКК, параметры которых представлены в форме СВ, позволит получать более подробные результаты моделирования, приводящие к более обоснованным управленческим решениям.

4. Анализ устойчивости параметров когнитивных моделей

Другой предлагаемый подход к учету фактора неопределенности значений параметров НКК основан на следующей идее: вместо устранения возможных ошибок, допущенных при определении значений параметров, целесообразно исследовать, как именно различные изменения этих значений влияют на результаты анализа модели. Такой тип исследования и соответствующий подход в целом будем далее называть анализом устойчивости когнитивной модели. Следует отличать данное понимание термина

«анализ устойчивости» от схожих процессов исследования когнитивных карт (описанных, например, в [Робертс, 1986]), при котором цель анализа состоит в выявлении возможной нестабильности динамики модели в условиях воздействия импульсного процесса, обусловленной особенностями структуры и параметров когнитивной карты.

Проведение анализа устойчивости предполагает формирование пространства гипотетически возможных результатов моделирования, полученных в условиях различных предположений о наличии и величине ошибки в тех или иных параметрах когнитивной модели. Анализ этого пространства с позиции различий между результатами позволит находить обоснованные ответы на следующие значимые вопросы:

- Каким параметрам когнитивной модели следует уделить наибольшее внимание при верификации? То есть, возможно выявление параметров, ошибки в которых наиболее существенно повлияют на результаты моделирования, в том числе тех параметров, ошибки в значениях которых могут привести к принципиально неверному предсказанию динамики моделируемой системы в конкретных условиях.

- Какие из возможных стратегий управления моделируемой системой являются наиболее робастными, т.е. остаются эффективными даже в условиях допущения значительных неточностей при построении когнитивной модели, а какие, напротив, следует использовать с осторожностью?

- Наличие обоснованных ответов на указанные вопросы позволит, во-первых, оптимизировать процесс верификации когнитивной модели, снизив его трудоемкость за счет введения приоритизации при проверке параметров, а во-вторых, при использовании когнитивного моделирования в задачах управления сложными системами, поспособствует обнаружению более надежных стратегий управления.

Обобщенный алгоритм анализа устойчивости когнитивной модели включает в себя следующие этапы.

1. Сбор и формализация представлений эксперта о возможных ошибках в значениях параметров НКК.

2. Генерация на основе исходной НКК и полученных данных множества альтернативных вариантов НКК с изменениями в значениях параметров.

3. Получение результатов моделирования для всех сгенерированных вариантов НКК.

4. Анализ результатов моделирования с целью нахождения ответов на сформулированные выше актуальные типы вопросов о модели. Используемые при этом методы и подходы могут быть ориентированы как на исследование количественных характеристик полученных результатов (в том числе посредством специально разработанных метрик), так и на использование когнитивных способностей человека-аналитика за счет привлечения механизмов визуализации информации [Захарова и др., 2013] и проведения визуального анализа [Захарова и др., 2017].

Реализация каждого из перечисленных этапов требует соответствующего математического, алгоритмического и программного обеспечения, разработка которого и является главной задачей развития описанного подхода.

Заключение

Представленные в работе подходы предложены в рамках исследования проблемной ситуации в когнитивном моделировании, связанной с невозможностью получения полностью достоверных значений параметров когнитивной модели при ее построении. Оба подхода в конечном итоге направлены на повышение результативности когнитивного моделирования как средства выработки стратегий управления сложными системами.

Разработка математического аппарата описанных в статье когнитивных моделей нового типа подразумевает существенную модернизацию всех этапов технологии когнитивного моделирования на основе НКК Силова за счет создания новых и адаптации существующих методов идентификации, верификации, структурно-целевого и сценарного анализа, а также визуализации модели.

Закономерным направлением дальнейшего развития подхода к анализу устойчивости когнитивных моделей является включение в рассмотрение помимо параметров модели также и ее структурных элементов.

Еще одним направлением исследований является выработка критериев сравнения результативности двух представленных подходов к противодействию параметрической неопределенности НКК, для дальнейшего проведения экспериментов на прикладных задачах и формирования обоснованных практикой представлений о перспективности развития каждого из подходов.

Список литературы

- [Борисов и др., 2012] Борисов В.В., Круглов В.В., Федулов А.С. Нечеткие модели и сети. – М.: Горячая линия – Телеком, 2012. – 284 с.
- [Борисов, 2024] Борисов В.В. Мягкие реляционные темпоральные когнитивные модели: прямые и обратные задачи // Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов XII Международной научно-практической конференции. Т. 1. – Смоленск: Универсум, 2024. – С. 85-93.
- [Захарова и др., 2013] Захарова А.А., Шкляр А.В. Метафоры визуализации // Научная визуализация. – 2013. – Т. 5, № 2. – С. 16-24.
- [Захарова и др., 2017] Захарова А.А., Вехтер Е.В., Шкляр А.В. Методика решения задач анализа данных при использовании аналитических визуальных моделей // Научная визуализация. – 2017. – Т. 9, № 4. – С. 78-88.
- [Захарова и др., 2020] Захарова А.А., Подвесовский А.Г., Исаев Р.А. Нечеткие когнитивные модели в управлении слабоструктурированными социально-экономическими системами // Информационные и математические технологии в науке и управлении. – 2020. – № 4(20). – С. 5-23. DOI: 10.38028/ESI.2020.20.4.001.

- [Исаев и др., 2017] Исаев Р.А., Подвесовский А.Г. Оценка согласованности суждений эксперта при построении функции принадлежности нечеткого множества методом множеств уровня // Современные информационные технологии и ИТ-образование. – 2017. – Т. 13, № 3. – С. 9-15. – doi: 10.25559/SPITO.2017.3.499.
- [Исаев и др., 2020] Исаев Р.А., Подвесовский А.Г. Верификация причинно-следственных связей в когнитивных моделях на основе применения метафор визуализации нечетких когнитивных карт // Научная визуализация. – 2020. – Т. 12, № 4.
- [Подвесовский и др., 2019] Подвесовский А.Г., Исаев Р.А. Идентификация структуры и параметров нечетких когнитивных моделей: экспертные и статистические методы // International Journal of Open Information Technologies. – 2019. – Т. 7, № 6.
- [Робертс, 1986] Робертс Ф.С. Дискретные математические модели с приложениями к социальным, биологическим и экологическим задачам: пер. с англ. – М.: Наука, 1986. – 496 с.
- [Силов, 1995] Силов В.Б. Принятие стратегических решений в нечеткой обстановке. – М.: ИНПРО-РЕС, 1995.
- [Aguilar, 2003] Aguilar J. A dynamic fuzzy cognitive map approach based on random neural networks // Int. J. Comput. Cognit. – 2003. – Vol. 1(4). – P. 91-107.
- [Borisov et al., 2022] Borisov V., Luferov V. Method for Forecasting Multidimensional Time Series Based on Neuro-Fuzzy Cognitive Temporal Models // In: Proceedings of the Fifth International Scientific Conference “Intelligent Information Technologies for Industry” (ITI’21). Lecture Notes in Networks and Systems. Vol. 330. – Springer, Cham, 2022. – doi: 10.1007/978-3-030-87178-9_15.
- [Borisov et al., 2024] Borisov V.V., Zharkov A.P., Luferov V.S. Scenario Modeling Based on Fuzzy Relational Temporal Cognitive Models // Pattern Recognit. Image Anal. – 2024. – Vol. 34(3). – P. 624-631. – doi: 10.1134/S10546661824700445.
- [Cai et al., 2010] Cai Y., Miao C., Tan A.-H., Shen Z., Li B. Creating an immersive game world with evolutionary fuzzy cognitive maps // IEEE J. Comput. Graphics Appl. – 2010. – Vol. 30(2). – P. 58-70.
- [Kosko, 1986] Kosko B. Fuzzy cognitive maps // International Journal of Man-Machine Studies. – 1986. – Vol. 24. – P. 65-75.
- [Yager, 1982] Yager R.R. Level sets for membership evaluation of fuzzy subset // Fuzzy Sets and Possibility Theory: Recent Developments (R.R. Yager, ed.). – Pergamon, New York, 1982. – P. 90-97.

УДК 004.82
doi: 10.15622/rcai.2025.039

ИНТЕРПРЕТИРУЕМЫЙ МНОГОСЛОЙНЫЙ КЛАССИФИКАТОР НА ОСНОВЕ НЕЧЕТКО-НЕЙРОСЕТЕВОЙ КОГНИТИВНОЙ МОДЕЛИ В ЗАДАЧАХ ТЕХНИЧЕСКОЙ ДИАГНОСТИКИ

Я.С. Нападайло (*napadailo@skynapp.com*)

В.А. Тайлаков (*vic7519@mail.ru*)

Г.В. Рыбина (*gvrybina@yandex.ru*)

Национальный исследовательский ядерный университет МИФИ,
Москва

В работе рассматривается интерпретируемый многослойный классификатор на основе нечетко-нейросетевой когнитивной модели для решения задач технической диагностики, в частности диагностики неисправностей ветрогенератора, а также оценивается эффективность и точность обнаружения неисправностей в сравнении с классификатором на базе рекуррентной нейронной сети. Сравниваются различные подходы к интерпретации вывода моделей типа “черный ящик”, применяемых в задачах технической диагностики, и дополненных нечеткой когнитивной моделью для объяснения результатов их работы, с подходом к интерпретации результатов диагностики неисправностей на основе предлагаемой авторами модели.

Ключевые слова: техническая диагностика, интерпретируемость, нечеткая когнитивная модель, интеллектуальные системы поддержки принятия решений.

Введение

В задачах технической диагностики и связанных с ними задачах управления объектами технической инфраструктуры в настоящее время широко применяются гибридные интеллектуальные системы, в частности интеллектуальные системы поддержки принятия решений для динамических областей, особенно в условиях неопределенности, нелинейности, зашумленности [Рыбина, 2014]. Непосредственный интерес для исследователя представляют методы “черного ящика”, возникшие как следствие

второй и третьей волн искусственного интеллекта [Забежайло и др., 2022]: глубокие нейронные сети, рекуррентные нейронные сети, нейро-нечеткие модели и др. На основе данных методов разрабатываются различные компоненты интеллектуальных систем, направленные на решение специфических задач в рамках гибридного подхода, обеспечивающего синергетический эффект в результате объединения двух и более диагностических парадигм.

Однако, с ростом популярности применения моделей “черного ящика” к задачам технической диагностики возникает беспокойство относительно причин, побудивших модель принять то или иное решение [Аверкин, 2023]. Эта проблема особенно актуальна для критически важных объектов технической инфраструктуры: оператор мониторинга должен получать как можно больше контекстной информации, касающейся возможных сбоев или аварийных ситуаций, и иметь основание доверять этим данным. Интерпретируемость как основа доверия человека интеллектуальной системе поддержки принятия решений является одним из ключевых принципов третьей волны искусственного интеллекта. Довольно перспективным и интересным направлением в области разработки интерпретируемых моделей становится исследование алгоритмов и методов нечеткой логики, в частности, нейро-нечетких моделей [Ефремова и др., 2017], а также обучаемых нечетких когнитивных моделей. Можно выделить три основные архитектуры построения интерпретируемых моделей на основе нейро-нечетких систем по типу взаимодействия нечеткой системы и модели “черного ящика”: кооперативные, параллельные и гибридные нейро-нечеткие модели [Аверкин, 2024]. В литературе также описаны интерпретируемые полиморфные гибридные модели для решения задач технической диагностики: например, в работе [Mansouri et al., 2023] поведение рекуррентной нейронной сети (РНС) имитируется с помощью нечеткой когнитивной карты (НКК), обученной на тех же данных, что и РНС.

В данной работе рассматривается интерпретируемый многослойный классификатор на основе нечетко-нейросетевой когнитивной модели для решения задач технической диагностики, в частности диагностики неисправностей ветрогенератора, а также оценивается эффективность и точность обнаружения неисправностей по сравнению с классификатором на базе рекуррентной нейронной сети. Сравниваются различные подходы к интерпретации вывода моделей типа “черный ящик”, применяемых в задачах технической диагностики, и дополненных нечеткой когнитивной моделью для объяснения результатов их работы, с подходом к интерпретации результатов диагностики неисправностей на основе предлагаемой авторами модели.

1. Нечетко-нейросетевая когнитивная модель

Нечеткие когнитивные модели (НКМ) [Захарова и др., 2020], в частности НКК, в качестве инструмента поддержки принятия решений находят широкое применение в экономике – для анализа сложных систем в условиях риска и неопределенности [Заграновская, 2018], на производстве – для анализа причин и последствий отказов технических систем в динамических средах [Wang et al., 2022], а также для мониторинга и управления техногенными рисками на предприятиях [Борисов, 2020а], в энергетике и робототехнике – для решения задач диагностики и управления сложными техническими объектами в динамических недетерминированных средах [Karatzinis et al., 2025].

Одной из перспективных разновидностей НКМ с точки зрения ее применения в задачах технической диагностики и управления, является обучаемая модель FCN [Karatzinis et al., 2023]. Отличительной особенностью модели является ее способность функционировать во взаимосвязи с моделируемой динамической системой, что позволяет “дообучать” параметры модели в процессе ее эксплуатации. Согласно классификации, основанной на систематизации нечетких когнитивных моделей [Борисов, 2020b] модель FCN следует отнести к гибридным нечетким моделям “с функциональным замещением”:

- в качестве основной модели берется НКК, которую согласно предложенной в [Кулинич, 2011] классификации когнитивных карт можно отнести к детерминированным качественным когнитивным картам, основанным на правилах, по двум основным классифицирующим признакам;
- в качестве дополнительных компонентов выступают линейный и билинейный адаптивные алгоритмы обучения весов и параметров активационных функций.

Таким образом модель FCN является гибридной нечетко-нейросетевой моделью с параметрической оптимизацией весов и активационных функций на основе алгоритмов обучения с использованием обучающей выборки.

Зададим модель FCN в общем виде:

$$FCN = \left\{ \begin{array}{l} \langle A, W, Z(a), Z(w), R, L \rangle \mid L = LPM \\ \langle A, C, W, Z(a), Z(w), Z(c), R, L \rangle \mid L = BPM \end{array} \right. , \quad (1.1)$$

где A – множество концептов; C – вектор коэффициентов угла наклона функции активации; W – матрица весов; $Z(a)$, $Z(w)$ и $Z(c)$ – шкалы значений концептов, силы влияния концептов и значений углов наклона сигмоидальной функции активации; R – база нечетких правил, которая хранит параметры стационарных состояний динамической системы, L – оператор обучения модели (LPM – линейный, BPM – билинейный). Шкалы значений концептов и силы влияния представлены в виде отображений:

$$Z(a): X \rightarrow \{\mu_x(a_i)\}, i = 1, \dots, m, X \in [0,1], \quad (1.2)$$

$$Z(w): X \rightarrow \{\mu_x(w_i)\}, i = 1, \dots, n, X \in [-1,1], \quad (1.3)$$

$$Z(c): X \rightarrow \{\mu_x(c_i)\}, i = 1, \dots, n, X \in \mathbb{R}^+, \quad (1.4)$$

где $\{\mu_x(a_i)\}$, $\{\mu_x(w_i)\}$ и $\{\mu_x(c_i)\}$ – функции принадлежности, отображающие значения концептов, весов и углов наклона активационных функций в степени принадлежности соответствующим нечетким подмножествам шкалы X . Множество нечетких правил R , устанавливающее соответствие между значениям концептов A , весов W и углов наклона активационных функций C и некоторым стационарным состоянием A^* , задается в виде:

$$R = \left\{ \begin{array}{l} (A, W) \rightarrow A^* \mid L = LPM \\ (A, W, C) \rightarrow A^* \mid L = BPM \end{array} \right. \quad (1.5)$$

Алгоритм построения модели FCN включает два этапа: структурную и параметрическую идентификацию. На первом этапе эксперт задает структуру НКК: множество концептов A и наличие причинно-следственных связей W между ними. Множество концептов задается в виде:

$$A = \langle U, V \rangle, \quad (1.6)$$

где U – множество входных концептов (управляемых и внешних), а V – множество всех остальных концептов (целевых и наблюдаемых). Входной концепт не имеет входящей причинной связи от других концептов и не меняет свое значение в процессе сходимости НКК к точке равновесия. На втором этапе выполняется параметризация НКК посредством обучения её весов на обучающей выборке и последующего за ним формирования базы нечетких правил. Для обучения весов и углов наклона активационных функций в модели FCN применяются линейный и билинейный адаптивные алгоритмы. Линейный адаптивный алгоритм LPM обучения модели FCN описывается системой уравнений [Boutalis et al., 2014]:

$$\varepsilon_i(k) = \frac{f_i^{-1}(A_i^{des}) - w_i(k-1)A^{des}}{c + (A^{des})^T A^{des}}, c > 0, \quad (1.7)$$

$$w_i(k) = w_i(k-1) + \alpha \varepsilon_i(k)(A^{des})^T, \alpha > 0, \quad (1.8)$$

где ε_i – ошибка оценки строки i матрицы весов W на шаге k , $f_i^{-1}(A_i^{des})$ – обратное значение функции активации f_i от заданной точки равновесия системы, $w_i(k-1)$ – строка матрицы весов W на шаге $k-1$, A^{des} – вектор значений концептов в заданной точке равновесия системы, c – гиперпараметр, α – скорость обучения, $c - , w_i(k)$ – обновленная строка матрицы W на шаге k . Для обеспечения сходимости и устойчивости НКК применяют методы ортогональной проекции на выпуклые множества S_1 и S_2 , заданные формулами (1.9) и (1.10).

$$S_1 = \{W \in \mathbb{R}^n | g(w_{ij}) \leq 0, g(w_{ij}) = |w_{ij}| - 1, \forall i, j \in \mathbb{N}\}, \quad (1.9)$$

где w_{ij} – значение веса для пары концептов (A_i, A_j) , $g(w_{ij})$ – ограничение по модулю веса w_{ij} .

$$S_2 = \{\underline{W} \in \mathbb{R}^{n^2} | g(\underline{W}) \leq 0, g(\underline{W}) = \|\underline{W}\| - 4, \quad (1.10)$$

где \underline{W} – вектор столбец, элементами которого являются строки матрицы весов w_i , $\|\underline{W}\|$ – евклидова норма вектор-столбца, а $g(\underline{W})$ – ограничение по норме $\|\underline{W}\|$.

Для уменьшения размера НКК (количества концептов) применяется билинейный адаптивный алгоритм обучения *BPM*, учитывающий вектор коэффициентов C углов наклона функции активации (сигмоиды).

Расширим модель *FCN* (1.1), добавив в ее описание интерпретируемый компонент T :

$$FCN = \left\{ \begin{array}{l} \langle A, W, T, Z(a), Z(w), Z(t), R, L \rangle | L = LPM \\ \langle A, C, W, T, Z(a), Z(w), Z(c), Z(t), R, L \rangle | L = BPM \end{array} \right. \quad (1.11)$$

где T – матрица взаимных влияний концептов, размерность которой равна размерности матрицы весов, $Z(t)$ – шкала значений взаимных влияний концептов. Пусть m – количество всех путей между двумя концептами A_i и A_j . Тогда косвенным эффектом влияния для пары концептов (A_i, A_j) по пути l , проходящему через промежуточные концепты $(A_{k_1^l}, A_{k_2^l}, \dots, A_{k_n^l})$ будем называть минимальное значение веса среди всех смежных пар концептов (A_p, A_{p+1}) на этом пути. Общим эффектом влияния для пары концептов (A_i, A_j) будем считать максимальное из значений косвенных эффектов влияния $I_l(A_i, A_j)$, по всем m возможным путям от A_i к A_j . Общие эффекты влияния для каждой пары концептов (A_i, A_j) вычисляются с помощью аппарата нечеткой причинной алгебры [Kosko, 1986]:

$$I_l(A_i, A_j) = \min\{e(A_p, A_{p+1}): (p, p+1) \in (i, k_1^l, \dots, k_n^l, j)\} \quad (1.12)$$

$$T(A_i, A_j) = \max_{1 \leq l \leq m} I_l(A_i, A_j). \quad (1.13)$$

2. Интерпретируемый многослойный классификатор

Под диагнозом будем понимать класс отклонений параметров динамической системы от их номинальных значений. Тогда задачу технической диагностики можно свести к задаче классификации, а задачу выявления причин, приведших к возникновению этих отклонений – к задаче интерпретации результатов классификации. Для этого построим классификатор, состоящий двух и более слоев, где каждый отдельный его слой представлен одной моделью *FCN*. Для большей ясности обратимся вновь к работе [Борисов, 2020b] и поясним, что из себя представляет этот классификатор согласно систематизации гибридных нечетких моделей. Поскольку мо-

дель *FCN*, представляющая отдельный слой, является гибридной нечетко-нейросетевой моделью с функциональным замещением, то предлагаемый многослойный классификатор будет ни что иное, как композиционная гибридная нечетко-нейросетевая модель для поддержки принятия решений в системах диагностики.

В данной работе ограничимся следующим набором подзадач для решения комплексной задачи технической диагностики: а) обнаружение дефекта (отклонения); б) локализация дефекта (отклонения); в) выявление причины, приведшей к неисправности и/или обоснование принятого решения. Для решения каждой из подзадач будем использовать отдельную модель *FCN*, формирующую отдельный слой классификатора. Методы, выбранные для решения каждой из подзадач, а также буквенные обозначения слоев приведены в табл. 1.

Таблица 1

№	Слой	Подзадача	Метод
#1	L_D	Обнаружение неисправности	Бинарная классификация
#2	L_{D*}	Локализация неисправности	Многоклассовая классификация
#3	L_X	Интерпретация результатов	Статический анализ

Обозначим модель классификатора как *FCNXAI* и опишем структуру модели:

$$FCNXAI = \langle L_D, L_{D*}, L_X \rangle, L_{D*}: \{S_1, \dots, S_m\} \rightarrow \{D_1, \dots, D_n\}, \quad (2.1)$$

где слой L_D – модель *FCN* обнаружения дефекта, которая соотносит текущее состояние объекта диагностирования к одному из режимов функционирования – штатному либо аварийному; слой L_{D*} – модель *FCN* локализации дефекта, $\{D_1, \dots, D_n\}$ – множество дефектов (классов отклонений) объекта диагностирования, n – общее число дефектов (классов отклонений), $\{S_1, \dots, S_m\}$ – множество признаков (симптомов) объекта диагностирования, m – общее число признаков; слой L_X – модель *FCN* интерпретации результатов диагностирования.

Построение классификатора выполняется следующим образом. Для слоя L_D сначала выполняется структурная идентификация модели *FCN*: эксперт задает множество концептов A_D в виде:

$$A_D = \langle U, S, M \rangle, \quad (2.2)$$

где U – множество управляемых и внешних концептов; S – множество признаков (симптомов); M – целевой концепт, который задает режим функционирования системы (штатный или аварийный). Далее веса

W_D обучаются с помощью алгоритма *LPM* на статистических данных. Затем эксперт выполняет структурную идентификацию слоя L_{D*} . Множество концептов A_{D*} задается в виде:

$$A_{D*} = \langle U, S, D \rangle, \quad (2.3)$$

где U – множество входных концептов; S – множество признаков (симптомов); D – множество дефектов (классов отклонений). После этого веса W_{D*} слоя L_{D*} обучаются на статистической выборке с помощью алгоритма *LPM*. Множество концептов A_X слоя L_X задается как:

$$A_X = \langle R, D \rangle, \quad (2.4)$$

где R – множество причин, приведших к неисправности, D – множество дефектов (классов отклонений). Матрицы W_X и T_X слоя L_X обучаются на статистической выборке с помощью алгоритма *LPM*. Наконец, модель *FCN* любого из слоев классификатора *FCNXAI* может быть оптимизирована (редуцирована) при необходимости с помощью алгоритма *BPM*.

Классификатор *FCNXAI* может применяться в задачах технической диагностики по следующему сценарию: на вход слоя L_D подается вектор A_{D*} значений концептов (U, S) , затем слой выполняет задачу обнаружения неисправности. Результат бинарной классификации M передается на слой L_{D*} . Если выявлено, что система работает в аварийном режиме – слой L_{D*} выполняет задачу локализации неисправности и определяет класс неисправности (или несколько классов) системы: для этого на вход слоя L_{D*} подается вектор A_{D*} значений концептов (U, S) . Далее результат многоклассовой классификации D , а также вектор A_X значений концептов (R) подается на слой L_X , который интерпретирует полученный на предыдущем шаге результат.

3. Анализ различных подходов к интерпретации в задачах технической диагностики

Примеры интерпретируемых моделей “черного ящика” в различных приложениях поддержки принятия решений подробно описаны в литературе. Так, в работе [Wang et al., 2021] представлена гибридная нечетко-нейросетевая модель *DFCM* на основе НКК для прогнозирования загруженности автомобильных дорог; веса модели представляют собой обученные глубокие нейронные сети (ГНС), моделирующие нелинейные зависимости динамической системы, а также включают рекуррентные нейронные сети (РНС), учитывающие влияние внешних возмущений. В [Mansouri et al., 2023] гибридная полиморфная нечеткая когнитивная модель *LFCM*, поддерживающая интерпретацию вывода РНС, обучается на тех же данных, что и модель РНС; модель *LFCM* применяется в задаче диагностики неисправностей ветрогенератора. Отметим, что в обоих случаях описанная модель представляет собой результат гибридизации ней-

ронной сети, моделирующей динамическое поведение системы, и НКК, описывающей причинно-следственные связи между отдельными компонентами системы (концептами).

В предложенном подходе нелинейные отношения между концептами каждого слоя классификатора моделируются с помощью множества нечетких правил R , которые описывают параметры стационарных состояний динамической системы; при этом алгоритмы LPM и BPM гарантируют сходимость модели FCN в процессе ее обучения, а также обеспечивают взаимодействие с моделируемой динамической системой в режиме реального времени. Встроенный механизм интерпретации результатов диагностики, в рамках системы поддержки принятия решений, позволяет выявить причину, приведшую к неисправности и предоставить оператору мониторинга обоснование принятого решения. Кроме того, модель обладает низкой вычислительной нагрузкой, что делает ее применимой в приложениях реального времени.

Рассмотрим применение предложенного классификатора на примере решения двух задач диагностики, описанных в подразделах 3.1 и 3.2.

3.1. Задача диагностики неисправностей подшипников

В качестве набора данных для проведения эксперимента был взят публичный датасет со значениями, снятыми с датчиков вибраций, собранными с 4-х промышленных подшипников на протяжении 4-х месяцев. Все подшипники имели одинаковые характеристики (класс 1, до 15 кВт), а в качестве порога вибрации для данного типа подшипников было выбрано среднее квадратичное значение виброскорости $v_{r.m.s.}$ 0.5 мм/с согласно стандарта ГОСТ ИСО 10816-1–97. Три подшипника находились в исправном состоянии, четвертый подшипник - неоднократно выходил из строя.

Для задачи диагностики неисправностей подшипников был построен классификатор $FCNXAI$, обученный на статистической выборке из экспериментального набора данных с помощью алгоритма LPM . После обучения модель $FCNXAI$ была проверена на тестовом наборе данных. Для проведения сравнительной оценки эффективности модели был выбран классификатор $LFCM$. Модель $LFCM$ демонстрирует немного более высокую точность на тестовой выборке; в то же самое время предлагаемая модель $FCNXAI$ обеспечивает существенно меньшую вычислительную сложность, а также высокую интерпретируемость (рис. 1). В отличие от модели $LFCM$, в которой элементы матриц весов W и T имеют фиксированные значения после обучения, модель $FCNXAI$ сохраняет нелинейные отношения в виде множества нечетких правил R , обеспечивая более точную и контекстно-зависимую интерпретацию в условиях выраженной нелинейности исходных данных.

Таблица сравнения:

	Train Time (s)	Accuracy	AUC	Inference Time (ms/sample)
RNN	84.7099	0.9763	0.9257	0.0228
CNN	107.3366	0.9755	0.9379	0.0425
LFCM	56.6601	0.9744	0.9426	0.0164
FCNXAI	50.4289	0.9507	0.9002	0.0150

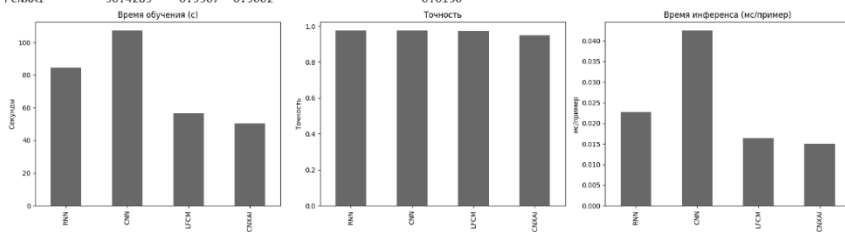


Рис. 1. Сравнение быстродействия моделей

3.2. Задача диагностики неисправностей обмотки генератора

Для решения задачи диагностики неисправностей обмотки генератора была построена тепловая модель *FCNXAI*, учитывающая влияние внешних факторов, таких как скорость ветра, температура окружающей среды и влажность. Внешние факторы представлены множеством входных концептов U , тогда как все остальные концепты, включая значения температуры обмоток генератора, образуют множество V . Значения внешних факторов для обучающей и тестовой выборок были взяты из метеорологической базы данных *gistemeo.kz* для города Алматы (Казахстан), а значения температуры обмоток генератора получены с температурных датчиков, установленных внутри корпуса 3-х фазного генератора с прямым приводом мощностью 1 кВт и частотой вращения вала 300 об/мин.

Модель позволяет заранее предсказать возможный перегрев обмоток с учетом изменения внешних факторов окружающей среды и эксплуатационных режимов, а также помогает выяснить, какие именно внешние факторы вносят наибольший вклад в рост температуры.

При проведении статического (структурно-целевого) и динамического (сценарного) анализа было установлено, что наиболее критичным является сочетание высоких температур воздуха (40°C) и низкой скорости ветра (менее 3 м/с), при котором температура обмоток может достигать 150°C , превышая допустимые значения для некоторых стандартных классов изоляции (ГОСТ Р МЭК 60085–2011). На рис 2 приведён анализ влияния внешних факторов на температуру обмотки генератора в виде радиальной диаграммы, включающей восемь концептов (A1–A8), где A1 – температура обмотки, A2 – скорость ветра, A3 – влажность, A8 – температура воздуха, A4–A7 – наблюдаемые параметры. В другом сценарии анализ показал, что повышенная влажность воздуха (более 70%) увеличивает температуру обмоток на 5–10% за счет ухудшения теплоотдачи, связанного с изменением термических свойств воздуха.

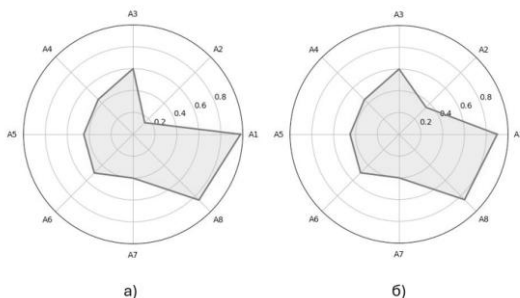


Рис. 2. Анализ влияния внешних факторов на температуру обмотки:
а) низкая скорость ветра, б) средняя скорость ветра

Заключение

В данной работе рассмотрен интерпретируемый многослойный классификатор *FCNXAI* на основе гибридной нечетко-нейросетевой модели для решения задач технической диагностики, в частности диагностики неисправностей ветрогенератора. Предлагаемый классификатор, с одной стороны, позволяет моделировать сложные нелинейные зависимости динамической системы, а также обеспечивает сходимость модели во время ее обучения, а с другой – предоставляет оператору мониторинга интерпретируемый вывод, обоснование принятого моделью решения. Эксперименты показали, что модель обладает высокой интерпретируемостью и более низкой вычислительной способностью по сравнению с другими гибридными нечетко-нейросетевыми моделями, такими как *LFCM*, что делает возможным ее применение в задачах реального времени.

Дальнейшие исследования будут направлены на изучение взаимодействия двух и более классификаторов, предназначенных для решения задачи диагностики отдельных компонентов сложной динамической системы. Кроме того, планируется уделить внимание исследованию методов оптимизации (упрощения структуры) модели с помощью алгоритма *BPM*.

Список литературы

- [Аверкин, 2023] Аверкин А.Н. Объяснимый искусственный интеллект как часть искусственного интеллекта третьего поколения // Речевые технологии. – 2023. – № 1. – С. 4-10.
- [Аверкин, 2024] Аверкин А.Н. Объяснительный искусственный интеллект в больших речевых моделях // Речевые технологии. – 2024. – № 1. – С. 3-13.
- [Борисов, 2020а] Борисов В.В. Нечёткие когнитивные модели как основа для исследования сложных систем и процессов // Речевые технологии. – 2020. – № 1-2. – С. 48-62.

- [Борисов, 2020b] Борисов В.В. Систематизация нечетких и гибридных нечетких моделей // Мягкие измерения и вычисления. – 2020. – Т. 29, № 4. – С. 98-120.
- [Ефремова и др., 2017] Ефремова Н.А., Аверкин А.Н., Ярушев С.А. Гибридные нечёткие когнитивные карты в задачах поддержки принятия решений и прогнозирования // Программные продукты, системы и алгоритмы. – 2017. – № 4. – С. 18-25.
- [Захарова и др., 2020] Захарова А.А., Подвесовский А.Г., Исаев Р.А. Нечеткие когнитивные модели в управлении слабоструктурированными социально-экономическими системами // Информационные и математические технологии в науке и управлении. – 2020. – № 4 (20). – С. 5-23.
- [Забежайло и др., 2022] Забежайло М.И., Борисов В.В. Об интерпретациях понятия “искусственный интеллект” // Речевые технологии. – 2022. – № 1. – С. 5-18.
- [Заграновская, 2018] Заграновская А.В. Системный анализ на основе нечётких когнитивных карт // Вестник Российского экономического университета имени Г.В. Плеханова. – 2018. – № 4. – С. 152-160.
- [Кулинич, 2011] Кулинич А.А. Классификация когнитивных карт и методы их анализа // 6-я международная научно-техническая конференция “Интегрированные модели и мягкие вычисления в искусственном интеллекте” (Коломна, 2011 г.). Труды конференции. Т. 1 – М.: Физматлит, 2011. – С. 124-135.
- [Рыбина, 2010] Рыбина Г.В. Основы построения интеллектуальных систем. – М: Финансы и статистика; ИНФРА-М, 2010. – 432 с.
- [Boutalis et al., 2014] Boutalis Y., Theodoridis D., Kottas T., Christodoulou M.A. Theory and Applications of the Neurofuzzy and Fuzzy Cognitive Network Models. System Identification and Adaptive Control. – Springer International Publishing, 2014. doi: 10.1007/978-3-319-06364-5.
- [Karatzinis et al., 2023] Karatzinis G.D., Apostolikas N.A., Boutalis Y.S., Papakostas G.A. Fuzzy Cognitive Networks in Diverse Applications Using Hybrid Representative Structures. International Journal of Fuzzy Systems. – 2023. – Vol. 25. – P. 2534-2554. – doi: 10.1007/s40815-023-01564-4.
- [Kosko, 1986] Kosko B. Fuzzy Cognitive Maps // International Journal of Man-Machine Studies. – 1986. – Vol. 24(1). – P. 65-75. – doi: 10.1016/S0020-7373(86)80040-2.
- [Mansouri et al., 2023] Mansouri T., Vadera S. Explainable fault prediction using learning fuzzy cognitive maps // Expert systems. – 2023. – Vol. 40(8). – Article e13316. – doi: 10.1111/exsy.13316.
- [Wang et al., 2021] Wang J., Peng Z., Wang X., Li C., Wu J. Deep Fuzzy Cognitive Maps for Interpretable Multivariate Time Series Prediction // IEEE Transactions on Fuzzy Systems. – 2021. – Vol. 29(9). – P. 2647–2660. – doi: 10.1109/TFUZZ.2020.3005293.
- [Wang et al., 2022] Wang W., Wang Y., Han X. A dynamic failure mode and effect analysis for train systems failures risk assessment using FCM and prospect theory // Management System Engineering. – 2022. – Vol. 1. – Article 8. – doi: 10.1007/s44176-022-00008-x.

УДК 004.8

doi: 10.15622/rcai.2025.040

ИСПОЛЬЗОВАНИЕ НЕЧЕТКИХ МНОЖЕСТВ ДЛЯ ПОСТРОЕНИЯ ГИБРИДНОГО ОБЪЯСНИТЕЛЬНОГО ИСКУССТВЕННОГО ИНТЕЛЛЕКТА НОВОГО ПОКОЛЕНИЯ¹

Ю.В. Трофимов (*ura_trofim@bk.ru*)

А.Н. Аверкин (*averkin2003@inbox.ru*)

А.К. Алексеев (*aak.24@uni-dubna.ru*)

А.В. Шевченко (*leviathan0909@gmail.com*)

Государственный университет Дубна, Дубна

В работе представлен обзор современных формализмов теории множеств, используемых для представления неопределенности в объяснимом искусственном интеллекте (ХАИ). Рассматриваются классические нечеткие множества Л. Заде, их расширения – интуиционистские нечеткие множества К. Атанасова, нейтрософические множества Ф. Смарандаче, а также новейший формализм плитогенных множеств. Показано, как эти подходы позволяют моделировать не только степенную неопределенность, но и неполноту и противоречивость знаний. Обсуждается роль данных формализмов в эволюции ХАИ от первого поколения (ХАИ 1.0) к более продвинутым концепциям ХАИ 2.0 и ХАИ 3.0.

Ключевые слова: ХАИ, искусственный интеллект, нечеткие множества.

Введение

Современные ИИ-системы достигли выдающихся результатов, но их внутренние механизмы остаются малопонятными даже разработчикам: модель выдает ответ, однако причины решения скрыты в миллионах параметров. Такая природа затрудняет проверку корректности вывода, юридическую ответственность и доверие пользователей. Именно эту проблему призван решить объяснимый искусственный интеллект (ХАИ), задача которого – вернуть человеку возможность проследить, почему модель

¹ Работа выполнена в рамках государственного задания Министерства науки и высшего образования Российской Федерации (тема № 124112200072-2).

пришла к тому или иному результату. Первое поколение подходов ХАИ фокусировалось преимущественно на post hoc объяснениях для статичных моделей: визуализация важности признаков, построение локальных аппроксимаций (например, LIME, SHAP), тепловые карты активаций нейросетей и т.п.. Эти методы позволяли частично пролить свет на решения модели, но были ограничены тем, что не устраняли саму причину неинтерпретируемости – сложность и непрозрачность внутреннего представления знаний. [Averkin et al., 2024] Осознание ограничений ХАИ 1.0 привело к формированию концепции ХАИ 2.0 – второго поколения объяснимого ИИ. Его отличает проактивная интеграция интерпретируемости в полный цикл разработки и акцент на встроенную (intrinsic) прозрачность. Методологическим ориентиром здесь служит классическая теорема Колмогорова – Арнольда, впервые показавшая, как любую многомерную функцию можно разложить в сумму элементарных одномерных преобразований [Колмогоров, 1957]. Именно на этом фундаменте строятся Kolmogorov-Arnold Networks (KAN): каждая связь сети реализует обучаемую одномерную кривую-сплайн, а узлы лишь агрегируют их в аналитически читаемое представление [Ziming et al., 2024]. Тем самым KAN выступают практическим воплощением идей «отцов-основателей» интерпретируемости, демонстрируя, как модели ХАИ 2.0 могут сочетать гибкость глубокого обучения с прозрачностью символьной регрессии, обеспечивая согласованность вывода и количественный учет неопределенности.

Помимо архитектурной объяснимости «Манифест ХАИ 2.0» указывает на 27 открытых исследовательских задач, решение которых предполагает междисциплинарный подход и дальнейшее развитие данной парадигмы [Longo et al., 2024]. Для достижения этих целей используются гибридные архитектуры, сочетающие методы машинного обучения с выразительными средствами нечеткой логики (soft-logic). Soft-logic охватывает нечеткие множества и их расширения – интуиционистские, нейтрософические, плитогенные множества, которые позволяют моделировать знания с учетом неопределенности, приближенности и противоречивости. Подобные модели способны объединять разные критерии оценки, учитывать контекст и явно представлять степень уверенности выводов, что соответствует целям ХАИ 2.0.

Новейшая концепция – ХАИ 3.0 – идет еще дальше и предполагает появление саморефлективных ИИ-систем, которые не только объясняют свои решения, но и способны оценивать собственные объяснения. Такие системы опираются на формализованное знание и выполняют внутреннюю мета-оценку своих выводов. От ХАИ 3.0 ожидают умения выявлять когнитивные конфликты, проверять соответствие вывода базовым принципам домена и оценивать устойчивость вывода при различных допущениях. Для реализации этих функций необходимы новые формальные инструменты, позволяющие ИИ представлять собственные знания на мета-

уровне и обнаруживать несогласованности. Одним из ключевых является плитогенный подход, предоставляющий математический аппарат для такой мета-оценки. Плитогенные множества, как будет показано, позволяют системе одновременно оперировать несколькими «взглядами» на вывод и количественно измерять согласованность между ними, что играет центральную роль в архитектуре ХАИ 3.0. В данной работе рассматриваются основные типы обобщенных множеств – нечеткие, интуиционистские, нейтрософические и плитогенные – с точки зрения их теоретических основ и применения в задачах объяснимости ИИ. Прослеживается эволюция использования этих формализмов в ХАИ от версии 1.0 к 2.0 и далее к зарождающимся идеям ХАИ 3.0.

Soft-logic как основа гибридного ИИ

Одним из краеугольных камней архитектуры ХАИ 3.0 является использование soft-logic – совокупности методов логического вывода и представления знаний, допускающих неопределенность, приближенность и противоречивость. К soft-logic относят, прежде всего, нечеткую логику (fuzzy logic) и обобщения классического понятия множества: нечеткие множества (Fuzzy Sets), а также их расширения – интуиционистские нечеткие множества (Intuitionistic Fuzzy Sets), нейтрософические множества (Neutrosophic Sets) и плитогенные множества (Plithogenic Sets). Эти формализмы позволяют более гибко описывать знание и логику рассуждений, приближаясь к тому, как человек оперирует понятиями неопределенности в реальных ситуациях.

Нечеткая логика была предложена Лотфи Заде в 1965 г. как развитие идей о плавных градуированных множествах. В нечетком множестве каждому элементу x присваивается степень принадлежности $\mu(x)$ в диапазоне $[0, 1]$, отражающая степень истинности утверждения, что x принадлежит данному понятию [Zadeh, 1965]. В отличие от классической двоичной логики, где утверждение либо истинно, либо ложно, нечеткая логика позволяет утверждению быть истинным в некоторой степени. Такой подход ближе к человеческому мышлению, где многие понятия имеют размытые границы (например, «высокий рост» или «теплая погода» не имеют четко определенного порога). Нечеткая логика и базы правил на ее основе исторически рассматривались как интерпретируемые модели: правила вида «ЕСЛИ X И Y , ТО Z с вероятностью P » легко понятны эксперту, а параметр P отражает степень уверенности. Именно способность работать с нечеткими понятиями делает этот аппарат ценным для ХАИ – он обеспечивает обработку неопределенности при сохранении прозрачности рассуждений. По сути, нечеткая логика служит краеугольным камнем объяснимости в ИИ, предлагая компромисс между строгой точностью и понятной человечностью рассуждений.

В России наиболее значимые результаты в области исследований и разработки современных методов нечетких логик, были получены в работах таких ведущих российских ученых, как Афанасьева Т.В., Батыршин И.З., Борисов В.В., Вагин В.Н., Воронцов К.В., Еремеев А.П., Заболеева-Зотова А.В., Ковалев С.М., Комарцова Л.Г., Курейчик В.В., Курейчик В.М., Лефевр В.А., Лепский В.Е., Молодцов Д.А., Осипов Г.С., Перфильева И.Г., Петровский А.Б., Поспелов Д.А., Силов В.Б., Тарасов В.Б., Фоминых И.Б., Ярушкина Н.Г. Среди зарубежных исследований можно выделить работы Armstrong J.S., Banerjee A., Chandola V., Chen Y.-Y., Forrest S., Keogh E., Kosko B., Kumar V., Mizutani E., Novak V., Povinelli R., Roger Jang J.-S., Rutkowska D., Sun C.-T., Wang L.-X., Xu X.

Интуиционистские нечеткие множества (ИНМ), введенные К. Атанасовым в 1986 г., расширяют аппарат нечетких множеств введением двух функций принадлежности: степени принадлежности $\mu(x)$ и степени непринадлежности $\nu(x)$ для каждого элемента x . В ИНМ не требуется, чтобы $\nu(x) = 1 - \mu(x)$, как это было бы в классическом случае; вместо этого допускается нерешенность или сомнение, равное $1 - \mu(x) - \nu(x)$ [Atanassov, 1986]. Это позволяет явно моделировать ситуацию, когда информация о принадлежности элемента к понятию недостаточна или противоречива – т.е. существует зона неопределенности (неуверенности). Интуиционистские множества точнее отражают реальный процесс экспертной оценки: эксперт может сказать, что некоторый факт частично подтверждает гипотезу (например, $\mu=0.6$), частично ей противоречит ($\nu=0.2$), а оставшаяся часть (0.2) – это неопределенность из-за нехватки данных или колебаний. Для ХАИ такие конструкции полезны тем, что позволяют отображать степень уверенности и сомнений прямо в объяснениях. Например, объяснение решения может содержать индикацию: «Диагноз А подтверждается на 60%, опровергается на 20%, оставшиеся 20% информации недостаточно для вывода» – что значительно информативнее, чем бинарное «Диагноз А или не А».

Нейтрософические множества – дальнейшее обобщение, предложенное Ф. Смарандэке в конце 1990-х, – вводят трехзначную меру принадлежности: помимо степени истины $T(x)$ и ложности $F(x)$ для элемента x , явно вводится степень неопределенности $I(x)$ [Smarandache, 2004]. Нейтрософическая логика исходит из философской концепции нейтрософии, постулирующей одновременное присутствие правды, лжи и нейтрального (неопределенного) элемента. В нейтрософическом множестве для каждого x допускаются независимые значения $T(x)$, $I(x)$, $F(x)$, не обязательно суммирующиеся к 1 или каким-либо фиксированным константам [Smarandache, 2004]. Это чрезвычайно гибкий аппарат, позволяющий выражать сложные состояния знаний: например, высокая неопределенность (I близко к 1) при одновременно высоких значениях T и F может отражать противоречивые данные или парадоксальную ситуацию. Нейтрософиче-

ский подход удобен для представления противоречивой информации в ХАИ: система может признавать, что аргументы «за» и «против» решения сильны одновременно, указывая на наличие конфликта, который сам по себе становится частью объяснения. Такой подход перекликается с требованием регуляторных стандартов фиксировать неопределенность и риск наряду с выводом – прозрачная система должна уметь показать, где она не уверена.

Плитогенные множества – новейшее расширение теории множеств, также разработанное Смарандэке – еще более укрепляют гибкость представления знаний за счет учета множественности атрибутов и их противоречий. В плитогенном множестве каждый элемент характеризуется не одним признаком, а целым набором атрибутов (потенциально разнородных), причем для каждого атрибута задается свое значение принадлежности элемента [Smarandache, 2018]. Более того, вводится понятие степени противоречия между значениями атрибутов. Идея состоит в том, что реальные решения зависят от множества факторов, которые могут конфликтовать друг с другом. Например, при решении о выдаче кредита фактор дохода клиента может «за» (высокий доход), а фактор кредитной истории – «против» (были просрочки). Обычное нечеткое правило могло бы усреднить эти влияния, а плитогенный подход позволяет явно указать степень противоречия между ними и учесть доминирующий фактор. Плитогенное множество обобщает все предыдущие типы множеств (четкие, нечеткие, интуиционистские, нейтрософические) как частные случаи, когда число атрибутов равно 1. Благодаря этому, плитогенный подход обеспечивает наиболее богатый язык для описания условий и объяснений решений. В контексте ХАИ он означает, что объяснение может одновременно учитывать несколько свойств ситуации, выражая для каждого свой уровень истинности и указывая, как разные аспекты взаимодействуют. Тем самым достигается более полное и достоверное объяснение сложных выводов: вместо упрощения, система может сказать «Решение принято, потому что фактор X и Y преимущественно указывают на положительный исход, хотя фактор Z противоречит им на 30%».

Объединение перечисленных компонентов soft-logic в рамках гибридного ИИ открывает возможность для создания символьно-нейросетевых систем. Объяснимость и доверие должны быть заложены в систему изначально, а не добавляться в виде постфактум надстроек. Должна быть сформулирована многоуровневая архитектура, где каждый слой повышает прозрачность, а функции принимают на себя взаимодействующие агенты: одни отвечают за вычислительное решение, другие – за генерацию и адаптацию объяснений, третьи – за непрерывную калибровку доверия пользователя к модели. Факты и правила, выраженные через нечеткие или нейтрософические множества, могут сосуществовать с обучаемыми нейрон-

ными модулями. Soft-logic выступает связующим звеном: она переводит числовые выходы модели в лингвистически понятные категории (через функции принадлежности) и агрегирует их по понятным правилам.

Переход к ХАИ 2.0 и ХАИ 3.0

ХАИ 1.0 характеризуется post hoc методами интерпретации уже после получения решения «черного ящика». Эти методы играли важную роль на начальном этапе, демонстрируя принципиальную возможность объяснения сложных моделей. Однако их ограниченность проявляется в отсутствии встроенной прозрачности: модель все еще изначально не ориентирована на интерпретируемость. Пользователь ХАИ 1.0 получает объяснение в виде важных признаков, но не видит логической структуры принятия решения.

ХАИ 2.0 возник как ответ на эти недостатки. Его идеология – проектировать модели, изначально объяснимые. Это достигается путем объединения моделей машинного обучения с логико-семантическими компонентами, обеспечивающими интерпретацию. Например, нейросеть может быть интегрирована с нечеткой экспертной системой или обучена таким образом, чтобы ее скрытые компоненты имели понятный смысл. Концепция ХАИ 2.0 предполагает, что объяснимые ИИ-системы обладают расширенными гибридными архитектурами, учитывающими сразу несколько критериев оценки решения, сохраняют согласованность с контекстом и явно работают с неопределенностью. Объяснения в ХАИ 2.0 должны быть многомерными, контекстуально непротиворечивыми и отражать степень уверенности и доверия пользователя к выводу. Для реализации таких требований используются формальные модели, способные объединять разнородные признаки и критерии, выявлять и количественно учитывать противоречия между ними, а также предоставлять меры уверенности в выводах. На практике переход к ХАИ 2.0 выражается в появлении моделей с внутренней интерпретируемостью. Кроме того, ключевую роль начинают играть описанные ранее формализмы неопределенности. Плитогенные множества оказались практически идеально согласующимися с философией ХАИ 2.0. Благодаря многокритериальной природе модели каждый элемент (решение) может быть описан совокупностью атрибутов (аспектов объяснения) с указанием степени принадлежности по каждому из них. В отличие от одноаспектных объяснений ХАИ 1.0 (когда дается лишь одна точка зрения – например, «важны признаки А, В, С»), плитогенный подход интегрирует несколько источников оценок в единое объяснение. При этом противоречивость между критериями не замалчивается, а формализуется явно: введена функция степени противоречия, которая количественно оценивает конфликт между различными признаками решения. Если два критерия дают противоречивые оценки элемента, соответствующая

величина отразит степень их несогласованности. Эти значения используются при агрегировании: операции пересечения и объединения в плитогенной логике определяются как линейная комбинация стандартных логических операций, взвешенная с учетом величины противоречия. Механизм объединения критериев становится адаптивным: высокая противоречивость между некоторыми атрибутами ослабляет их совместное влияние (сдвигая операцию в сторону дизъюнктивного объединения), тогда как согласованные критерии агрегируются ближе к конъюнкции. В результате система автоматически фильтрует конфликтующие факторы, снижая их вклад, и обеспечивает контекстуальную согласованность итогового вывода. Кроме того, плитогенный формализм дает средства для явного представления неопределенности, что критично для ХАИ 2.0. Поскольку степени принадлежности могут задаваться в нечеткой, интуиционистской или нейтрософической шкале, модель учитывает не только «истину» и «ложь» критериев, но и долю неопределенности в них. Например, используя нейтрософические оценки внутри плитогенного множества, система для каждого критерия может указать степень истинности его свидетельства, степень ложности и степень неопределенности. Тем самым неопределенность становится операционализированной и учитывается количественно при вычислении вывода.

Таким образом, на этапе ХАИ 2.0 мы видим слияние методов машинного обучения с *soft-logic* – аппаратами нечеткой и многозначной логики – для достижения глубоко интерпретируемых решений. Плитогенные и смежные формализмы выступают как связующее звено, позволяющее описать сложное поведение модели на языке, понятном человеку.

ХАИ 3.0 пока во многом является концепцией, формирующейся в исследованиях. Это видение системы ИИ, которая способна не только объяснить конечному пользователю свой вывод, но и самостоятельно проанализировать и обосновать этот вывод, обнаружить свои слабые места. По сути, ХАИ 3.0 добавляет мета-уровень: ИИ рассуждает о собственных рассуждениях. Предполагается, что такие системы будут онтологически обоснованы – то есть обладать внутренней мета-моделью знаний (например, онтологией предметной области и критериев решения) – и за счет этого смогут контролировать соответствие своих решений этой мета-модели. От ХАИ 3.0 ожидают, в частности, что система сможет указать: есть ли когнитивный конфликт, насколько вывод соответствует фундаментальным принципам предметной области (доверие к выводу со стороны доменной теории), и насколько вывод устойчив к смене условий (например, если изменятся входные данные в разумных пределах, останется ли решение таким же). Эти проверки выступают аналогом самоконтроля ИИ. Для реализации саморефлексии необходим все тот же богатый математический аппарат *soft-logic*. В частности, как отмечается, плитогенная

модель предоставляет средства для мета-оценки вывода ИИ на всех требуемых уровнях. Каждый вывод или факт в системе ХАИ 3.0 можно представить как элемент плитогенного множества с несколькими атрибутами, отражающими разные «срезы» знания о нем. Один атрибут может соответствовать степени истинности вывода на основе данных, другой – степени согласования вывода с онтологическими ограничениями, третий – отражать выполнение вывода при различных допущениях или в различных сценариях. В терминах плитогенной логики это означает, что одно суждение характеризуется множеством значений истинности по разным основаниям, после чего вычисляется его совокупная интегральная истинность.. Благодаря этому система ХАИ 3.0 может внутренне анализировать: согласуются ли между собой разные «точки зрения» на утверждение. Противоречия, возникающие между атрибутами, будут явно представлены через вычисленные степени их несогласованности. Высокое значение функции противоречия между некоторыми аспектами оценки укажет на когнитивный конфликт, то есть на наличие несовместимых убеждений или данных в базе знаний относительно данного вывода. Опираясь на онтологию, система сможет интерпретировать, какие именно аспекты находятся в конфликте (например, вывод противоречит одному из принципов теории), и понизить доверие к такому выводу на количественную величину, пропорциональную степени конфликтности. Напротив, если все атрибуты (все точки зрения) согласованы, система фиксирует высокое общее значение истинности вывода и высокое доверие к нему.

Применение плитогенного подхода в ХАИ 3.0 делает объяснения максимально богатыми: ИИ способен прямо заявить, например, «мой вывод X конфликтует с принципом Y на 30%, поэтому степень уверенности в X снижена» – и таким образом продемонстрировать пользователю учет внутренних конфликтов и ограничений. Такая прозрачная мета-оценка призвана укрепить доверие пользователя, поскольку ИИ явно показывает, почему считает свой вывод правильным, и честно указывает на обнаруженные им же самому потенциальные противоречия или ограничения применимости вывода. Плитогенный формализм, таким образом, предоставляет целостный логико-алгебраический язык для описания и объяснения процесса рассуждения ИИ на новом уровне – уровне самообъяснения.

Эволюционные шкалы доверия

Последовательный переход от post hoc методик ХАИ 1.0 к встроенной прозрачности, характерной для ХАИ 2.0, и далее к многоуровневой самообъясняющейся архитектуре ХАИ 3.0, можно отразить сквозь ось «уровень объяснимости». Этапы L0-L6 сведены в схематическую лестницу (рис. 1), где каждой ступени сопоставлены характерные инструменты и инженерные решения.

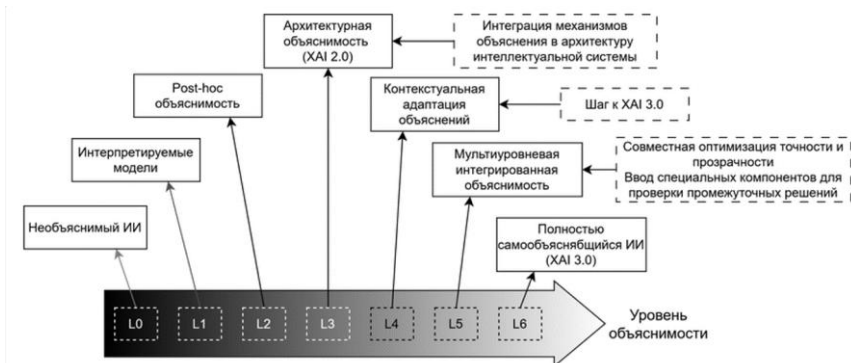


Рис. 1. Лестница объяснимости

Если рис. 1 описывает постепенное углубление объяснимости модели, то следующая измерительная ось L0-L7 характеризует эволюцию доверия пользователя к искусственному интеллекту (рис. 2)

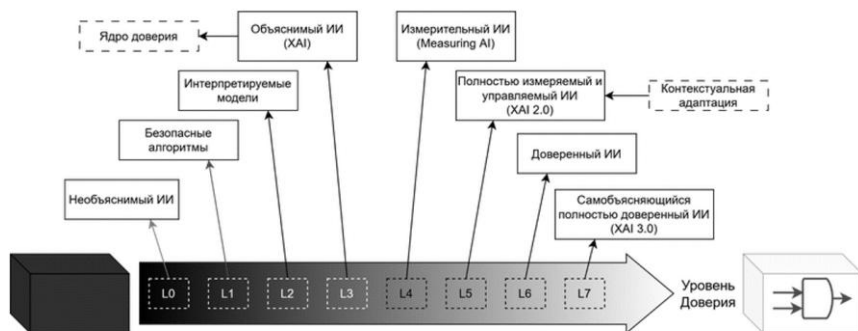


Рис. 2. Лестница доверия к искусственному интеллекту

В совокупности рассмотренные этапы формирования объяснимого интеллекта образуют непрерывную траекторию наращивания прозрачности, ответственности и доверия системы. Для наглядности эволюции была разработана «эволюционная лестница», каждая ступень которой отражает не только технический уровень интерпретируемости, но и характер взаимодействия человека с искусственным интеллектом. Графическое представление приведено на рис. 3.

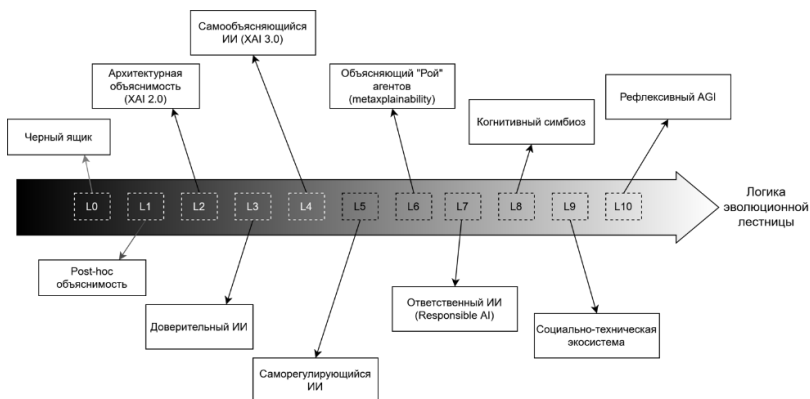


Рис. 3. Эволюционная лестница доверия и объяснимости ИИ

Заключение

Эволюция объяснимого ИИ неразрывно связана с развитием формальных моделей представления неопределенности и знаний. Если XAI 1.0 опирался на постфактум объяснения и относительно простые метрики важности, то XAI 2.0 интегрирует объяснимость в сами модели, используя гибридные подходы и богатый математический аппарат для представления знаний. Нечеткие, интуиционистские, нейтрософические множества и особенно плитогенные множества сыграли важную роль в этом переходе. Они обеспечили язык для описания сложных решений машины в терминах, понятных человеку: через степени принадлежности, неопределенности, противоречия, множественные аспекты.

Плитогенный формализм, будучи обобщением всех предыдущих, открывает путь к XAI 3.0 – системам, способным не только объяснять, но и оценивать свои объяснения. За счет многомерного представления вывода и вычисления согласованности между ними, такие системы смогут выявлять свои внутренние противоречия и явно сообщать о них. Это новый уровень прозрачности, который потенциально преобразует взаимодействие человека с ИИ: доверие будет строиться на понимании не только *что* модель решила, но и *почему она уверена и насколько устойчиво* ее решение.

Можно ожидать, что дальнейшие исследования XAI 3.0 будут связаны с развитием удобных для пользователя методов визуализации многомерных объяснений, автоматического вывода текстовых отчетов о метаоценке вывода, стандартизации онтологий доверия и когнитивного конфликта. Однако уже сейчас ясно, что для достижения полной прозрачности ИИ-систем необходим именно симбиоз машинного обучения с нечеткой логикой и расширенными теориями множеств.

Благодарности. Авторы выражают особую благодарность компании «Yadro «КНС Групп» за предоставленное необходимого вычислительного оборудования. Авторы выражают свою признательность Муравьеву И.П. за иллюстрации.

Список литературы

- [Колмогоров, 1957] Колмогоров А.Н. О представлении непрерывных функций нескольких переменных в виде суперпозиций непрерывных функций одного переменного и сложения // Доклад АН СССР. – 1957. – Т. 114. – С. 953-956.
- [Atanassov, 1986] Atanassov K.T. Intuitionistic fuzzy sets // Fuzzy Sets and Systems. – 1986. – Vol. 20, No. 1. – P. 87-96. – doi: 10.1016/S0165-0114(86)80034-3.
- [Averkin et al., 2024] Averkin A.N., Volkov E.N., Yarushev S.A. Explainable artificial intelligence in deep learning neural netsbased digital images analysis // Journal of Computer and Systems Sciences International. – 2024. – Vol. 63. – P. 175-203. – doi: 10.1134/S1064230724700138.
- [Longo et al., 2024] Longo L. [et al.]. Explainable Artificial Intelligence (XAI) 2.0: A manifesto of open challenges and interdisciplinary research directions // Information Fusion. – 2024. – Vol. 106.
- [Smarandache, 2004] Smarandache F. Neutrosophic Set – A Generalization of the Intuitionistic Fuzzy Set // International Journal of Pure and Applied Mathematics. – 2004. – No. 24.
- [Smarandache, 2018] Smarandache F. Plithogenic Set, an Extension of Crisp, Fuzzy, Intuitionistic Fuzzy, and Neutrosophic Sets – Revisited // Neutrosophic Sets and Systems. – 2018. – Vol. 21. – P. 153-166.
- [Zadeh, 1965] Zadeh L.A. Fuzzy sets // Information and Control. – 1965. – Vol. 8, No. 3. – P. 338-353. – doi: 10.1016/S0019-9958(65)90241-X.
- [Ziming et al., 2024] Ziming L. [et al.]. KAN: Kolmogorov-Arnold Networks [Электронный ресурс] // arXiv. 2024. URL: <http://arxiv.org/abs/2404.19756> (дата обращения: 13.06.2025).

УДК 004.8

doi: 10.15622/rcai.2025.041

СОЗДАНИЕ ПРОТОТИПА РЕКУРСИВНОГО ОБЪЯСНИМОГО ИСКУССТВЕННОГО ИНТЕЛЛЕКТА НА ОСНОВЕ ПОВЕДЕНЧЕСКО-ЛОГИЧЕСКИХ МОДЕЛЕЙ¹

Ю.В. Трофимов (*ura_trofim@bk.ru*)
А.Н. Аверкин (*averkin2003@inbox.ru*)
И.П. Муравьев (*mip.22@uni-dubna.ru*)
А.К. Алексеев (*aak.24@uni-dubna.ru*)
Е.М. Кузнецов (*Kot454556@yandex.ru*)

Государственный университет «Дубна», Дубна

В работе представлен прототип рекурсивного объяснимого искусственного интеллекта, сочетающий поведенческо-логические модели мышления с современными нейросетевыми технологиями. Архитектура построена на когнитивной двухсистемной концепции, где Система 1 реализована как гибрид графовой и капсульной нейросети, обеспечивающих первичное интуитивное распознавание. Система 2 построена на базе дифференцируемого логического выводчика Neural Theorem Prover, реализующего формальный вывод в векторном пространстве признаков. Переключение между режимами осуществляется посредством Kolmogorov-Arnold Network (KAN), динамически управляющей распределением доверия между модулями. Дополнительную согласованность обеспечивает нечёткая логика, выступающая связующим звеном между нейросимвольными уровнями. Предложенная архитектура иллюстрирует принципы ХАИ 2.0 и демонстрирует высокую степень интерпретируемости решений без снижения точности. Система сопровождает каждый вывод внутренними логическими обоснованиями, обеспечивая прозрачность работы и устойчивость к неопределённости в сложных когнитивных задачах.

Ключевые слова: ХАИ, графовые нейросети, капсульные нейросети, Neural Theorem Prover. Kolmogorov–Arnold Network, нечёткая логика.

¹ Работа выполнена в рамках государственного задания Министерства науки и высшего образования Российской Федерации (тема № 124112200072-2).

Введение

Развитие объяснимого искусственного интеллекта (ХАИ) требует объединения высокой точности алгоритмов с их прозрачностью и интерпретируемостью. Ещё А.Н. Колмогоров и В.И. Арнольд заложили теоретический фундамент подобных подходов: их знаменитая теорема представления утверждает, что любая непрерывная функция многих переменных может быть представлена в виде суперпозиции конечного числа непрерывных функций одной. [Колмогоров, 1957]

$$f(x) = \sum_{q=0}^{2n} \Phi_q \left(\sum_{p=1}^n \Phi_{q,p}(x_p) \right).$$

Ученик Колмогорова В. Арнольд предложил конструктивный способ такой декомпозиции, благодаря чему данный результат получил название теоремы Колмогорова-Арнольда. Этот принцип лёг в основу современных ХАИ 2.0-подходов, нацеленных на встроенную интерпретируемость моделей. В манифесте ХАИ 2.0 отмечается, что создание надёжного ИИ требует сочетания субсимвольных нейросетевых методов с символьными алгоритмами объяснения [Longo et al., 2024]. Иными словами, необходимо изначально строить гибридные архитектуры, способные объяснять свои решения на внутреннем уровне, а не полагаться лишь на постфактум "расшифровку" чёрного ящика.

В первой вариации системы был разработан прототип системы ХАИ, сочетающий глубокие нейросети с нечеткой логикой, для решения задачи медицинской диагностики. [Трофимов, 2025] В частности, для выявления коронарных стенозов использовалась многоэтапная архитектура: свёрточная сеть ResNet-34 выделяла диагностически значимые кадры, модифицированная U-Net сегментировала коронарные сосуды с постобработкой CRF (коэф. Dice≈0,84; IoU≈0,78), а результаты дополнялись ХАИ-визуализацией Grad-CAM, LIME и Score-CAM. Для "прозрачности" заключения выходы нейросети были преобразованы в форму понятийных правил с помощью нейро-нечеткого модуля ANFIS. Такой гибридный подход продемонстрировал высокую точность диагностики (точность выявления стеноза > 90% на уровне кадра) и повысил доверие врачей за счёт интеграции механизмов объяснимости. Более того, включение нечётко-логических правил придало системе надёжность в сложных случаях: граничные ситуации, где нейросеть могла ошибиться из-за низкого контраста изображения, автоматически помечались как неопределённые или корректировались на основе базы знаний. Данный пример иллюстрирует концепцию ХАИ 2.0 на практике: объединение глубокой нейросети и логических правил позволило получить интерпретируемое решение без потери качества.

Данная работа развивает эти идеи и предлагает новую архитектуру рекурсивного объяснимого ИИ на основе поведенческо-логических моделей мышления. В основе – когнитивная двухсистемная теория Д. Канемана о Системе 1 и Системе 2 (интуитивные и аналитические процессы), дополненная современными нейросетевыми технологиями (графовые сети, капсульные сети) и механизмами для переключения режимов (Kolmogorov-Arnold Networks) с нечётко-логической интеграцией.

Двухсистемная архитектура интеллектуальной среды

Согласно Канеману, человеческое поведение определяется взаимодействием двух типов процессов: быстрой интуиции (Система 1) и медленно-го логического анализа (Система 2) [Kahneman, 2011]. В прикладном ИИ это находит отражение в гибридных архитектурах, где модуль Системы 1 реализован в виде глубокой нейросети – "черного ящика", мгновенно выдающей ответ, а Системы 2 – в виде более медленного рассуждающего модуля, который подключается при необходимости для углубленной проверки и объяснения решения. Эффективная когнитивная архитектура должна динамически переключаться между этими режимами, подобно тому как мозг человека задействует либо автоматизм, либо сознательный контроль в зависимости от сложности ситуации. В рассматриваемой системе данная парадигма реализована через связку нейросетевого компонента Системы 1 и логического компонента Системы 2, между которыми встроен интеллектуальный переключатель. Нечёткая логика при этом служит "мягким мостом", обеспечивая плавное взаимодействие подсистем. На рис. 1 изображен полный алгоритм работы системы.

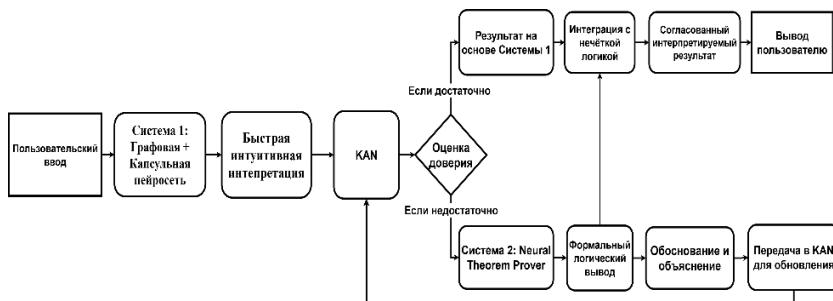


Рис. 1. Алгоритм работы системы

Система 1 – гибридная на основе графовой и капсульной нейросети

В роли Системы 1 выступает модуль глубокого обучения, отвечающий за быстрое "интуитивное" распознавание. Он реализован как гибридная глубокая модель, сочетающая графовую нейросеть с механизмом внима-

ния (Graph Neural Network) [Graph Neural Network, 2008] и капсульную сеть (CapsNet) [Sabour, 2017] в последовательной иерархической связке. Этот модуль отвечает за быстрое, "интуитивное" распознавание входных данных. Классическим выбором для Системы 1 являются сверточные нейросети, однако для более сложных и произвольных по структуре данных перспективнее применение графовых нейросетей. GNN способны эффективно работать с произвольными структурами связей, интегрируя информацию о взаимоотношениях между элементами входных данных. В отличие от данных на регулярных решётках, где связи между элементами фиксированы, графовое представление допускает моделирование не-евклидовых структур данных – онтологий знаний.

Особый подкласс графовых нейросетей с механизмом внимания дополнительно выделяет наиболее значимые связи в графе, присваивая каждому ребру обучаемый коэффициент значимости. Это позволяет Системе 1 фокусировать "внимание" на ключевых фрагментах входной структуры, подавляя шумовые или малосущественные связи. Фактически GNN на первом этапе выполняет роль интуитивного распознавания шаблонов: за один-два шага распространения активности по графу модель вычленивает главные характеристики входного сигнала и формирует его компактное скрытое представление. Механизм внимания облегчает интерпретацию результатов – можно визуализировать, на какие именно связи графа модель опиралась при распознавании в наибольшей степени. В результате после прохождения через GNN Система 1 выдаёт осмысленное векторное представление наблюдения, дополненное сопутствующей информацией.

Для повышения точности, на следующем этапе Система 1 использует капсульную сеть. CapsNet состоит из вычислительных "капсул", каждая из которых представляет собой группу нейронов, совместно кодирующих все параметры обнаруженного примитива или признака объекта (положение, масштаб и ориентацию отдельного сегмента). Каждая капсула более высокого уровня получает на вход вектор выходных сигналов от нескольких капсул низшего уровня, благодаря чему в представлении явно сохраняется иерархическая структура типа "часть-целое". Так, при обработке изображений капсульная сеть сначала выявляет низкоуровневые капсулы, соответствующие элементарным деталям (границам, углам), затем объединяет их в капсулы более высокого уровня, отображающие крупные фрагменты или части объекта, и в конечном итоге формирует капсулу верхнего уровня, представляющую целостный образ.

Подобное многоуровневое представление обеспечивает две ключевые выгоды с точки зрения объяснимости результатов. Во-первых, оно более интерпретируемо: можно проследить, какие именно составные части образа распознаны моделью и как эти части организованы между собой в рамках целого. Во-вторых, CapsNet обладает устойчивостью к аномаль-

ным конфигурациям признаков: модель способна правильно распознать объект, даже если его составные части расположены в необычных или нетипичных взаимных позициях. Кроме того, капсульная архитектура характеризуется свойством согласованности преобразований: при трансформациях входных данных активации капсул изменяются предсказуемо и пропорционально произведенной трансформации, вместо потери существенной информации, как это происходит при использовании операций пулинга в классических свёрточных сетях. В данной архитектуре капсульная сеть интегрирована сразу после графовой и необходима для повышения точности модели.

Система 1 представляет собой последовательное сочетание GNN и CapsNet, соединяя интуитивное шаблонное распознавание с итеративным уточнением и проверкой структурной согласованности. Выходные данные Системы 1 – компактное векторное описание наблюдения, подкреплённое картой значимых связей (внимания) и согласованной иерархией выделенных признаков.

Система 2 – модуль логического рассуждения на основе Neural Theorem Prover

В роли Системы 2 выступает модуль логического вывода, реализованный на основе подхода Neural Theorem Prover (NTP). NTP был предложен как дифференцируемый аналог классического выводчика Prolog, выполняющего backward chaining – пошаговый обратный вывод с использованием правил базы знаний. [Rocktäschel et al., 2017] В NTP классическая дискретная унификация терминов заменена на вычисление сходства между их векторными представлениями, то есть логический вывод реализуется как непрерывная дифференцируемая процедура на подсимвольных репрезентации. Такой подход позволяет комбинировать нейросетевое представление знаний с формальной логикой, превращая процесс доказательства фактов в end-to-end обучаемую нейросетевую модель. Данный подход был применён и успешно протестирован на широком спектре задач [Концепция иерархически организованных..., 2025].

Благодаря градиентному обучению NTP осваивает ряд важных возможностей логического вывода. Во-первых, модель обучается располагать векторные эмбединги семантически схожих символов близко друг к другу, что реализует механизм "нечёткой" унификации – способность сопоставлять неидентичные, но похожие объекты. Во-вторых, NTP использует такое сходство для многошагового доказательства новых фактов на основе неполной базы знаний, распространяя вывод по цепочке правил. В-третьих, модель способна автоматически индуцировать новые логические правила на основе данных, обнаруживая скрытые закономерности. Наконец, NTP может применять как априорно заданные человеком правила, так и самостоятельно выведенные, комбинируя их для получения

сложных выводов. Заметим, что использование явных правил базы знаний гарантирует логическую корректность рассуждений: система 2 выводит новые утверждения только путём применения допустимых логических трансформаций к уже известным фактам. Более того, за счёт работы в векторном пространстве признаков даже одно заданное правило может быть обобщено на множество конкретных ситуаций с похожими объектами, что повышает эффективность и универсальность логического модуля.

Ключевым достоинством NTP как Системы 2 является высокая степень объяснимости выводов. Каждый факт доказывается через явную последовательность шагов – дерево доказательства, состоящее из применённых правил и найденных совпадений. Такая цепочка служит естественным объяснением полученного результата для пользователя или разработчика системы. NTP способен в процессе обучения выявлять латентные логические зависимости, которые затем могут быть декодированы в правила легкочитаемые человеком. Таким образом, помимо самого ответа система генерирует понятные логические обоснования, демонстрируя на каких правилах и фактах основывается её вывод. Это существенно повышает доверие к модели и прозрачность её работы в контексте объяснимого ИИ.

Модуль NTP органично сочетается с Системой 1. Поскольку NTP оперирует распределёнными векторными представлениями символов, он напрямую потребляет на вход скрытые признаки и отношения, сформированные глубокой моделью первого уровня. Другими словами, выход Системы 1 поступает в NTP как база фактов для рассуждения. Благодаря полному дифференцированному описанию, обе системы могут обучаться совместно, образуя сквозной конвейер нейросимвольного ИИ. Быстрые интуитивные выводы Системы 1 дополняются строгой логической проверкой в Системе 2. В результате интеграции систем итоговое решение основывается не только на статистических шаблонах, но и на формальных знаниях, что обеспечивает и высокую точность, и интерпретируемость.

Стоит заметить, что прямое применение NTP сопряжено с комбинаторным ростом числа возможных путей доказательства, из-за чего наивная реализация плохо масштабируется на большие базы знаний. Для решения этой проблемы был разработан Greedy NTP (GNTP), который при доказательстве ограничивает поиск лишь топ-k наиболее перспективными фактами и правилами на каждом шаге. [Minervini et al., 2018] Такой жадный отбор значительно снижает вычислительную сложность, позволяя успешно применять нейронное доказательство теорем на реальных датасетах.

Kolmogorov-Arnold Network как интеллектуальный переключатель

Для координации взаимодействия между системами используется специальный модуль, в основе которого лежит Kolmogorov-Arnold Network (KAN) [Ziming et al., 2024]. Его роль – это слежение за ходом решения

задачи и выяснение, достаточно ли быстрого ответа или требуется углубленный анализ. В каждый момент KAN получает на вход показатели сложности и неопределённости текущей ситуации и вырабатывает сигнал переключения режима. Если задача простая и Система 1 уверенно справляется, переключатель оставляет основное решение без изменений. Если же входные данные сложны или нейросетевая модель не уверена в ответе, KAN активирует Систему 2. Формально выход переключателя – это коэффициент, лежащий в интервале от 0 до 1, определяющий вклад Системы 2 в итоговый вывод. В простейшем случае можно реализовать двоичный выбор: $g = 0$ допускает только ответ Системы 1, а $g = 1$ запускает полный логический вывод Системы 2. Однако более гибким является мягкое смешивание: итог, вычисляется как $y = g(x) \times y_{s2} + (1 - g(x)) \times y_{s1}$, где y_{s1} – ответ нейросети, а y_{s2} – ответ логического модуля. KAN особенно удобен для задания такой функции, поскольку его обучаемые одномерные элементы способны реализовывать пороговые или сигмоидоподобные зависимости от признаков сложности. По сути это реализует правило типа: "если доверие сети падает ниже X%, то подключить Систему 2", только не жёстко заданное, а выученное на данных и оптимально адаптированное под задачу. Аналогично по семантическим признакам входа KAN может давать системе "сигнал внимания": если обнаружено взаимодействие нескольких объектов, система переходит в режим вдумчивого, пошагового анализа. В итоге достигается баланс скорости и точности: глубокое рассуждение привлекается только при необходимости, что экономит ресурсы и одновременно повышает надёжность вывода.

Интерпретируемость переключателя

Важное достоинство KAN – его встроенная прозрачность. В классической нейросети влияние отдельных признаков скрыто в массе численных весов, и для оценки важности приходится применять косвенные методы. В KAN зависимость выхода от каждого признака задана явной функцией – графиком $\phi(x)$ на одном из ребер сети. Проанализировав форму этой функции, исследователь может напрямую оценить, как данный входной фактор влияет на результат. Сеть KAN предоставляет интерпретируемые параметры – кривые зависимостей, которые можно изучать подобно правилам. В контексте переключения режимов KAN позволяет понять, почему система решила привлечь или не привлекать модуль глубокого рассуждения. Иначе говоря, достигается мета-объяснимость: объяснимым становится не только базовое решение, выход Системы 1, но и факт его проверки. В результатах работы система может обосновать: "нейросеть дала неуверенный ответ, поэтому включён логический модуль, который, проанализировав признаки А, В, С, скорректировал вывод...". Такая многоуровневая прозрачность выгодно отличает предлагаемый подход от обычного post hoc XAI.

Ограничения KAN к ряду типов данных

Практическое применение KAN сопряжено с рядом ограничений, в задачах, характеризующихся высокой стохастичностью, структурной дискретностью или контекстной зависимостью.

KAN слабо устойчивы к входным данным, содержащим выраженные высокочастотные компоненты, неструктурированный шум или импульсные артефакты. Поскольку базовые элементы KAN реализуют гладкие одномерные аппроксиматоры, наличие нерегулярных возмущений приводит либо к переобучению, либо к агрессивному сглаживанию входной информации. Особенно это проявляется в биомедицинских задачах (анализ ЭКГ или МЭГ), где шумовые искажения носят негауссовский и высокоамплитудный характер.

Базовая архитектура KAN не предусматривает нативную обработку дискретных входных переменных. Классы, представленные в виде бинарных или категориальных признаков, требуют предварительного кодирования. Отсутствие встроенного механизма дискретизации ограничивает применение KAN в задачах, включающих логику принятия решений на основании номинативных категорий, таких как медицинские диагнозы, социодемографические характеристики и т.п.

Нечёткая логика как "мягкий мост" между модулями

Для интеграции результатов Системы 1 и Системы 2 используется нечёткая логика. Прежде всего, нечёткие продукционные правила позволяют избежать жёсткого разделения на случаи – условия правил могут выполняться с некоторой степенью принадлежности, а не только истина или ложь. Это соответствует характеру человеческих рассуждений, где многие понятия имеют размытие границ. В нашем прототипе выходы глубокой модели конвертируются в термины высокоуровневых признаков через лингвистические переменные и функции принадлежности. Например, числовое значение длины объекта превращается в понятие "большой", степень проявления признака – в "высокий" и т.д. Далее на основе обучающей выборки нейро-нечёткий модуль (тип ANFIS) автоматически извлекает базу правил. Такие правила привязаны к понятным характеристикам задачи. В ранее описанной системе диагностики они имели вид: "ЕСЛИ протяжённость стеноза большая И контрастность поражённого сегмента низкая, ТО степень стеноза значительная" [Трофимов, 2025]. Каждое правило имеет числовую степень активации вычисляемую на основе выхода нейросети. Мягкая интеграция заключается в следующем: даже если Система 2 не включается полностью, её нечёткие правила всё равно могут частично скорректировать или обогатить вывод Системы 1. В рамках формулы смешивания это соответствует случаю $0 \leq g < 1$. Тогда финальный ответ получается, как быстрый нейросетевой прогноз, дополненный

логическим заключением, взвешенным по степени уверенности. Такой подход реализует идею – мягкого переключения, исключающего резкие скачки в поведении системы. Кроме того, нечётко-логический модуль служит связующим звеном между субсимвольным представлением нейросети и символической базой знаний. Он сопоставляет распределенные признаки с понятиями из онтологии, активируя соответствующие узлы графа знаний. Далее по графу знаний распространяется активность: если набор признаков соответствует некоторой концепции, она становится гипотезой решения; продукционные правила проверяют целостность этой гипотезы, отсекая противоречивые сочетания. В результате высокоуровневый модуль формирует объяснимое заключение в терминах предметной области – фактически, в читаемой для человека логике, подтверждённой данными. Такой вывод может сопровождаться указанием: какие признаки выявлены нейросетью и как они связаны с выводом через базу правил. Тем самым Система 2 становится "гарантом и пояснителем" Системы 1. Интеграция через нечёткую логику обеспечивает требуемую согласованность между подсистемами и понятность рассуждений для эксперта.

Такой двухсистемный подход был реализован с помощью альтернативных моделей в работах [Автоматизация анализа рентгеновских снимков..., 2025], [Кузнецов, 2025], посвящённых комплексному анализу рентгеновских снимков грудной клетки.

Рекурсивность алгоритма

Рекурсивность алгоритма строится вокруг KAN-переключателя, определяющего качество объяснений и заключений Системы 1. При удовлетворительном результате работы первой системы, переключатель «пропускает» на выход результат работы данной системы, но как только результаты работы падают и становятся некорректными, KAN-переключатель запускает Систему 2 для генерации новых правил и корректировки Системы 1. Система 1 проходит валидацию с новыми корректировками, и если система улучшается, то изменения вносятся в базу знаний. После применения корректировок, Система 1 перезапускается и происходит повторная оценка результатов. Таким образом, данный сегмент алгоритма рекурсивно дообучает Систему 1 за счет Системы 2, до тех пор, пока KAN-Переключатель не пропустит результат работы Системы 1. Фрагмент алгоритма представлен на рис. 2, где KAN-переключатель принимает оценку сложности n и оценку неопределенности m , а $const_1$, $const_2$ – допустимые значения для данных оценок соответственно. Такой подход позволяет реализовать адаптивность на уровне архитектуры ко внешним условиям и данным, что требует манифест XAI 2.0.

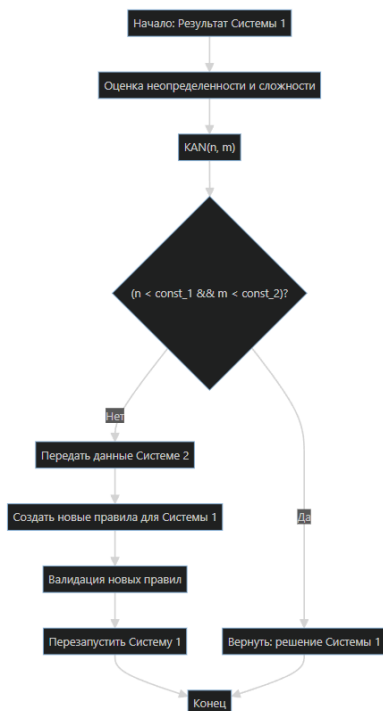


Рис. 2. Фрагмент алгоритма

Заключение

В работе была представлена архитектура прототипа рекурсивного объяснимого ИИ, сочетающая поведенческо-логические модели с современными нейросетевыми методами. Показано, что двухсистемный подход в сочетании с "переключателем" KAN и нечёткой логикой позволяет достичь одновременно высокой точности и интерпретируемости вывода. Система 1 обеспечивает быстрое распознавание образов и отношений, Система 2 вносит смысловой контроль и объяснение решений. KAN-модуль гибко управляет их взаимодействием, включая "медленное мышление" лишь тогда, когда это действительно нужно, а нечёткие правила служат мягким связующим звеном, переводя числовые активации в понятия и выводы на естественном языке.

Полученный прототип соответствует концепции XAI 2.0. Такая система не только выдаёт прогноз, но и сопровождает его развернутым самообъяснением, понятным для пользователя. В перспективе разработка подобных рекурсивных объяснимых систем открывает путь к более дове-

ренному ИИ, которому можно поручать задачи в критически важных областях без опасений непрозрачности решений. Кроме того, заложенные идеи могут быть развиты в сторону интеграции с нейросетями нового поколения. В частности, комбинация КАН-переключателей с большими языковыми моделями позволит строить саморефлексирующие агенты. Модель не только решает задачу, но и динамически формирует объяснение своих шагов, приближаясь к уровню осознанного человеческого рассуждения. Таким образом, подход, основанный на теоретических принципах Колмогорова и Арнольда, прокладывает новую траекторию к созданию гибридного ИИ, способного учиться, рассуждать и объяснять свои решения в едином интеллектуальном цикле.

Список литературы

- [Автоматизация анализа рентгеновских снимков..., 2025] Беляев М.И., Аверкин А.Н., Трофимов Ю.В., Шевченко А.В., Муравьев И.П. Автоматизация анализа рентгеновских снимков грудной клетки с использованием методов глубокого обучения и объяснимого искусственного интеллекта // Системный анализ в науке и образовании. – 2025. – № 2. – С. 32-41.
- [Колмогоров, 1957] Колмогоров А.Н. О представлении непрерывных функций нескольких переменных в виде суперпозиций непрерывных функций одного переменного и сложения // Доклад АН СССР. – 1957. – Т. 114. – С. 953-956.
- [Концепция иерархически организованных..., 2025] Трофимов Ю.В., Шевченко А.В., Аверкин А.Н., Муравьев И.А., Кузнецов Е.М. Концепция иерархически организованных объяснимых интеллектуальных систем: синтез глубоких нейронных сетей, нечеткой логики и инкрементального обучения в медицинской диагностике // VI Международная конференция по нейронным сетям и нейротехнологиям (NeuroNT) (Санкт-Петербург, 4-5 июнь 2025 г.): Труды конференции. – Санкт-Петербург: IEEE, 2025. – С. 14-17. – doi: 10.1109/NeuroNT66873.2025.11049976.
- [Кузнецов, 2025] Кузнецов Е.М., Трофимов Ю.В., Муравьев И.П. Объяснимый инкрементный подход в диагностике пневмонии: от свёрточных нейросетей до генеративных текстовых заключений / (науч. рук. Аверкин А.Н.) // XIV Конгресс молодых ученых ИТМО (НИУ ИТМО, 07-11 апрель, 2025 г.): Сборник тезисов докладов конгресса молодых ученых. [Электронное издание]. – URL: <https://kmu.itmo.ru/digests/article/15266> (дата обращения: 14.06.2025).
- [Трофимов, 2025] Трофимов Ю.В. [и др.]. Нечёткие продукционные правила и нейросети глубокого обучения: объяснимый искусственный интеллект 2.0 для диагностики коронарных стенозов // Системный анализ в науке и образовании. – 2025. – № 2. – С. 73-82.
- [Graph Neural Network, 2008] Scarselli F., Gori M., Tsoi A.C., Hagenbuchner M., Monfardini G. The graph neural network model // IEEE Transactions on Neural Networks. – 2008. – Vol. 20, No. 1. – P. 61-80. – doi: 10.1109/TNN.2008.2005605.
- [Kahneman, 2011] Kahneman D. Thinking, Fast and Slow. – New York: Farrar, Straus and Giroux, 2011. – 499 p.

- [Longo et al., 2024] Longo L. [et al.]. Explainable Artificial Intelligence (XAI) 2.0: A manifesto of open challenges and interdisciplinary research directions // Information Fusion. – 2024. – Vol. 106.
- [Minervini et al., 2018] Minervini P. [et al.]. Towards Neural Theorem Proving at Scale [Электронный ресурс] // arXiv. 2018. – URL: <https://arxiv.org/abs/1807.08204> (дата обращения: 14.06.2025).
- [Rocktäschel et al., 2017] Rocktäschel T., Riedel S. End-to-End Differentiable Proving [Электронный ресурс] // arXiv. 2017. – URL: <https://arxiv.org/abs/1705.11040> (дата обращения: 14.06.2025).
- [Sabour, 2017] Sabour S., Frosst N., Hinton G.E. Dynamic routing between capsules // Advances in Neural Information Processing Systems 30 (Long Beach, 4–9 Dec. 2017 г.): Труды конференции. В 1-м томе. Т. 1. – Нью-Йорк: Curran Associates, Inc., 2017. – С. 3856-3866. – doi: 10.48550/arXiv.1710.09829.
- [Ziming et al., 2024] Ziming L. [et al.]. KAN: Kolmogorov-Arnold Networks [Электронный ресурс] // arXiv. 2024. – URL: <http://arxiv.org/abs/2404.19756> (дата обращения: 14.06.2025).

УДК 004.827

doi: 10.15622/rcai.2025.042

О ЧИСЛОВЫХ ХАРАКТЕРИСТИКАХ НЕОПРЕДЕЛЕННОСТИ В КОНЕЧНЫХ РАЗМЫТЫХ МОДЕЛЯХ

Г.Э. Яхьяева (*gul_nara@mail.ru*)

О.Д. Пальчунова (*o.palchunova@g.nsu.ru*)

Новосибирский государственный университет, Новосибирск

Теория размытых моделей представляет собой альтернативный способ формализации неточности и/или неполноты знаний по сравнению с подходом, основанным на нечетких множествах, предложенным Лотфи Заде. При использовании размытых моделей для формализации знаний о предметных областях возникает необходимость количественно описать степень неопределенности и размытости этих моделей. В статье были рассмотрены различные количественные характеристики, учитывающие неопределенность конечных размытых моделей. В работе представлены различные методы формализации степени неопределенности размытой модели: вероятностный подход, который учитывает степень случайности события в предметной области; нечеткий подход, связанный с неопределенностью понятий в этой области; и объектный подход, описывающий неопределенность на уровне объектов. Также анализируется коэффициент сепарабельности размытой модели, показывающий, на сколько независимых подмоделей можно разбить данную размытую модель.

Ключевые слова: размытая модель, энтропия, коэффициент сепарабельности.

Введение

В середине XX века американский математик Клод Шеннон ввел понятие информационной энтропии, которая интерпретируется как мера неопределенности, связанной с возможными символами или исходами [Shannon, 1945]. Это мера неопределенности или среднего количества информации, связанного с набором вероятностей в системе.

В 1972 году Альфредо Де Лука и Сеттимо Термини впервые предложили рассматривать энтропию нечеткого множества [De Luca et al., 1972] – меру неопределённости в системах с нечётко определёнными парамет-

рами, расширяющая классическую концепцию Шеннона на нечеткие множества. На сегодняшний день существует порядка двадцати альтернативных определений энтропии нечеткого множества, отражающие различные аспекты неопределенности, заложенной в понятие нечеткого множества [Al-Sharhan et al., 2001].

Энтропия имеет важное значение для анализа и проектирования нечетких систем, поскольку она позволяет количественно оценивать уровень неопределенности и оптимизировать управление этими системами. Это, в свою очередь, способствует повышению надежности гипотез, выдвигаемых интеллектуальными системами [Кобринский и др., 2024]. В отличие от классической теории вероятностей, в нечетких системах энтропия отражает степень неопределенности, связанную с принадлежностью элементов к нечетким множествам.

Теория размытых моделей является альтернативным подходом к формализации неточности и/или неполноты знаний по отношению к формализации с помощью нечетких множеств, предложенных Лотфи Заде [Yakhyaeva et al., 2023a], [Пальчунов, 2022]. В отличие от функциональных нечетких логик, в которых не выполняются те или иные базовые логические тождества (такие, как, закон невозможности противоречия, закон исключённого третьего и др.), в теории размытых моделей сохраняются все теоретико-модельные тождества, то есть она является консервативным расширением классической теории моделей. Это с одной стороны решает проблему логических противоречий нечетких логик (в стиле Л.Заде), а с другой стороны даёт возможность одновременной работы как с классическими, так и с нечеткими моделями.

При формализации знаний о предметных областях с помощью размытых моделей также встает вопрос о количественном описании степени неопределенности/размытости этих моделей. Важно различать два основных типа неопределенности: вероятностную (стохастическую) и нечеткую (лингвистическую). Вероятностная неопределенность связана со случайностью возникновения событий в предметной области. Описание количественных характеристик такой неопределенности в размытых моделях посвящен параграф 2 данной статьи.

Нечеткая неопределенность относится к неоднозначности, расплывчатости или отсутствию четких границ в определении самих понятий или множеств. Энтропии нечеткого множества, рассматриваемые в параграфе 3, направлены на количественную оценку этого типа неопределенностей в размытой модели.

При рассмотрении нечетких неопределённостей акцент делается на неопределенность понятий, но не объектов предметной области. Описание объектной (или семантической) неопределенности посвящен параграф 4 данной статьи.

Рассматривая события, происходящие в предметной области мы сталкиваемся с понятием независимых событий, т.е. когда исход одного из которых не влияет на вероятность наступления другого. Исходя из этого, размытая модель может быть разложена на произведение независимых друг от друга подмоделей. Описанию таких разложений и их количественной характеристики посвящен параграф 5 данной статьи.

1. Основные определения и обозначения

Рассмотрим некоторую предметную область Δ , описываемую языком (сигнатурой) σ . В данной работе (для упрощения изложения) мы будем рассматривать чисто предикатную сигнатуру, т.е. не содержащую функциональных символов и символов констант. Через $S(\sigma)$ будем обозначать множество предложений сигнатуры σ , а через $S_a(\sigma)$ – множество атомарных предложений сигнатуры σ .

Мы будем говорить об истинности формул на размытой модели \mathfrak{M} . Для того чтобы говорить не об истинности произвольных формул, а только об истинности предложений, будем обогащать сигнатуру σ новыми константами. Будем использовать сигнатуру $\sigma_A = \sigma \cup \{c_a \mid a \in A\}$, где $\{c_a \mid a \in A\} \cap \sigma = \emptyset$. При этом на \mathfrak{M} будет выполняться $c_a^{\mathfrak{M}} = a$.

Определение 1. [Яхьяева, 2025] *Тройку $\mathfrak{M} = \langle A, \sigma, \mu \rangle$ будем называть размытой моделью, если истинностная функция $\mu: S(\sigma_A) \rightarrow [0,1]$ является аддитивной нечеткой мерой, определенной на алгебре Линденбаума-Тарского $S(\sigma_A)/\sim$.*

Размытую модель \mathfrak{M} будем называть **точной** (или классической), если ее истинностная функция μ принимает только значения 0 или 1, т.е. $\mu: S(\sigma_A) \rightarrow \{0; 1\}$.

Обозначим через $K(A, \sigma)$ класс всех размытых моделей сигнатуры σ , определенных на множестве A и через $K_{cr}(A, \sigma)$ – класс всех точных моделей сигнатуры σ , определенных на множестве A . Очевидно, что $K_{cr}(A, \sigma) \subset K(A, \sigma)$.

Размытую модель \mathfrak{M} будем называть *конечной*, если она определена на конечном множестве A и описывается конечной сигнатурой σ , т.е. если множество атомарных предложений $S_a(\sigma_A)$ – конечно.

Заметим, что если $\|S_a(\sigma_A)\| = n$, то $\|K_{cr}(A, \sigma)\| = 2^n$. Однако, по Теореме о разложении (сформулированной и доказанной в следующем параграфе) мощность класса $K(A, \sigma)$ будет континуальной.

Данная статья посвящена изучению конечных размытых моделей.

2. Прецедентная энтропия размытой модели

Одним из подходов к описанию меры неопределенности размытой модели \mathfrak{U}_μ является явное описание класса прецедентов предметной области Δ , формализованных при помощи размытой модели \mathfrak{U}_μ . Для этого нам понадобится понятие взвешенной суммы.

Определение 2. Будем говорить, что размытая модель $\mathfrak{B}_\mu = \langle A, \sigma, \mu \rangle$ раскладывается во **взвешенную сумму** размытых моделей $\mathfrak{U}_{\mu_1} = \langle A, \sigma, \mu_1 \rangle, \dots, \mathfrak{U}_{\mu_n} = \langle A, \sigma, \mu_n \rangle$, если найдется такая последовательность чисел $\alpha_1, \dots, \alpha_n \in [0; 1]$, что

1. $\alpha_1 + \dots + \alpha_n = 1$;
2. Для любого предложения $\varphi \in S(\sigma_A)$ выполняется
$$\mu(\varphi) = \alpha_1 \cdot \mu_1(\varphi) + \dots + \alpha_n \cdot \mu_n(\varphi).$$

Будем записывать $\mathfrak{B}_\mu = \alpha_1 \cdot \mathfrak{U}_{\mu_1} + \dots + \alpha_n \cdot \mathfrak{U}_{\mu_n}$

Теорема 1 (о разложении). Любая конечная размытая модель раскладывается во взвешенную сумму точных моделей.

Доказательство. Рассмотрим конечную размытую модель $\mathfrak{U}_\mu = \langle A, \sigma, \mu \rangle$. Пусть $S_a(\sigma_A) = \{\varphi_1, \varphi_2, \dots, \varphi_n\}$. Для каждого предложения $\varphi_i \in S_a(\sigma_A)$ и для каждой алгебраической системы $\mathfrak{U} \in K_{cr}(A, \sigma)$ введем обозначения

$$\varphi_i^{\mathfrak{U}} = \begin{cases} \varphi_i, & \mathfrak{U} \models \varphi_i; \\ \neg\varphi_i, & \mathfrak{U} \not\models \varphi_i. \end{cases}$$

Определим отображение $v: K_{cr}(A, \sigma) \rightarrow [0; 1]$ следующим образом:

$$v(\mathfrak{U}) = \mu(\varphi_1^{\mathfrak{U}} \& \varphi_2^{\mathfrak{U}} \& \dots \& \varphi_n^{\mathfrak{U}}),$$

где $\mathfrak{U} \in K_{cr}(A, \sigma)$.

Покажем, что модель \mathfrak{U}_μ раскладывается во взвешенную сумму

$$\mathfrak{U}_\mu = \sum_{\mathfrak{U} \in K_{cr}(A, \sigma)} v(\mathfrak{U}) \cdot \mathfrak{U}.$$

Для этого нам необходимо проверить свойства из Определения 2.

(1) В силу аддитивности нечеткой меры μ получим:

$$\begin{aligned} \sum_{\mathfrak{U} \in K_{cr}(A, \sigma)} v(\mathfrak{U}) &= \sum_{\mathfrak{U} \in K_{cr}(A, \sigma)} \mu(\varphi_1^{\mathfrak{U}} \& \varphi_2^{\mathfrak{U}} \& \dots \& \varphi_n^{\mathfrak{U}}) = \\ &= \mu \left(\bigvee_{\mathfrak{U} \in K_{cr}(A, \sigma)} (\varphi_1^{\mathfrak{U}} \& \varphi_2^{\mathfrak{U}} \& \dots \& \varphi_n^{\mathfrak{U}}) \right) = \mu(\mathbf{1}) = 1. \end{aligned}$$

(2) Рассмотрим предложение $\varphi \in S(\sigma_A)$. Введем обозначение:

$$\Omega = \{\varphi_1^{\mathfrak{A}} \& \varphi_2^{\mathfrak{A}} \& \dots \& \varphi_n^{\mathfrak{A}} \mid \mathfrak{A} \in K_{cr}(A, \sigma)\}.$$

Так как модель \mathfrak{A}_μ конечная, то предложение φ представимо в виде СКНФ, содержащей все атомарные предложения сигнатуры σ_A , т.е. найдутся такие $\omega_1, \dots, \omega_k \in \Omega$, что

$$\varphi = \omega_1 \vee \dots \vee \omega_k.$$

В силу аддитивности меры μ получим

$$\mu(\varphi) = \mu(\omega_1) + \dots + \mu(\omega_k) = \nu(\mathfrak{A}_1) + \dots + \nu(\mathfrak{A}_k),$$

где $\mathfrak{A}_i \in K_{cr}(A, \sigma)$ и $\mathfrak{A}_i \models \omega_i$.

А так как модель \mathfrak{A}_i (для любого $i = \overline{1, k}$) является точной, то

$$\mu_i(\omega_i) = 1 \text{ и } \mu_i(\omega_j) = 0 \ (j \neq i).$$

Таким образом, $\mu(\varphi) = \nu(\mathfrak{A}_1)\mu_1(\omega_1) + \dots + \nu(\mathfrak{A}_k)\mu_k(\omega_k)$.

Откуда и следует выполнение свойства (2).

Теорема доказана.

Отображение $\nu: K_{cr}(A, \sigma) \rightarrow [0; 1]$, определенное выше, назовем **распределением**, соответствующем размытой модели \mathfrak{A}_μ .

Класс точных моделей

$$K_{\mathfrak{A}_\mu} = \{\mathfrak{A} \in K_{cr}(A, \sigma) \mid \nu(\mathfrak{A}) > 0\}$$

назовем **классом прецедентов** предметной области Δ , формализуемой посредством размытой модели \mathfrak{A}_μ .

Заметим, что чем меньше различных прецедентов имеет место быть в данной предметной области, тем меньше неопределенностей в ней возникает, и если предметная область содержит ровно один прецедент, то описывающая ее модель становится точной.

Таким образом, мы можем задать первую характеристику неопределенности размытой модели

Определение 3. Рассмотрим размытую модель \mathfrak{A}_μ и класс прецедентов $K_{\mathfrak{A}_\mu}$, соответствующий этой модели. **Прецедентной энтропией** размытой модели \mathfrak{A}_μ назовем величину, заданную следующим образом:

$$P(\mathfrak{A}_\mu) = \frac{\|K_{\mathfrak{A}_\mu}\|}{\|K_{cr}(A, \sigma)\|}.$$

Для любого прецедента $\mathfrak{A} \in K_{\mathfrak{A}_\mu}$ меру $\nu(\mathfrak{A})$ можно интерпретировать как "значимость" прецедента \mathfrak{A} для данной предметной области. Таким образом, если $\nu(\mathfrak{A})$ стремится к нулю, то прецедент \mathfrak{A} является "несущественным" для данной предметной области.

Рассматривая класс точных моделей как значения некоторой случайной величины, а распределение $\nu: K_{cr}(A, \sigma) \rightarrow [0; 1]$ как вероятностное распределение этой случайной величины, мы можем ввести понятие энтропии Шеннона для размытой модели.

Определение 4. Рассмотрим размытую модель \mathfrak{U}_μ и соответствующее ей распределение $\nu: K_{cr}(A, \sigma) \rightarrow [0; 1]$. *Энтропией Шеннона размытой модели \mathfrak{U}_μ назовем величину, заданную следующим образом:*

$$H(\mathfrak{U}_\mu) = - \sum_{\mathfrak{U} \in K_{\mathfrak{U}_\mu}} \nu(\mathfrak{U}) \log_2 \nu(\mathfrak{U}).$$

Очевидно, что чем больше значения прецедентной энтропии и энтропии Шеннона, тем больше неопределенности заложено в размытой модели \mathfrak{U}_μ . Модель \mathfrak{U}_μ имеет максимальную степень неопределенности, если описываемая ею предметная область содержит всевозможные точные модели как прецеденты и при этом все прецеденты равновероятны. В этом случае $P(\mathfrak{U}_\mu) = 1$ и $H(\mathfrak{U}_\mu) = \|S_a(\sigma_A)\|$.

3. Энтропии нечеткого множества

В классической теории моделей атомарная диаграмма модели – это некоторое подмножество множества $S_a(\sigma_A)$ всех атомарных предложений данной сигнатуры. Атомарная диаграмма модели описывает знания об истинности базовых (атомарных) понятий данной предметной области. Поскольку мы имеем дело с моделями, в которых все предложения имеют оценочную характеристику (нечеткую оценку), то под атомарной диаграммой размытой модели \mathfrak{U}_μ будем понимать нечеткое подмножество множества $S_a(\sigma_A)$, функция принадлежности которого совпадает с мерой μ , определяющей сигнатуру этой модели.

Множество упорядоченных пар $AD(\mathfrak{U}_\mu) = \{\langle \varphi, \mu(\varphi) \rangle \mid \varphi \in S_a(\sigma_A)\}$ будем называть *атомарной диаграммой* размытой модели \mathfrak{U}_μ .

В работе [Yakhyayeva, 2025] было показано, что атомарная диаграмма (в отличие от классического случая) не описывает однозначно размытую модель. С другой стороны, означивание всех бескванторных предложений данной сигнатуры является избыточным условием для однозначного описания размытой модели.

Пусть $S_a(\sigma_A) = \{\varphi_1, \varphi_2, \dots, \varphi_n\}$ Обозначим через $S_p(\sigma_A)$ множество всех позитивных конъюнктов сигнатуры σ_A , т.е. $S_p(\sigma_A) = \{\varphi \mid \exists k = \overline{1, n}: \varphi = \varphi_{i_1} \& \dots \& \varphi_{i_k}\}$. Тогда множество упорядоченных пар

$$PD(\mathfrak{U}_\mu) = \{\langle \varphi, \mu(\varphi) \rangle \mid \varphi \in S_p(\sigma_A)\}$$

будем называть *позитивной диаграммой* размытой модели \mathfrak{U}_μ .

Теорема 2. [Yakhyaeva, 2025] Любая размытая модель однозначно задается своей позитивной диаграммой.

Таким образом, рассматривая различные нечеткие энтропии позитивной диаграммы размытой модели, мы можем ввести дополнительные характеристики неопределенности размытой модели.

В качестве примера приведем формулировки, в контексте размытых моделей, энтропий Де Луки-Термини [Sheikh et. al., 2018] и индекс нечеткости Кауфмана [Kral, 2005].

Определение 5. Рассмотрим размытую модель \mathfrak{A}_μ и соответствующую ей позитивную диаграмму $PD(\mathfrak{A}_\mu)$. Энтропией Де Луки-Термини размытой модели \mathfrak{A}_μ назовем величину, заданную следующим образом:

$$L(\mathfrak{A}_\mu) = - \sum_{\varphi \in PD(\mathfrak{A}_\mu)} [\mu(\varphi) \log_2 \mu(\varphi) + (1 - \mu(\varphi)) \log_2 (1 - \mu(\varphi))].$$

Энтропия Де Луки и Термини более тесно связана с теорией информации и акцентирует внимание на неопределенности, подобно энтропии Шеннона. Также, как и в энтропии Шеннона, работает соглашение, что $0 \cdot \log_2 0 = 0$. Исходя из этого мы получаем, что $L(\mathfrak{A}_\mu)$ принимает нулевое значение тогда и только тогда, когда позитивная диаграмма размытой модели $PD(\mathfrak{A}_\mu)$ является четким множеством. А это, в свою очередь, является необходимым и достаточным условием того, что модель \mathfrak{A}_μ являлась точной.

Индекс нечеткости Кауфмана проще в вычислении и фокусируется на геометрической интерпретации нечеткости как расстояния до четкого множества. Индекс нечеткости измеряет, насколько нечеткое множество отличается от ближайшего четкого множества, основываясь на расстоянии (обычно евклидовом или Хэмминга).

Определение 6. Рассмотрим размытую модель \mathfrak{A}_μ и соответствующую ей позитивную диаграмму $PD(\mathfrak{A}_\mu)$. Индексом нечеткости Кауфмана размытой модели \mathfrak{A}_μ назовем величину, заданную следующим образом:

$$F(\mathfrak{A}_\mu) = \frac{1}{2^{n-1}} \sum_{\varphi \in PD(\mathfrak{A}_\mu)} \min(\mu(\varphi), 1 - \mu(\varphi)),$$

где $n = \|S_a(\sigma_A)\|$.

Не трудно проверить, что $F(\mathfrak{A}_\mu) = 0$ тогда и только тогда, когда модель \mathfrak{A}_μ является точной.

Предложение 1. Рассмотрим размытую модель $\mathfrak{A}_{\mu_A} = \langle A, \sigma, \mu_A \rangle$. Тогда

$$F(\mathfrak{U}_{\mu_A}) = 1 \Rightarrow P(\mathfrak{U}_{\mu_A}) = \frac{1}{2^{n-1}},$$

где $n = \|\mathcal{S}_a(\sigma_A)\|$.

Таким образом, если размытая модель характеризуется максимальным индексом нечеткости Кауфмана, то она имеет почти минимальную прецедентную энтропию, т.е. она раскладывается во взвешенную сумму всего лишь двух точных моделей. Однако эти модели образуют контрарную пару, т.е. максимально «удалены» друг от друга.

4. Объектная энтропия размытой модели

Во втором параграфе, рассматривая характеристики неопределенности размытой модели, мы рассматривали только количество прецедентов предметной области и их вероятностное распределение. Но мы не сравнивали прецеденты между собой. Однако, у нас может сложиться ситуация, когда имеются две различные размытые модели, обладающие одним и тем же количеством прецедентов и одним и тем же вероятностным распределением, заданном на классах этих прецедентов. Тогда прецедентные энтропии и энтропии Шеннона таких моделей будут равны.

Однако, если мы посмотрим на структуру этих прецедентов, то в одном случае они могут лишь не значительно отличаться друг от друга, а в другом случае, например, содержать контрарные пары прецедентов. Очевидно, что в этом случае степени неопределенности у них будут разные. И нам нужны величины, отражающие эти различия.

Определение 7. [Yakhyaeva, 2023b] Пусть $\mathfrak{U}_{\mu_A} = \langle A, \sigma, \mu_A \rangle$ и $\mathfrak{B}_{\mu_B} = \langle B, \sigma, \mu_B \rangle$ размытые модели одной сигнатуры σ . Будем говорить, что размытая модель \mathfrak{B}_{μ_B} является **подмоделью** размытой модели \mathfrak{U}_{μ_A} (обозначение $\mathfrak{B}_{\mu_B} \subseteq \mathfrak{U}_{\mu_A}$), если $B \subseteq A$ и для любого бескванторного предложения $\varphi \in S(\sigma_B)$ имеет место $\mu_B(\varphi) = \mu_A(\varphi)$.

Пусть $\mathfrak{U}_{\mu_A} = \langle A, \sigma, \mu_A \rangle$. Заметим, что так как в начале данной статьи мы договорились рассматривать чисто предикатную сигнатуру σ , то для любого подмножества $B \subseteq A$ возможно построить подмодель $\mathfrak{B}_{\mu_B} = \langle B, \sigma, \mu_B \rangle$.

Подмодель \mathfrak{B}_{μ_B} модели \mathfrak{U}_{μ_A} будем называть **ядерной**, если она является точной моделью, т.е. если $\mathfrak{B}_{\mu_B} \in K_{cr}(B, \sigma)$. Заметим, что у размытой модели может быть более одной ядерной подмодели. Модель, определенную на объединение носителей всех ядерных подмоделей, будем называть **ядром** размытой модели \mathfrak{U}_{μ_A} и обозначать через $Ker(\mathfrak{U}_{\mu_A})$. Если же размытая модель \mathfrak{U}_{μ_A} не имеет ядерных подмоделей, то будем говорить, что она обладает пустым ядром и записывать $Ker(\mathfrak{U}_{\mu_A}) = \emptyset$.

Подмодель \mathfrak{B}_{μ_B} будем называть **телом** размытой модели \mathfrak{A}_{μ_A} и обозначать через $Body(\mathfrak{A}_{\mu_A})$, если $Ker(\mathfrak{B}_{\mu_B}) = \emptyset$ и для любой модели \mathfrak{C}_{μ_C} такой, что $\mathfrak{B}_{\mu_B} \subseteq \mathfrak{C}_{\mu_C} \subseteq \mathfrak{A}_{\mu_A}$ выполняется условие: $Ker(\mathfrak{C}_{\mu_C}) \neq \emptyset$.

Заметим, что если модель \mathfrak{A}_{μ_A} является точной, то она совпадает со своим ядром. В этом случае $Body(\mathfrak{A}_{\mu_A}) = \emptyset$. Исходя из этого мы можем ввести еще одну характеристику неопределенности размытой модели.

Определение 8. Рассмотрим размытую модель $\mathfrak{A}_{\mu_A} = \langle A, \sigma, \mu_A \rangle$, обладающую телом $Body(\mathfrak{A}_{\mu_A}) = \mathfrak{B}_{\mu_B} = \langle B, \sigma, \mu_B \rangle$. **Объектной энтропией** размытой модели \mathfrak{A}_{μ_A} назовем величину, заданную следующим образом:

$$O(\mathfrak{A}_{\mu_A}) = \frac{\|B\|}{\|A\|}.$$

Таким образом, если размытая модель обладает пустым ядром, то она имеет максимальную объектную энтропию, равную 1. Если же модель является точной, то ее объектная энтропия равна 0.

Предложение 2. Рассмотрим размытую модель $\mathfrak{A}_{\mu_A} = \langle A, \sigma, \mu_A \rangle$. Тогда

$$P(\mathfrak{A}_{\mu_A}) = 1 \Rightarrow O(\mathfrak{A}_{\mu_A}) = 1.$$

Таким образом, данное предложение утверждает, что если размытая модель имеет максимальную прецедентную энтропию, то она будет иметь и максимальную объектную энтропию. Обратное утверждение будет не верно.

5. Коэффициент сепарабельности размытой модели

Одной из ключевых концепций теории вероятности и статистики является понятие независимости событий. Она позволяет проводить более точные и простые расчеты, так как можно использовать более простые формулы для нахождения совместной вероятности. Независимые события имеют важное значение в областях, где необходимо анализировать случайные процессы. Также независимость событий можно рассматривать в ключе решения проблемы сепарабельности квантовых систем [Яхьяева и др., 2024].

При вероятностном подходе к изучению свойств размытых моделей мы можем говорить о независимости различных событий, описываемых размытой моделью. Независимость событий мы будем формализовывать, используя понятие сепарабельного произведения моделей.

Определение 9. Пусть $\mathfrak{A}_{\mu_{A_1}} = \langle A_1, \sigma, \mu_{A_1} \rangle, \dots, \mathfrak{A}_{\mu_{A_n}} = \langle A_n, \sigma, \mu_{A_n} \rangle$ размытые модели одной сигнатуры σ . **Сепарабельным произведением** моделей $\mathfrak{A}_{\mu_{A_1}}, \dots, \mathfrak{A}_{\mu_{A_n}}$ будем называть класс размытых моделей $\mathfrak{C}_{\mu_C} \in K(\bigcup_{i=1}^n A_i, \sigma)$, для которых выполняются условия:

а) Для любого $i \in \overline{1, n}$ выполняется: $\mathfrak{U}_{\mu_{A_i}} \subseteq \mathfrak{C}_{\mu_C}$.

б) Для любых $i, j \in \overline{1, n}$ выполняется:

$$\mu_C(\varphi_i \& \varphi_j) = \mu_{A_i}(\varphi_i) \cdot \mu_{A_j}(\varphi_j),$$

где $\varphi_i \in S_p(\sigma_{A_i})$ и $\varphi_j \in S_p(\sigma_{A_j})$.

Заметим, что не для любых моделей существует сепарабельное произведение. Необходимым и достаточным условием существования сепарабельного произведения моделей $\mathfrak{U}_{\mu_{A_1}}, \dots, \mathfrak{U}_{\mu_{A_n}}$ является условие, что либо $\bigcap_{i=1}^n A_i = \emptyset$, либо существование общей подмодели с носителем $\bigcap_{i=1}^n A_i$.

Будем говорить, что модель \mathfrak{C}_{μ_C} раскладывается в сепарабельное произведение подмоделей $\mathfrak{U}_{\mu_{A_1}}, \dots, \mathfrak{U}_{\mu_{A_n}}$, если $\mathfrak{C}_{\mu_C} \in \mathfrak{U}_{\mu_{A_1}} \circledast \dots \circledast \mathfrak{U}_{\mu_{A_n}}$.

Модель, которая не раскладывается в сепарабельное произведение будем называть *запутанной* [Yakhyaeva et al., 2024]. В противном случае модель будем называть *сепарабельной* [Yakhyaeva, 2023c].

Предложение 3. Любая размытая модель \mathfrak{U}_{μ_A} с непустым ядром раскладывается в сепарабельное произведение своего ядра и тела, т.е.

$$\mathfrak{U}_{\mu_A} \in \text{Ker}(\mathfrak{U}_{\mu_A}) \circledast \text{Body}(\mathfrak{U}_{\mu_A}).$$

Заметим, что тело размытой модели не всегда раскладывается в сепарабельное произведение своих подмоделей.

Пусть $\mathfrak{C}_{\mu_C} \in \text{Ker}(\mathfrak{U}_{\mu_C}) \circledast \mathfrak{U}_{\mu_{A_1}} \circledast \dots \circledast \mathfrak{U}_{\mu_{A_n}}$. Данное разложение в сепарабельное произведение назовем *каноническим*, если модели $\mathfrak{U}_{\mu_{A_1}}, \dots, \mathfrak{U}_{\mu_{A_n}}$ являются запутанными. Если модель \mathfrak{C}_{μ_C} является моделью с пустым ядром, то ее каноническое разложение выглядит следующим образом: $\mathfrak{C}_{\mu_C} \in \mathfrak{U}_{\mu_{A_1}} \circledast \dots \circledast \mathfrak{U}_{\mu_{A_n}}$. Если же модель является запутанной, то ее каноническое разложение совпадает с самой моделью.

Теорема 3. Для любой размытой модели существует единственное каноническое разложение в сепарабельное произведение ее подмоделей.

Используя утверждение Теоремы 3, мы можем теперь ввести еще одну количественную характеристику неопределенности размытой модели.

Определение 10. Рассмотрим размытую модель $\mathfrak{U}_{\mu_A} = \langle A, \sigma, \mu_A \rangle$. Количество запутанных моделей, входящих в каноническое разложение модели \mathfrak{U}_{μ_A} в сепарабельное произведение назовем *коэффициентом сепарабельности* модели \mathfrak{U}_{μ_A} и будем обозначать через $\text{sep}(\mathfrak{U}_{\mu_A})$.

Рассмотрим размытую модель \mathfrak{U}_{μ_A} и ее прецедентную энтропию $P(\mathfrak{U}_{\mu_A}) = \frac{k}{2^n}$, где $n = \|S_a(\sigma_A)\|$ и $k \in \overline{1, 2^n}$. Не трудно понять, что $\text{sep}(\mathfrak{U}_{\mu_A}) = 0$ тогда и только тогда, когда $k = 1$, т.е. модель \mathfrak{U}_{μ_A} является

точной. Если же $\text{sep}(\mathcal{X}_{\mu_A}) = m$ ($m > 0$), то $k \geq 2^m$. Следовательно, если размытая модель является сепарабельной, то она формализуется как минимум четырьмя прецедентами.

Заключение

В статье были проанализированы различные количественные характеристики учета неопределенности конечных размытых моделей. Общей чертой всех этих характеристик является то, что они достигают своего минимального значения, когда модель точна, и увеличиваются с ростом неопределенности. Однако максимальные значения этих характеристик наблюдаются на различных размытых моделях. Это объясняется тем, что каждая из характеристик предоставляет свое уникальное понимание неопределенности, присущей размытой модели.

Например, прецедентная энтропия и энтропия Шеннона акцентируют внимание на вероятностных (статистических) аспектах размытой модели. В то время как энтропии нечетких множеств и объектная энтропия рассматривают семантический подход к описанию неопределенности размытой модели.

Особый интерес представляет коэффициент сепарабельности размытой модели, который указывает, на сколько независимых подмоделей можно разделить данную размытую модель. Определение этого коэффициента связано с семантикой размытой модели (т.е. с ее истинностной функцией), но также прослеживается и связь с вероятностным распределением прецедентов размытой модели.

В дальнейшем мы планируем более подробно изучить взаимосвязь статистических и семантических характеристик размытых моделей и описать закономерности этих взаимосвязей.

Список литературы

- [Кобринский и др., 2024] Кобринский Б.А., Николаев А.А. Вероятностное представление измеряемых признаков и их отражение в системах искусственного интеллекта // XII международная научно-практическая конференция «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2024) (Коломна, 14-17 мая 2024 г.). – Т. 2. – С. 86-93.
- [Пальчунов, 2022] Пальчунов Д.Е. Теория моделей предметных областей. I // Алгебра и логика. – 2022. – Т. 61, № 2. – С. 239-250. – Doi: 10.33048/alglog.2022.61.207.
- [Яхьяева и др., 2024] Яхьяева Г.Э., Пальчунова О.Д. О квантовой интерпретации теории нечетких моделей // XII международная научно-практическая конференция «Интегрированные модели и мягкие вычисления в искусственном интеллекте» (ИММВ-2024) (Коломна, 14-17 мая 2024 г.). – Т. 1. – С. 225-236.
- [Яхьяева, 2025] Яхьяева Г.Э. Классы нечетких моделей // Известия Иркутского государственного университета. Серия Математика. – 2025. – Т. 51. – С. 151-166. – Doi: 10.26516/1997-7670.2025.51.151

- [**Al-Sharhan et al., 2001**] Al-Sharhan S., Karray F., Gueaieb W., Basir O. Fuzzy entropy: A brief survey. In: 10th IEEE International Conference on Fuzzy Systems, Melbourne, Australia. 2001. – P. 1135-1139. – Doi: 10.1109/FUZZ.2001.1008855.
- [**De Luca et al., 1972**] De Luca A., Termini S. A definition of a nonprobabilistic entropy in the setting of fuzzy sets theory // Information and Control. – 1972. – Vol. 20(4). – P. 301-312. – Doi: 10.1016/S0019-9958(72)90199-4.
- [**Kral, 2005**] Kral P., Cardinality and Entropy of IF sets. Ninth // Int. Conf. on IFSs, Sofia, 7-8 May 2005, NIFS. – 2005. – Vol. 11. – P. 78-87.
- [**Shannon, 1945**] Shannon C.E. A Mathematical Theory of Communication // Bell System Technical Journal. – 1945. – Vol. 27 (3 and 4). – P. 379-423, 623-656. – Doi: 10.1002/j.1538-7305.1948.tb01338.x.
- [**Sheikh et al., 2018**] Sheikh B.Q. and Baig M.A.K., An Overview of Fuzzy Entropy-Some NonParametric Generalizations and Applications // Journal of Basic and Applied Engineering Research p-ISSN: 2350-0077; e-ISSN: 2350-0255; Vol. 5, Issue 6; October-December, 2018. – P. 496-499.
- [**Yakhyaeva et al., 2023a**] Yakhyaeva G.E., Palchunova O.D. Fuzzy Models as a Formalization of Expert's Evaluative Knowledge // Pattern Recognition and Image Analysis. – 2023. – Vol. 33(3). – P. 529-535. – Doi: 10.1134/S105466182303046X.
- [**Yakhyaeva, 2023b**] Yakhyaeva G.E. On the Local Coordination of Fuzzy Valuations // The Bulletin of Irkutsk State University. Series Mathematics. – 2023. – Vol. 46. – P. 130-144. – Doi: 10.26516/1997-7670.2023.46.130.
- [**Yakhyaeva, 2023c**] Yakhyaeva G.E. Separable Fuzzy Models // In: IEEE 16th International conference of actual problems of electronic instrument engineering (APEIE). – 2023. – P. 1480-1483. – Doi: 10.1109/APEIE59731.2023.10347792.
- [**Yakhyaeva et al., 2024**] Yakhyaeva, G.E., Palchunova, O.D. Entangled Fuzzy Models // In: IEEE International Multi-Conference on Engineering, Computer and Information Sciences, SIBIRCON 2024. – 2024. – P. 314-318. – Doi: 10.1109/SIBIRCON63777.2024.10758476.
- [**Yakhyaeva, 2025**] Yakhyaeva G.E., Semantic modelling of subject domains using precedent and blurry (fuzzy) models. Sirius Mathematical Journal. 2025 (в печати).

УДК 004.8

doi: 10.15622/rcai.2025.043

**КЛАССИФИКАЦИЯ СОСТОЯНИЯ ПТИЦЫ
НАПОЛЬНОГО СОДЕРЖАНИЯ НА ОСНОВЕ
ИСПОЛЬЗОВАНИЯ YOLO**

Д.Г. Буханов (*dgbukh@gmail.com*)^A

С.В. Черников (*serg4315@mail.ru*)^B

Ю.А. Кошлич (*koshlich@yandex.ru*)^A

^A Белгородский государственный технологический университет
им. В.Г. Шухова, Белгород

^B ООО «Энтерсофт», Белгород

Анализ состояния исследования в области автоматического определения функционального состояния птицы напольного содержания показал, что существующие подходы не учитывают специфику реальных условий производственных площадок выращивания птицы. Сложность размещения полноценных рабочих станций или удаленное подключение к камерам наблюдения в корпусе приводит к необходимости использования малопроизводительных или одноплатных компьютеров для анализа данных. Предложена модификация классификатора на основе архитектуры YOLO посредством вынесения предобработки изображения в отдельный этап, выполняемый единожды для каждого входного изображения. Проведены эксперименты классификации птицы напольного содержания, обнаружения падежа и определения наполненности кормушек. В результате эксперимен-

тов был получен прирост производительности на 3% и 12,5% для рабочей станции с GPU и одноплатного компьютера Raspberry 5 соответственно. По результатам экспериментов сделан вывод о применимости предлагаемого подхода на одноплатных или малопроизводительных компьютерах с целью повышения производительности систем анализа функционального состояния птицы.

Ключевые слова: анализ видеоизображений, YOLO, искусственные нейронные сети, машинное зрение.

Введение

Под анализом функционального состояния птицы подразумевается анализ нескольких аспектов: определение и отслеживание положения птицы, ее состояния здоровья, выявление падежа, расчет предполагаемой живой массы и измерение прочих связанных параметров [Supriyanto et al., 2023], [Neethirajan, 2022]. Несмотря на наличие множества векторов исследования в области анализа функционального состояния птицы все они основываются на применении технологий машинного зрения.

Подход решения такого рода задач состоит в применении камер наружного наблюдения с последующим анализом полученных изображений. В качестве анализатора или классификатора используются искусственные нейронные сети, в частности, сверточные.

Проведенный анализ состояния исследования продемонстрировал увеличение частоты применения сверточных нейронных сетей на основе архитектуры YOLO (You Only Look Once) [Redmon et al., 2016]. Так, нейронные сети на основе архитектуры YOLO применяются для обнаружения птицы напольного и клеточного содержания [Paramathma et al., 2024], [Chen et al., 2023], [Chemme et al., 2024], трекинга [Triyanto et al., 2023], [Mehdizadeh et al., 2024], [Siriani et al., 2022], подсчета птиц [Zhu et al., 2022], [Sun et al., 2023], [Wu et al., 2025], выявления падежа [Bist et al., 2025], [Syafaah et al., 2024], [Yang et al., 2024], отслеживания здоровья [Tong et al., 2023] и расчета предполагаемой живой массы [Sun et al., 2024], [Mahmoud et al., 2025].

В настоящее время исследования, проводимые в экспериментальных условиях, не учитывают специфику реальных условий производственных площадок птичного хозяйства. Корпуса, используемые при выращивании птицы напольного содержания (ПНС), обладают значительными размерами: от 1 тысячи квадратных метров на 20 тысяч голов. Для покрытия данной площади требуется от 10 камер наружного наблюдения, изображения с которых должны сниматься и анализироваться одновременно.

Также производственные корпуса зачастую не обладают свободными помещениями или пристройками, в которых могли бы быть размещены значительные серверные мощности для анализа изображений в режиме

реального времени, а расположение производственных площадок на удалении от населенных пунктов затрудняет удаленный доступ к камерам наблюдения.

Решением данных проблем является применение мини- и одноплатных компьютеров, чьи размеры позволяют установить их в подсобных помещениях производственных корпусов, а мощности достаточны для отложенного анализа изображений с заданным интервалом.

Данное исследование направлено на модификацию сверточных нейронных сетей на основе архитектуры YOLO с целью повышения их производительности при анализе функционального состояния ПНС и уменьшения интервала, требуемого для отложенного анализа изображений.

1. Классификация объектов видеоизображений при анализе состояния птицы напольного содержания

В исследовании предлагается модификация классификаторов на основе семейства YOLO с целью уменьшения затрат времени на этап предобработки изображений.

Для анализа функционального состояния птицы были разработаны 3 классификатора на основе семейства YOLO:

- классификатор для обнаружения ПНС;
- классификатор для обнаружения падежа;
- классификатор для определения наполненности кормушек.

Для каждого классификатора были составлены и размечены оригинальные наборы данных на основе трех тестовых туров полного жизненного цикла птицы, описанные в табл. 1.

Таблица 1

Набор экспериментальных данных

Классификатор	Классификатор для обнаружения птицы		Классификатор обнаружения падежа		Классификатор определения наполненности кормушек	
Количество кадров	157		103		99	
Количество объектов	Птица	10650	Птица	12200	Пустая	159
	Кормушка	1005	Падеж	145	Полная	138
	Поилка	930				

Результаты обучения классификатора обнаружения ПНС приведены на рис. 1.

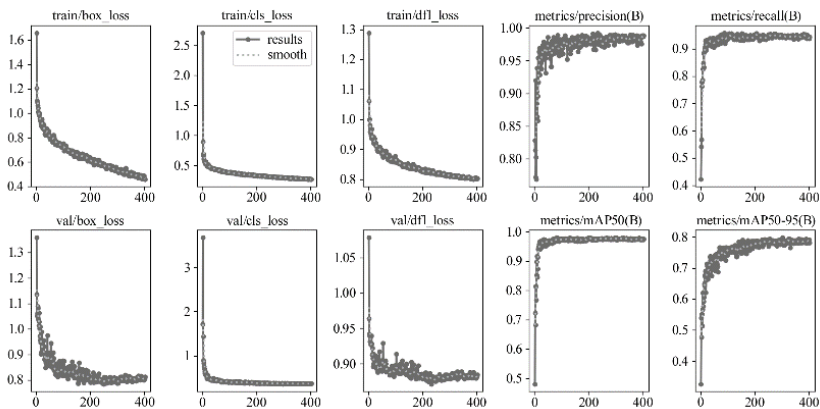


Рис. 1. Результаты обучения классификатора ПНС

Результаты обучения классификатора обнаружения птицы на кадре приведены на рис. 2.

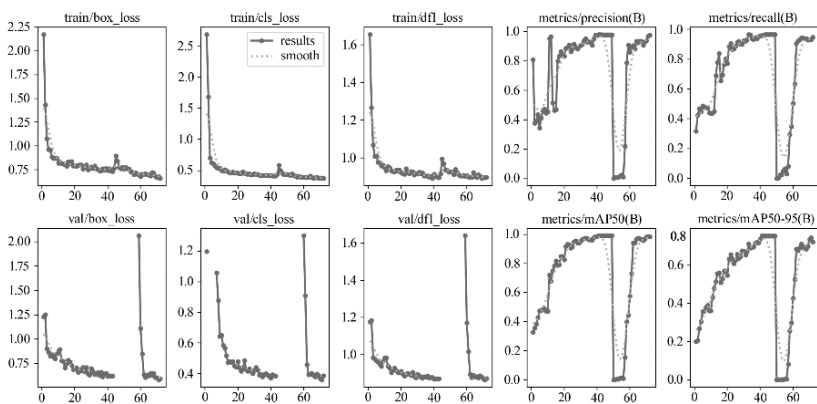


Рис. 2. Результаты обучения классификатора обнаружения падежа

Результаты обучения классификатора обнаружения птицы на кадре приведены на рис. 3.

Разработанные классификаторы позволяют частично оценивать функциональное состояние ПНС при совместном применении к анализируемому кадру изображения с камеры наблюдения. Для проведения полного анализа функционального состояния птицы могут потребоваться дополнительные классификаторы.

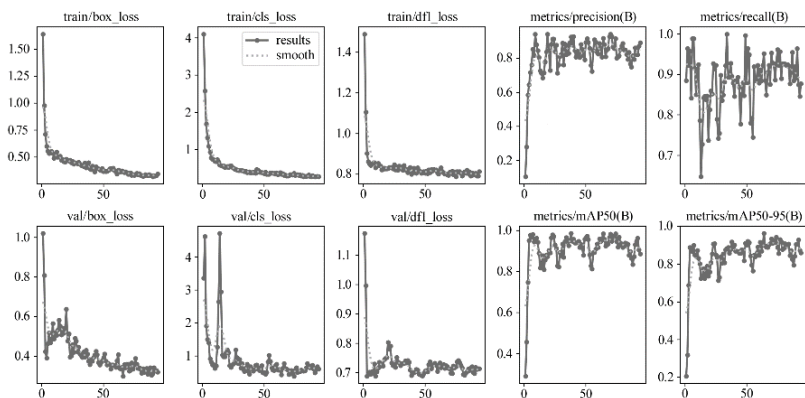


Рис. 3. Результаты обучения классификатора определения наполненности кормушек

2. Модификация структуры искусственной нейронной сети на основе архитектуры YOLO

Каждый из разработанных классификаторов по умолчанию реализует этап предобработки изображения. Данный этап состоит из следующих шагов:

- 1) загрузка изображения;
- 2) преобразование цвета изображения к формату RGB;
- 3) изменение размера изображения к стандартному размеру (640x352) с сохранением пропорций;
- 4) заполнение пустой части измененного по размеру изображения серым цветом;
- 5) нормализация полученных значений;
- 6) конвертация нормализованного изображения в тензор.

При простом последовательном использовании классификаторов этап предобработки будет выполнен при каждом применении классификатора.

В рамках данного исследования этап предобработки был вынесен в отдельное действие, производимое до применения классификаторов. Результат предобработки подается в качестве входных данных для соответствующего кортежа классификаторов (рис. 4).

Для апробации разработанного подхода были проведены эксперименты с использованием различного количества входных изображений и классификаторов. Первый эксперимент производился с использованием GPU. Результаты первого эксперимента приведены в табл. 2.



Рис. 4. Схема предлагаемого подхода

Таблица 2

Результаты эксперимента по измерению производительности подходов с использованием GPU

Кол-во классификаторов \ Кол-во изображений		1	2	3
1	YOLO, с.	0.015	0.0307	0.0426
	Модификация YOLO, с.	0.016	0.0284	0.0401
2	YOLO, с.	0.0172	0.0325	0.0494
	Модификация YOLO, с.	0.0167	0.0317	0.0474
3	YOLO, с.	0.0191	0.0366	0.06
	Модификация YOLO, с.	0.0184	0.0379	0.0589
4	YOLO, с.	0.0232	0.0445	0.1049
	Модификация YOLO, с.	0.0219	0.0436	0.0693
8	YOLO, с.	0.0459	0.0867	0.1276
	Модификация YOLO, с.	0.0403	0.074	0.1147
16	YOLO, с.	0.0828	0.1713	0.2511
	Модификация YOLO, с.	0.0798	0.1604	0.2491
32	YOLO, с.	0.1705	0.3257	0.5059
	Модификация YOLO, с.	0.1658	0.3163	0.4931

Представленные результаты учитывают вынесенный этап предобработки при измерении среднего времени, затрачиваемого предлагаемым подходом. В среднем время предобработки одного кадра с использованием GPU занимает до 0.0011 секунд.

Наиболее распространенной площадью промышленных корпусов является 2-2,5 тысячи метров квадратных на 40-50 тысяч голов. Для покрытия данной площади в среднем потребуется от 16 до 32 камер наблюдения. Средние результаты прироста производительности во время эксперимента, проводимого на GPU для 16-32 изображений, составили 3%.

Однако ситуация значительно изменяется при проведении того же эксперимента не на производительной станции с GPU, а одноплатном компьютере Raspberry 5. Результаты эксперимента приведены в табл. 3.

Таблица 3

Результаты экспериментов на Raspberry 5

Кол-во изображений \ Кол-во классификаторов		1	2	3
1	YOLO, с.	0.4509	0.9043	1.3637
	Модификация YOLO, с.	0.4741	0.8536	1.2403
2	YOLO, с.	0.8926	1.8409	2.6967
	Модификация YOLO, с.	0.8993	1.608	2.3981
3	YOLO, с.	1.3553	2.6686	4.0825
	Модификация YOLO, с.	1.2283	2.4321	3.673
4	YOLO, с.	1.7053	3.3909	5.0617
	Модификация YOLO, с.	1.5189	3.0545	4.6378
8	YOLO, с.	3.395	7.0359	10.1816
	Модификация YOLO, с.	3.0735	6.1258	9.3211
16	YOLO, с.	7.1735	14.748	21.9926
	Модификация YOLO, с.	6.5346	12.9744	20.0597
32	YOLO, с.	14.799	28.1167	42.0866
	Модификация YOLO, с.	12.5306	25.0595	37.5513

Представленные результаты учитывают вынесенный этап предобработки при измерении среднего времени, затрачиваемого предлагаемым подходом. В среднем время предобработки одного кадра занимает до 0.006 секунд.

Средние показатели прироста производительности во время эксперимента, проводимого на Raspberry 5 для 16-32 изображений, составили 12,5%.

Заключение

В результате экспериментов по модификации сверточных нейронных сетей на основе архитектуры YOLO был получен прирост производительности на 3% и 12,5% для рабочей станции с GPU и одноплатного компьютера Raspberry 5 соответственно. По результатам экспериментов сделан вывод о применимости предлагаемого подхода по модификации сверточных сетей семейства YOLO при работе на одноплатных или малопроизводительных компьютерах с целью повышения производительности систем анализа функционального состояния ПНС.

Разработанный подход позволяет уменьшить интервал, необходимый для отложенного анализа изображений, и становится более актуальным с ростом количества применяемых классификаторов и камер наблюдения.

Список литературы

- [Supriyanto et al., 2023] Supriyanto E., Isnanto R.R., Purnomo S.H. Computer vision in chicken monitoring system using machine learning: A General Review // In: E3S Web of Conferences (E3S Web Conf), Semarang, Indonesia, 2023, 02014. – doi: 10.1051/e3sconf/202344802014.
- [Neethirajan, 2022] Neethirajan S. Automated tracking systems for the assessment of farmed poultry // Animals. – 2022. – Vol. 12(3). – P. 232-245. – doi: 10.3390/ani12030232.
- [Redmon et al. 2016] Redmon J., Divvala S., Girshick R., Farhadi A. You Only Look Once: Unified, Real-Time Object Detection // In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016. – P. 779-788. – doi: 10.48550/arXiv.1506.02640.
- [Paramathma et al., 2024] Paramathma M.K., Kumar I.B., Karuppasamyandian M. YOLO Based Automatic Poultry Monitoring System. In: 3rd International Conference for Advancement in Technology (ICONAT), GOA, India, 2024. – P. 1-6. – doi: 10.1109/ICONAT61936.2024.10774960.
- [Chen et al, 2023] Chen Y., Peng X., Cai L., Jiao M., Fu D., Xu C.C., Zhang P. Research on automatic classification and detection of chicken parts based on deep learning algorithm // Journal of Food Science. –2023. –Vol. 88(10). – P. 4180-4193. – doi: 10.1111/1750-3841.16747.
- [Chemme et al., 2024] Chemme K.S., Alitappeh R.J. An End-to-End Model for Chicken Detection in a Cluttered Environment // In: 13th Iranian/3rd International Machine Vision and Image Processing Conference (MVIP), Tehran, Iran, 2024. – P. 1-7. – doi: 10.1109/MVIP62238.2024.10491186.
- [Triyanto et al., 2023] Triyanto W.A., Adi K., Suseno J.E. Detection and Tracking of Broiler Flock Movements in The Chicken Coop using YOLO // In: E3S Web of Conferences (E3S Web Conf), Semarang, Indonesia, 2023, 02064. – doi: 10.1051/e3sconf/202344802064.
- [Mehdizadeh et al., 2024] Mehdizadeh S.A., Siriani A.L.R., Pereira D.F. Optimizing Deep Learning Algorithms for Effective Chicken Tracking through Image Processing // AgriEngineering. – 2024. – Vol. 6(30). – P. 2749-2767. – doi: 10.3390/agriengineering6030160.
- [Siriani et al., 2022] Siriani A.L.R., Kodaira V., Mehdizadeh S.A., de Alencar Nääs I., de Moura D.J., Pereira D.F.. Detection and tracking of chickens in low-light images using YOLO network and Kalman filter // Neural Computing and Applications. – 2022. – Vol. 34. – P. 21987-21997. – doi: 10.1007/s00521-022-07664-w.
- [Zhu et al., 2022] Zhu X., Wu C., Yang Y., Yao Y., Wu Y. Automated Chicken Counting Using YOLO-v5x Algorithm. In: 8th International Conference on Systems and Informatics (ICSAI), Kunming, China, 2022. – P. 1-6. – doi: 10.1109/ICSAI57119.2022.10005522.

- [Sun et al., 2023] Sun E., Xiao Z., Yuan F., Wang Z., Ma G., Liu J. Method of classified counting of mixed breeding chickens based on YOLOV5. In: 42nd Chinese Control Conference (CCC), Tianjin, China, 2023. – P. 7644-7649. – doi: 10.23919/CCC58697.2023.10240449.
- [Wu et al., 2025] Wu Z., Yang J., Zhang H., Fang C. Enhanced Methodology and Experimental Research for Caged Chicken Counting Based on YOLOv8 // *Animals*. – 2025. – Vol. 15 (6). – 853. – doi: 10.3390/ani15060853.
- [Bist et al., 2025] Bist R.B., Subedi S., Yang X., Chai L. Automatic Detection of Cage-Free Dead Hens with Deep Learning Methods // *AgriEngineering*. – 2023. – Vol. 5(2). – P. 1020-1038. – doi: 10.3390/agriengineering5020064.
- [Syafaah et al., 2024] Syafaah L., Faruq A., Setyawan N., Khair M.I. Sick and dead chicken detection system based on YOLO algorithm // *Ingénierie des Systèmes d'Information*. – 2024. – Vol. 29(5). – P. 1723-1729. – doi: 10.18280/isi.290506.
- [Yang et al., 2024] Yang J., Zhang T., Fang C., Zheng H., Ma C., Wu Z. A detection method for dead caged hens based on improved YOLOv7 // *Computers and Electronics in Agriculture*. – 2024. – Vol. 226. – 109388. – doi: 10.1016/j.compag.2024.109388.
- [Tong et al., 2023] Tong Q., Zhang E., Wu S., Xu K., Sun C. A real-time detector of chicken healthy status based on modified YOLO // *SIViP*. – 2023. – Vol. 17. – P. 4199-4207. – doi: 10.1007/s11760-023-02652-6.
- [Sun et al., 2024] Sun S., Wei L., Chen Z., Chai Y., Wang S., Sun R. Nondestructive estimation method of live chicken leg weight based on deep learning // *Poultry Science*. – 2024. – Vol. 103(4). – 103477. – doi: 10.1016/j.psj.2024.103477.
- [Mahmoud et al., 2025] Mahmoud Y.S., Wael M.E., Awad A.T.O., Ahmed E., Ali S., Pankaj K., Tamer M.El-Messery, Tarek A.E-H., Mohamed F.A., Gomaa G.A.E-W., Ibrahim S.E-S., Abdallah E.E. Automated on-site broiler live weight estimation through YOLO-based segmentation // *Smart Agricultural Technology*. – 2025. – Vol. 10. – 100828. – doi: 10.1016/j.atech.2025.100828.

УСКОРЕНИЕ ОБМЕНА ДАННЫМИ ВОЗЛЕ ПРЕПЯТСТВИЙ ПРИ ГРУППОВОЙ МНОГОАГЕНТНОЙ ФУРАЖИРОВКЕ¹

Д.И. Зворыкин (*zuev.di@phystech.edu*)

П.С. Сорокоумов (*petr.sorokoumov@yandex.ru*)

Национальный исследовательский центр «Курчатовский институт»,
Москва

При решении задачи фуражировки группой автономных агентов весьма важно правильно организовать обмен информацией между участниками группы. В децентрализованной системе можно выполнять его путём попарных взаимодействий близкорасположенных агентов. В работе рассматривается роль препятствий и узких мест в формировании благоприятной для многочисленных обменов обстановки. Проведенное моделирование обучения с подкреплением групп агентов с разными организациями процессов обмена и при разных формах препятствий показало слабое воздействие на результаты случайно расположенных препятствий и более значительный эффект протяжённых преград. Роль обмена данными оказалась более высокой на начальном этапе работы системы и в условиях изменяющейся среды функционирования. Результаты работы могут использоваться для организации обмена данными при обучении групп агентов.

Ключевые слова: многоагентная фуражировка, обучение с подкреплением, обмен данными.

Введение

Задача фуражировки – одна из типичных проблем для систем групповой робототехники, имеющая множество теоретических и практических приложений. Среди прочих важных факторов её эффективности особую роль играет механизм обмена информацией. При децентрализованном управлении знания каждого отдельного агента о среде ограничены, по-

¹ Работа выполнена за счет государственного задания НИЦ «Курчатовский институт».

этому обмен позволяет активнее изучать окружение и эффективнее планировать действия. Логично предположить, что повышение числа обменов может в отдельных случаях ускорить оптимизацию поведения. В данной работе путём моделирования агентов, обучающихся с подкреплением по различным алгоритмам, оценивается ускорение обмена, получаемое при введении в окружение препятствий, повышающих интенсивность трафика в малых областях пространства.

На практике такая конфигурация часто встречается в постройках социальных насекомых – муравьёв и термитов. Узкие проходы внутри их построек, а также проложенные по грунту «дороги» позволяют многим особям постоянно встречаться и, потенциально, воздействовать друг на друга. Весьма полезно было бы оценить влияние такого сосредоточения на распространение информации внутри децентрализованной группы. Сделанные выводы можно применять в искусственных системах, выполняющих сходную с фуражировкой работу, чтобы оптимизировать их функционирование. Понятно, что объяснять таким образом поведение насекомых было бы неправильно, потому что биологические системы разнообразнее и сложнее абстрактных моделей, но во многих задачах робототехники полученные результаты могли бы оказаться полезными.

Далее в работе сначала приведён краткий обзор подходов к обмену данными между агентами децентрализованных систем, затем описаны варианты организации взаимодействия и постановка задачи обучения с подкреплением. Выполненное моделирование процессов обучения в разных условиях позволило оценить влияние разных вариантов препятствий на эффективность.

1. Методы обмена данными между агентами децентрализованных систем

Проблематика обмена данными между агентами при обучении рассматривалась неоднократно, при этом отмечалась значительная сложность этой задачи. Даже коммуникация между людьми, которые способны планировать свои действия для достижения общей цели и объяснять решения друг другу, оказывается весьма проблематичной [Wu et al., 2025]. В обзоре задач, стоящих перед разработчиками мультиагентных систем [Wong et al., 2023], коммуникация указана среди наиболее интересных проблем наряду с комбинированием централизованного обучения и децентрализованного исполнения, моделированием оппонента, групповой координацией и формированием функции награды. Задача коммуникации при этом может ставиться в виде обучения протоколу взаимодействия [Zhu et al., 2024]. Его выполняют разными способами, например на глубокой Q-сети общего вида [Foerster et al., 2016] или на более специализированной – трансформере [Yang et al., 2022]. При этом создать общий протокол взаи-

модействия для всех агентов весьма сложно, потому что децентрализация системы запутывает процедуру его согласования. Для целей данной работы обучение протоколу представляется излишне затратным, хотя оно оправдано при помехах в каналах связи, проблемах с идентификацией и защитой информации [Ahmed et al., 2024].

Решением проблем повторного использования полученных знаний в рамках обучения с подкреплением занимается отдельная область исследований – обучение путём переноса (transfer learning). В обзоре [Silva et al., 2019] описаны многие проверенные подходы данной области, например метрики успешности переноса знаний и методы их повторного использования как внутри агентов, так и при коммуникации между ними. Среди вариантов передаваемых между агентами знаний упоминаются функции награды, политики, оценки чужих состояний, советы по выбору действий (т.е. действие, выбранное бы агентом, если бы его состояние было тем же, что у партнёра по коммуникации), опыт выбора действий (тройки из состояний, выбранных действий и полученных за них наград) и др. Нарботки такого рода вполне применимы к решаемой задаче [Azmani et al., 2023].

Также хорошо известно использование среды для неявной коммуникации путём оставления следов (stigmergy) [Shaw et al., 2022], но его сложнее воплотить из-за технических проблем с реализацией требуемых следов на реальных роботах.

В целом многие механизмы обмена данными между фуражирующими агентами хорошо известны и проверены, но отношения между их эффективностью и расположением препятствий в среде изучены недостаточно.

2. Организация межагентного взаимодействия

При обмене данными между агентами хотелось бы найти оптимальный баланс между полнотой передаваемых данных, затратами на коммуникацию, скоростью и качеством распространения информации. На данный момент выделяют несколько семейств методов, направленных на достижение этой цели, которые условно можно разделить на три основные группы.

К реактивным (стигмергическим) подходам относятся механизмы косвенной координации через изменение среды. Агенты оставляют в среде маркеры, которые влияют на поведение других агентов, что создает обратную связь, основанную на принципах самоорганизации через усиление/затухание маркеров. Вычислительная сложность таких решений минимальна. Можно выделить такие подклассы следов, как феромоны, градиентные поля, трассировки. Агент, обнаружив ресурс, оставляет феромонный след с интенсивностью, пропорциональной качеству ресурса; получаемое градиентное поле обновляется по некоторому заданному правилу. Другие агенты двигаются вдоль градиента поля и корректируют маршрут при обнаружении более сильных сигналов.

К рыночным механизмам относится распределение задач через аукционы с применением рассчитанной на основе локальной информации предполагаемой награды как «валюты». Разрешение конфликтов можно реализовать, например, через модифицированный аукцион Викри [Vickrey, 1961], в котором сравниваются «общественные полезности» без агента и при его участии.

Третий подход основан на обмене обученными моделями и их частями. Агенты, использующие машинное обучение, совместно корректируют политики, периодически обмениваясь параметрами моделей для передачи опыта. Можно организовать, например, обмен Q-значениями (для табличных методов) или тройками <состояние, действие в нём, полученная награда>. В данной работе применяется именно этот принцип.

Возможно и сочетание указанных подходов, например использование феромонов для быстрого реагирования на важные события и обучения с подкреплением для долгосрочного планирования.

Проанализируем эффективность обмена данными на примере группы агентов, выполняющих фуражировку на клеточном поле (рис. 1). В центральной ячейке среды находится база агентов, в которую они должны доставлять пищу из её источников. Если агент войдёт на ячейку источника, он получит пищу, количество которой – случайная величина с заданным распределением. Эти распределения со временем не меняются (т.е. пища в источниках не иссякает), но у разных источников параметры распределений различны, причём агенты заранее их не знают. Полученную пищу агенты доставляют на базу, после чего снова двигаются к тому же или другому источнику.

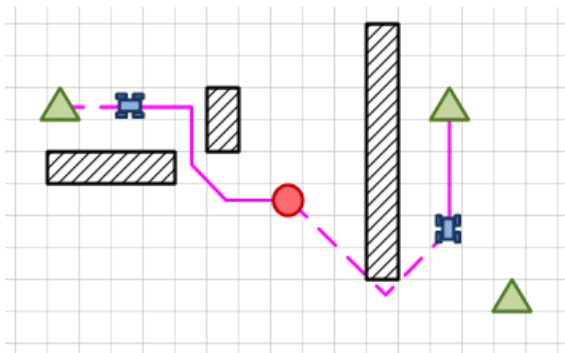


Рис. 1. Среда функционирования агентов: клеточное поле, по которому агенты-роботы доставляют пищу из источников (зелёных треугольников) к базе (красному кругу), избегая препятствий (заштрихованных прямоугольников)

Чтобы сосредоточиться на высокоуровневых взаимодействиях агентов, будем далее считать, что агенты знают заранее координаты всех источников (но не их обильность) и способны проложить маршрут к каждому из них, если препятствия (непроходимые ячейки) это позволяют, и маршрут не блокируется другими агентами. Это значит, что в задаче обучения с подкреплением в качестве действий рассматриваются не элементарные движения, а целые процедуры сбора пищи из источника. Каждая из них включает движение к источнику с базы по кратчайшему маршруту, получение пищи, движение обратно на базу и автоматическая выгрузка пищи. Количество этих действий равно числу источников пищи, доступных с базы.

Без обмена данными можно было бы свести задачу в такой постановке к простейшему случаю «многорукого бандита», но обмен существенно меняет ситуацию. Будем считать, что обмен между двумя агентами происходит, если они находятся поблизости друг от друга, т.е. расстояние между ними не превышает некоторый фиксированный порог. Обмен выполняется мгновенно, без дополнительных затрат времени, и не является отдельным действием, которое агент должен был бы намеренно выбирать. Это снижает гибкость системы, но устраняет проблемы при изучении агентом протокола взаимодействия: обмен данными и последующие дообучение моделей происходят в фоновом режиме, не прерывая выполнение текущей поведенческой процедуры агента.

Таким образом, действиями в задаче обучения с подкреплением являются комплексные поведенческие процедуры сбора пищи из источников; число таких действий равно числу источников пищи. Наградой агента является среднее число добытой пищи за такт моделирования, т.е. число собранной пищи, разделённое на длину маршрута. В состояние агента включаются текущие оценки обильности источников. При каждом акте обмена данными любые агенты передают друг другу координаты и обильность последнего посещенного участка, сведения о ранее совершённых действиях (прежнее состояние, действие, награда, следующие состояние) и текущие параметры своих моделей (Q-таблицы – при их наличии).

Это позволяет агентам корректировать свои политики. Если, например, при Q-обучении значение Q для некоторых аргументов у текущего агента было равно $Q^{current}$, а у другого агента Q^{other} , то новое значение Q^{new} будет получено по формуле:

$$Q^{new} = Q^{current} + \alpha (Q^{other} - Q^{current}) \frac{E^{other}}{E^{other} + E^{current}},$$

где α – скорость внесения корректив, $E^{current}$ – число посещений соответствующего участка текущим агентом, E^{other} – число посещений его же другим агентом. Это значит, что степень сдвига зависит от относительного опыта агентов в совершении конкретного действия.

Если один агент мешает другому пройти к нужной точке, применяется трехуровневый протокол разрешения конфликтов: 1) агенты с грузом получают приоритет, 2) в узких проходах используется правило правой руки для расхождения, 3) при невозможности разрешения конфликта один из агентов случайно отступает на свободную клетку. Ситуации полной блокировки предотвращаются принудительным сбросом цели при превышении лимита шагов миссии.

Таким образом, задача обучения с подкреплением формализована следующим образом:

- состояние агента – две координаты, количество имеющейся пищи, расстояние до базы, номер группы агентов, текущий градиент (для глубокого Q-обучения);
- набор допустимых действий – перемещения к каждому из кормовых участков с автоматическим сбором пищи и её доставкой на базу;
- награда при доставке пищи на базу – количество пищи, делённое на пройденное расстояние;
- методы обмена данными: дообучение каждого агента на последнем действии другого (как если бы он сам совершил это действие в том же состоянии и с тем же результатом) и, для классического Q-обучения, усреднение Q-значений описанным выше способом. Обмен выполняется каждый такт для каждой пары, сблизившейся на расстояние меньше порогового.

Выбранный метод обращения к алгоритму обучения с подкреплением осложняет применение готовых средств обучения многоагентных систем (PettingZoo и подобных), потому что в них, как правило, обращение к обучающему алгоритму происходит каждый шаг моделирования. К данной задаче из-за сочетания непрерывного обмена информацией и дообучения в произвольные моменты применение таких подходов затруднено, поэтому пришлось реализовывать небольшую собственную среду многоагентного моделирования, поддерживающую алгоритмы обычного и глубокого Q-обучения:

- для классического Q-обучения применялись скорость обучения $\alpha = 0.1$, коэффициент дисконтирования $\gamma = 0.95$; вероятность исследования ϵ меняется от 1.0 вплоть до 0.1, уменьшаясь каждый шаг в 0.997 раз;
- глубокое Q-обучение использовало для предсказания Q-функции персептрон с двумя скрытыми слоями по 64 нейрона с функциями активации ReLU, вычисляющий прогнозируемое Q-значение для каждого действия; использован оптимизатор Adam со скоростью обучения 0.0001.

В разработанном ПО реализованы также алгоритмы глубокого обучения PPO и PolicyGradient. Поскольку предварительные тестовые измерения показали, что они обучаются значительно медленнее Q-обучения, детальные исследования их поведения не проводились.

3. Моделирование и обсуждение результатов

Если взаимодействия между агентами ограничены некоторой дальностью связи, то целесообразно оценить влияние разных препятствий на расстояние между ними. Наличие большого количества препятствий в среде заставит агентов обходить их, что приведёт к появлению зон повышенной плотности агентов. Так как конфигурации препятствий могут быть весьма разнообразными, оценим влияние только двух простейших вариантов: случайно расположенных равномерно распределённых препятствий и сплошных непроходимых стенок с маленькими проходами. Некоторые другие конфигурации можно получить как комбинацию двух указанных, чтобы распространить на них сделанные выводы.

Чтобы оценить расстояния между агентами в разных окружениях, рассмотрим группу из двух агентов, которые независимо друг от друга перемещаются по клеточной карте из базы к случайно выбранной свободной точке и обратно. При этом возможные пункты назначения лежат внутри некоторой области досягаемости, скорости агентов одинаковы, задержек в пунктах назначения нет, и агенты не препятствуют движению друг друга (считаются точечными). Для сбора данных о расстоянии была реализована вспомогательная имитационная модель. Оказалось, что плотность распределения вероятности расстояния между агентами имеет вид, показанный на рис. 2. При этом доля времени, когда агенты могут обмениваться информацией, определяется как часть площади под полученной кривой, ограниченная порогом дальности коммуникации.

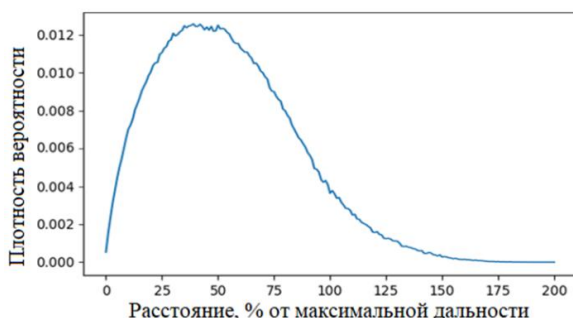


Рис. 2. Плотность распределения расстояния между агентами при отсутствии препятствий. Максимальное удаление одного робота от базы принято за 100%

Введение препятствий модифицирует данное распределение, потому что агенты располагаются друг к другу в среднем несколько ближе. При случайно расположенных препятствиях кривая слегка смещается к малым значениям, сохраняя форму (рис. 3). При этом росту вероятности появле-

ния препятствия в каждой ячейке с 0 до 0.5 соответствует падение медианного расстояния между агентами от 54% до 52% от дальности хода робота. Таким образом, влияние таких препятствий на расстояние слабо.

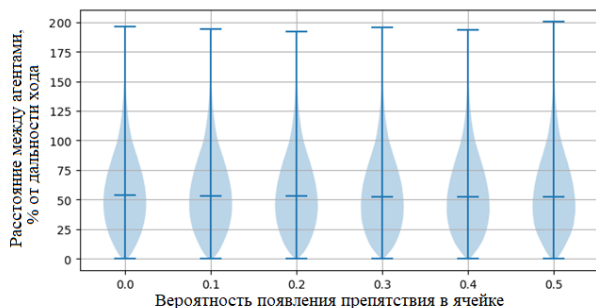


Рис. 3. Плотность распределения расстояния между агентами при независимо расположенных препятствиях; размер карты – 100x100 ячеек, дальность хода агента – 50 ячеек

Если препятствия образуют сплошную стенку с одним проходом вокруг базы, форма распределения существенно меняется (рис. 4).

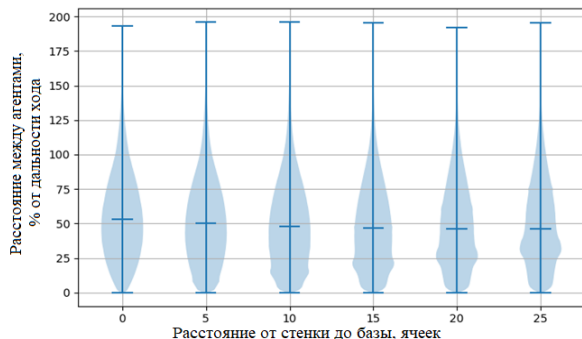


Рис. 4. Плотность распределения расстояния между агентами при препятствиях, образующих вокруг базы квадрат со стороной 20 ячеек с одним отверстием посередине одной стороны; размер карты – 100x100 ячеек, дальность хода агента – 50 ячеек

Видно, что медианное значение уменьшается за счёт повышения вероятности малого расстояния между агентами. При этом доля самых малых расстояний существенно растёт: на карте без препятствий роботы находились на расстоянии меньше 5 ячеек с вероятностью 1.5%, при наличии стенки в 10 ячеек от базы – с вероятностью 4%.

С учётом полученных результатов было проведено моделирование обучения многоагентных групп, участники которых автоматически обмениваются информацией при сближении. Квадратная карта имела сторону в 150 ячеек, кормовых участков на ней – 16, дальность связи – 5 ячеек. Обмены ускоряют сбор пищи даже без препятствий (рис. 5).

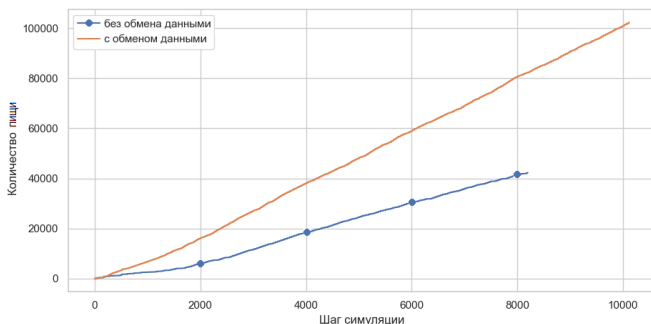


Рис. 5. Сравнение производительности групп с обменом данными и без него. Препятствия отсутствуют. В модели участвовали два робота, обучающиеся по алгоритму глубокого Q-обучения; среднее количество пищи в участках распределено равномерно от 1 до 100

Также при наличии обменов система быстрее находит наилучший участок из имеющихся (рис. 6) и активнее его эксплуатирует.

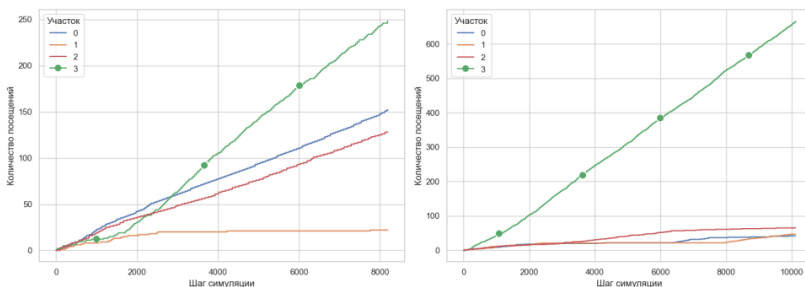


Рис. 6. Количество посещений агентами каждого из четырёх случайно расположенных кормовых участков. Один из них (№3) обильнее других; обмен информацией позволяет быстрее определить его. В модели участвовали два робота, применяющие алгоритм классического Q-обучения с обменом данными

Моделирование работы нескольких алгоритмов обучения с подкреплением в среде со случайно расположенными препятствиями показало, что при небольшой доле препятствий (до 40%) результаты у всех реализованных алгоритмов менялись незначительно, при большем числе препятствий – несколько падали из-за удлинения путей к пище.

Моделирование среды с протяжённым препятствием - сплошной стеной в 10 ячейках вокруг базы с одним малым отверстием – показало, что несмотря на рост числа обменов данными, производительность агентов в целом упала (рис. 7). Причиной этого стали большие расстояния, которые пришлось проходить для сбора пищи, и временные затраты агентов для разрешения коллизий. Это значит, что интенсивный обмен данными помог системе приспособиться к среде, но одновременно препятствия ухудшили её свойства, удлинив маршруты до цели.

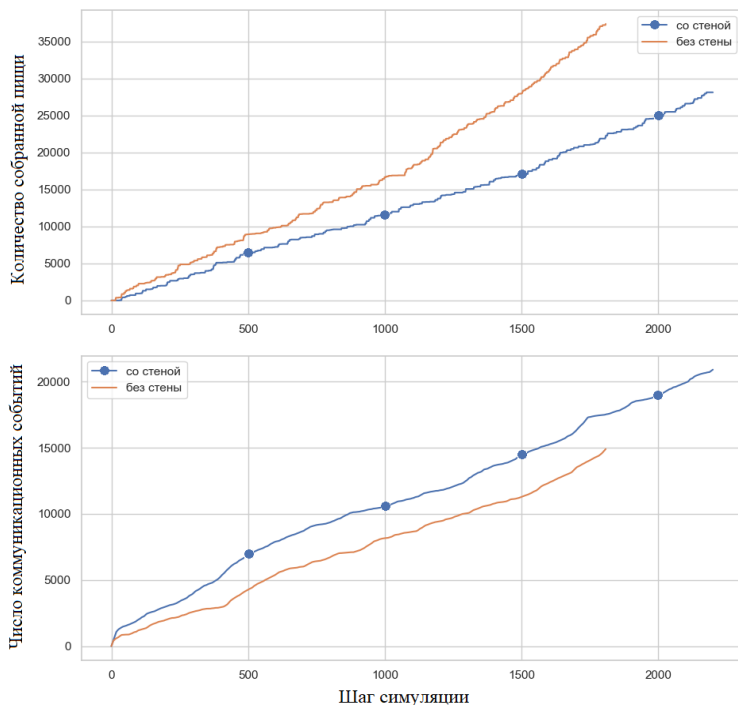


Рис. 7. Количество собранной пищи (сверху) и число обменов данными (внизу) при сборе пищи в среде с протяжённым препятствием - сплошной стенкой вокруг базы с одним проходом. Использован алгоритм глубокого Q-обучения

Таким образом, повышение интенсивности обмена желательно проводить менее жёсткими методами; для поставленной задачи это, например, использование не одного узкого прохода, а четырёх, расположенных в центрах сторон квадратного ограждения. Кроме того, при постоянных параметрах среды обмен играет роль только до нахождения всеми агентами оптимальной стратегии, поэтому можно ожидать, что его роль сильнее

проявится в изменяющейся среде. Для проверки этой гипотезы было проведено моделирование в условиях, когда обилие источников периодически менялось: количество пищи, которое мог получить каждый агент из источника на шаге t , при этом равнялось

$$A \frac{1 + \sin(\frac{2\pi t}{T} + \varphi)}{2},$$

где A – случайно выбранное для каждого источника в начале моделирования максимальное значение, φ – случайная фаза из диапазона $[0, 2\pi)$, также своя для каждого источника; $T=1000$. Если стена достаточно сильно удалена от базы (на 20 ячеек), оказалось, что на начальном этапе скорость сбора пищи с препятствиями превышает скорость без препятствий (рис. 8). Связано это с большим временем, проводимым агентами вместе, и с большими их затратами на движение к далёкой пище.

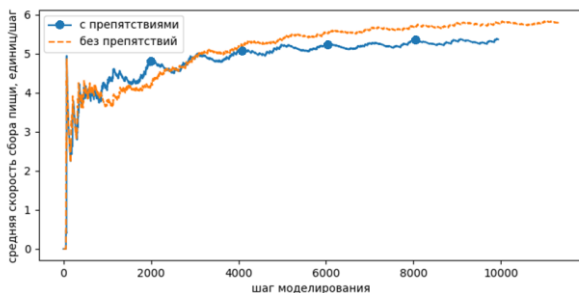


Рис. 8. Скорость сбора пищи при меняющейся среде и 4 отверстиях в ограждении. Видно, что на отрезке от 1000 до 3000 шагов препятствия увеличивают её, далее – снижают

Исследование показало, что характер получаемых результатов – как по скорости обучения, так и по эффективности системы, – для алгоритмов классического и глубокого Q-обучения совпадает.

Заключение

Таким образом, было показано, что препятствия могут интенсифицировать обмен данными, если для него требуется сближение агентов. Более эффективны при этом протяжённые преграды, тогда как при их случайном расположении эффект незначителен. Иногда после этого повышается общая эффективность работы многоагентной системы. Однако наличие чрезмерно больших препятствий слишком сильно ухудшает свойства среды, затрудняя работу, поэтому следует тщательно планировать положение мест скопления агентов. Также при малых дальности локальной связи весьма важен будет эффективный механизм расхождения агентов в узких проходах, который в данной работе детально не исследовался.

Список литературы

- [Ahmed et al., 2024] Ahmed M.H., Ghasemi M. Privacy-preserving decentralized actor-critic for cooperative multi-agent reinforcement learning // Proceedings of The 27th International Conference on Artificial Intelligence and Statistics Proceedings of Machine Learning Research / eds. S. Dasgupta, S. Mandt, Y. Li.: PMLR, 2024. – P. 2755-2763.
- [Azmani et al., 2023] Azmani H. et al. Cooperative foraging behaviour through multi-agent reinforcement learning with graph-based communication // Sixteenth European Workshop on Reinforcement Learning. Vrije Universiteit Brussel, Brussels, Belgium, 2023.
- [Foerster et al., 2016] Foerster J. et al. Learning to communicate with deep multi-agent reinforcement learning // NIPS'16: Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016. – P. 2145-2153. – DOI: 10.5555/3157096.3157336.
- [Shaw et al., 2022] Shaw S. et al. ForMIC: Foraging via multiagent RL with implicit communication // IEEE Robot. Autom. Lett. – 2022. – Vol. 7. – P. 4877-4884. – DOI: 10.1109/LRA.2022.3152688.
- [Silva et al., 2019] Silva F., Costa A. A Survey on Transfer Learning for Multiagent Reinforcement Learning Systems // J. Artif. Intell. Res. – 2019. – Vol. 64. – DOI: 10.1613/jair.1.11396.
- [Vickrey, 1961] Vickrey W. Counterspeculation, Auctions, and Competitive Sealed Tenders // J. Finance. – 1961. – Vol. 16, Issue 1. – P. 8-37. – DOI: 10.2307/2977633.
- [Wong et al., 2023] Wong A. et al. Deep multiagent reinforcement learning: challenges and directions // Artif. Intell. Rev. – 2023. – Vol. 56, Issue 6. – P. 5023-5056. – DOI: 10.48550/arXiv.2106.15691
- [Wu et al., 2025] Wu C. M. et al. Adaptive mechanisms of social and asocial learning in immersive collective foraging // Nat. Commun. – 2025. – Vol. 16, Issue 1. – P. 1-15. – DOI: 10.1038/s41467-025-58365-6.
- [Yang et al., 2022] Yang Y. et al. Transformer-based working memory for multiagent reinforcement learning with action parsing // NIPS'22: Proceedings of the 36th International Conference on Neural Information Processing Systems. Article. – 2022. – No. 2527. – P. 34874-34886.
- [Zhu et al., 2024] Zhu C., Dastani M., Wang S. A survey of multi-agent deep reinforcement learning with communication // Autonomous Agents and Multi-Agent Systems. – 2024. – Vol. 38, Issue 1. – P. 2845-2847. – DOI: 10.1007/s10458-023-09633-6.

УДК 004.8

doi: 10.15622/rcai.2025.045

НЕКОТОРЫЕ БИОИНСПИРИРОВАННЫЕ АГЕНТНЫЕ МОДЕЛИ СТРОИТЕЛЬСТВА ГНЕЗД¹

В.Э. Карпов (*karpov.ve@gmail.com*)

Национальный исследовательский центр «Курчатовский институт»,
Москва

Рассматриваются две агентные модели строительной деятельности. На примере строительства аналога гнезда муравьев показано, что агенты способны возводить купол и имитировать прокладывание тоннелей, используя базовый закон группового движения без необходимости активного межагентного взаимодействия и без применения процедуры планирования. В основе базового механизма управления движением лежит обобщенный метод потенциалов, когда для различных классов объектов формируются положительные и отрицательные потенциалы. Их суммирование с учетом значений вектора коэффициентов факторов позволяет реализовывать различные эффекты движения: стремление держаться вместе, отталкивание, движение по пеленгу или к выбранной точке и т.п. Переключение между формами движения осуществляется конечным автоматом. Строительство купола основано на возможности агентов взбегать друг на друга, а областью интереса согласованного движения группы является множество агентов, находящихся в терминальных состояниях. Строительство тоннелей является более сложной задачей, для которой следует определять как генеральное направление (пеленг) движения, так и задавать условия его начала/завершения.

Ключевые слова: агент, модели поведения, метод потенциалов, строительство гнезд, автоматное управление.

Введение

подавляющее большинство работ в области групповой робототехники посвящено согласованному движению – основе основ деятельности роботов. Задачи, в которых группы роботов занимаются преобразующей, на-

¹ Работа проведена в рамках выполнения государственного задания НИЦ «Курчатовский институт».

пример, строительной, деятельностью менее исследованы. Причиной является масса сложностей технического характера: распознавание, манипуляции объектами, планирование, согласование деятельности и т.п. Тем не менее, попытки реализации «строительной» деятельности на группах реальных роботах имеют давнюю историю. См. например, работу [Parker et al., 2003], в которой роботы расчищали поверхность (т.н. "слепое бульдозерирование" – blind bulldozing), или работу [Werfel, 2012], в которой описываются роботы, не только реализующие манипуляционные функции, но и способные взбираться на уложенные ими же блоки. А в работе [Huang et al., 2022] представлен весьма функциональный роботизированный механизм, выполняющий множество строительных действий в сухом гранулированном материале и способный работать в группе. При этом чаще всего групповая строительная деятельность сводится к моделям, основу которых составляет групповое движение, как, например, коллективное перемещение грузов ([Faria Dias et al., 2021]). Важно, что в основном используемые модели управления являются формальными, допускающими решение оптимизационных задач. Так, в работе [Czyzowicz et al., 2021] описывается система управления роботом на основе конечного автомата, задача которого построить максимально компактную возможную структуру – гнездо, при этом оптимизируется время этого строительства.

Здесь представлен иной – биоинспирированный – подход к решению задачи строительства гнезда. При этом модельным объектом будут общественные насекомые – муравьи. Строго говоря, этот подход биоинспирирован частично, т.к. достоверные данные о принципах, которыми руководствуются муравьи при строительстве, отсутствуют, и во многом механизмы строительства основаны на гипотезах и предположениях. Интерес именно к биоинспирированному подходу вызван тем, что, работая в реальных сложных, недетерминированных и слабоформализуемых средах, целесообразно иметь реализацию небольшого набора базовых механизмов управления и межагентного взаимодействия, адаптированных под конкретные условия, так, как это осуществляется в природе.

В этой работе будут рассмотрены два феномена строительной деятельности – строительство купола гнезда и создание тоннелей под ним. Литературы, посвященной гнездам муравьев, достаточно много. Так, структуры гнезд описываются в работе [Захаров и др., 2015] или в [Длусский, 1967], а в работе [Длусский, 1981], детально описываются действия муравьев при постройке гнезд. Здесь нам важно, что строительное поведение рабочих особей при этом состоит из серии стереотипных действий или блоков поведенческих актов, при этом сами эти акты (рытье, отгребание, формовка и т.п.), точнее, их реализация, нас не интересуют.

Интересно, что в той же работе [Длусский, 1981] отмечается, что не вся гнездостроительная деятельность муравьев может быть сведена к набору стереотипных блоков. Более того, там же говорится о программе 1975 г, моделирующей постройку гнезда муравьями. Характерно, что полученные результаты хорошо описывали прокладку хода, но не предсказывали ни ветвления ходов, ни постройки камер. Этот аспект будет обсужден в работе позже.

Итак, будем полагать, что нас интересует поведение агентов (строго говоря – аниматов как агентов, имитирующих поведение животных), направленное на а) создание купола, а также тоннелей, см. рис. 1,а, при этом «семантика» тоннелей и их структура нас не интересует (см. рис. 1,б).

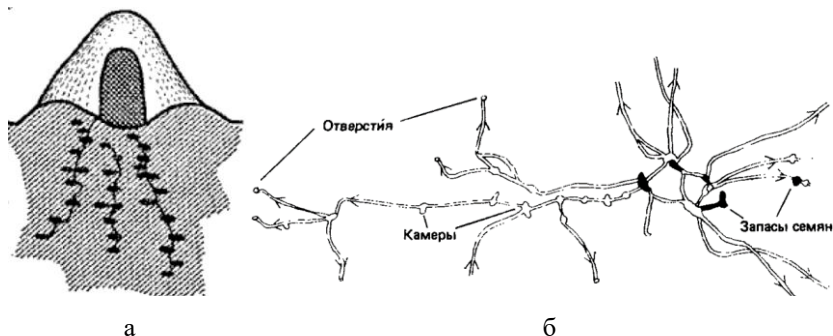


Рис. 1. а – гнездо с наземным куполом, [Захаров, 2018],
б – план гнезда, [Брайен, 1986]

2. Модель агента

Агент представляет собой объект, оснащенный сенсорами и способный к перемещению в трехмерном пространстве. Агент A задается пятеркой

$$A = \langle \text{name}, C, R, S, F \rangle, \quad (2.1)$$

где *name* – имя агента, C – множество характеристик агента, R – его пространственное положение, S – состояние сенсорной системы, F – функция управления.

Функция управления F определяет пространственное положение агента:

$$F(C, S, R) \rightarrow R. \quad (2.2)$$

Базовыми характеристиками агента для нашей задачи является тройка

$$C = \langle t, r, \omega, q \rangle,$$

где t – тип агента $t \in \mathbb{N}$, r – размер (геометрические характеристики, $r \in \mathbb{R}$) агента, ω – вес агента ($\omega \in \mathbb{R}$), а q – состояние агента (элемент конечного множества, $q \in \mathbb{N}$).

Пространственное положение R задается тройкой $R = \langle X, v, A \rangle$, где X – координаты агента, $X \in \mathbb{R}^3$; v – скорость движения ($v \in \mathbb{R}$); A – тройка углов Эйлера, $A = \langle \alpha_{yaw}, \alpha_{pitch}, \alpha_{roll} \rangle$. α_{yaw} , α_{pitch} , α_{roll} – углы рыскания, тангажа и крена соответственно.

Сенсорная система агента позволяет получить информацию о состоянии, координатах, скоростях и углах ориентации некоторого множества агентов Ω_s , попавших в его зону видимости – некоторую область пространства вблизи агента \mathbb{R}_s^3 , например, его переднюю полусферу.

Движение агента. Пусть направление и скорость движения агента определяется методом потенциальных полей. Это значит, что каждый агент из области видимости Ω_s рассматривается как источник потенциала (положительного или отрицательного), и сумма этих потенциалов определяет значение функции управления

$$F_{ctl} = \langle v_{ctl}, A_{ctl} \rangle, A_{ctl} = \langle \alpha_{yaw}, \alpha_{pitch}, \alpha_{roll} \rangle.$$

Обозначим через $\langle X_s, v_s, A_s \rangle$ средние значения координат, скорости и ориентации наблюдаемых в Ω_s агентов. Компоненты координат центра масс наблюдаемых агентов вычисляются как

$$X_s = \frac{\sum_{i \in \Omega_s} X_{ik}}{\sum_{i \in \Omega_s} \omega_i}, k = 1, 2, 3. \quad (2.3)$$

Здесь X_{ik} – компоненты вектора координат X_i наблюдаемого агента, ω_i – его вес. Аналогично определяются средние скорости v_s и вектор ориентации A_s .

Эти характеристики области Ω_s позволяют задавать гибкие законы движения: не только двигаться по направлению, определяемому центром масс X_s , но и согласовывать с другими агентами направление движения и скорость, решать задачу преследования и т.п.

Итак, на входе имеются состояние агента $\langle X, v, A \rangle$ и наблюдаемые величины $\langle X_s, v_s, A_s \rangle$, задан закон взаимодействия в виде функции $F_A(X, X_s, A_s)$, а искомым выходом является управление $\langle v_{ctl}, A_{ctl} \rangle$.

F_A возвращает компоненты вектора управления $\langle v_{ctl}, A_{ctl} \rangle$, причем

$$v_{ctl} = |F_A(X, X_s, A_s)|, A_{ctl} = \arg(F_A(X, X_s, A_s)).$$

Например, для закона когезии (стремления держаться поближе к «своим») F_{coh} итоговая скорость в общем случае может определяться собственной скоростью агента и расстоянием до центра масс группы r_s , определяемым по X и X_s :

$$v_{ctl} = |F_{coh}| = \min\left(v_{max}, \frac{k_v}{r_s}\right). \quad (2.4)$$

Здесь v_{max} – предельная скорость движения, k_v – некоторый коэффициент. А направление движения может быть определено из списка всех наблюдаемых "своих".

Если нас интересует простое движение агента по заданному пеленгу, заданному парой углов α_{yaw} и α_{pitch} , то $F_{bear} = \langle v_{ctl}, \mathbf{A}_{ctl} \rangle = \langle v_{ctl} = const, \alpha_{yaw}, \alpha_{pitch}, 0 \rangle$.

Общий закон управления движением. Пусть область Ω_s позволяет выделять классы объектов Ω_s^i , по отношению к которым реализуются разные законы движения F_i , как зависящие от области видимости (приближение, отталкивание), не связанные с ней (свободное блуждание, движение по пеленгу и т.д.).

Управление в общем случае будет определяться как

$$F_{ctl} = \sum_{i=1}^n k_i F_i(\Omega_s^i) = \begin{bmatrix} k_1 \\ \dots \\ k_n \end{bmatrix} [F_i(\Omega_s^i) \quad \dots \quad F_i(\Omega_s^i)]. \quad (2.5)$$

Здесь $F_i(\Omega_s^i)$ – сумма векторов сил, определяемых положением объектов из области Ω_s^i по отношению к агенту (либо вектор силы между агентом и центром тяжести объектов из Ω_s^i), $F_i(\Omega_s^i) = \sum_{p \in \Omega_s^i} \frac{w_p}{r^2} \begin{bmatrix} x_a - x_p \\ y_a - y_p \\ z_a - z_p \end{bmatrix}$, (x_a, y_a, z_a) и (x_p, y_p, z_p) – координаты агента и объекта p , w_p – вес объекта p , r – расстояние от агента до объекта; $K = [k_1, \dots, k_n]^T$ – вектор коэффициентов, значения которых определяют характер поведения агента (как и на что он реагирует). В простейшем случае подбор этих коэффициентов позволяет, например, реализовать классическую модель движения группы агентов К. Рейнольдса [Reynolds, 1987].

Перейдем далее к описанию более сложных форм поведения агента. При этом будем полагать, что все веса агентов ω постоянны и равны 1, а их линейная скорость v либо равна 0 (агент неподвижен), либо постоянна. Т.е. управление сводится к определению направления движения.

Отметим, что предлагаемые модели строительного поведения не сводятся исключительно к движению. Просто в цепочке триад "*определение состояния*" \rightarrow "*перемещение в целевую точку*" \rightarrow "*выполнение операции*" в данной работе рассматриваются первые два компонента в силу того, что "*выполнение операции*" – это уже частная задача, зависящая от конкретики операций и конструкции агента (робота).

3. Формирование купола

Пусть купол строится из "тел" самих агентов. Это допущение введено из соображений простоты, т.к. суть такой деятельности не будет отличаться от той, когда агенты будут нести строительный материал. Само строительство заключается в том, что агенты пытаются взобраться друг на друга, если позволяет высота, т.е. агент может взобраться на другого, если последний свободен. Это означает, что можно взобраться на высоту, оп-

ределяемую линейным размером агента. Условно такой процесс изображен на рис. 2,а. Агент может находиться в одном из трех состояний: F (свободен), A (активен) и T (заключительное неподвижное состояние). В состоянии F агент реализует процедуру поиска агентов, находящихся в состоянии T (агент ищет место строительства). Когда агент оказывается в непосредственной близости к агенту в состоянии T , он переходит в состояние A . В случае если агент в состоянии A может взобраться на соседа, он это делает и продолжает пребывать в этом активном состоянии. Если взобраться нельзя (не на кого или не позволяет высота), то агент переходит в состояние T . Такое поведение иллюстрируется на рис. 2,б.

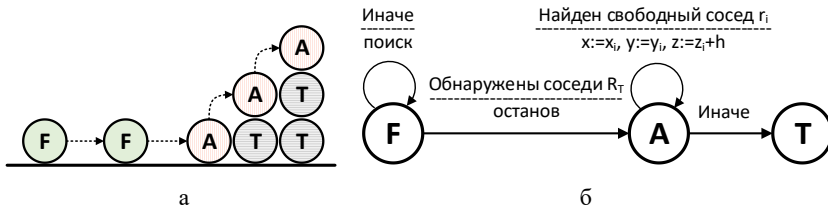


Рис. 2. а – стадии строительства, б – диаграмма состояний агента

Алгоритм работы агента a приведен ниже.

```

1: function Think(a)
2:   if  $q_a = T$  then return endif
3:    $R_T \leftarrow \{r_i \mid \text{dist}(a, r_i) < \varepsilon, z_{r_i} = z_a, q_{r_i} = T\}$  -- Неподвижные соседи агента
4:   if  $q_a = F$  then -- Агент свободен
5:     if  $R_T = \emptyset$  then -- Соседей нет
6:        $\langle v_a, A_a \rangle \leftarrow F_{ct}$  -- Закон группового движения к агентам T
7:     Else
8:        $q_a \leftarrow A$ 
9:     Endif
10:    Return
11:  Endif
12:  -- Агент активен
13:  if  $R_T = \emptyset$  then -- Соседей нет
14:     $q_a \leftarrow T$ 
15:    Return
16:  Endif
17:  for  $e \in R_T$  do
18:    if can_climb(a, e) then -- Определение занятости соседа
19:       $z_a \leftarrow z_e$ 
20:      Return
21:    Endif
22:  Enddo
23:   $q_a \leftarrow T$ 

```

Временная сложность процедуры построения гнезда для N агентов, в силу ее линейности, оценивается как $O(N)$. На рис. 3 показаны результаты имитационного моделирования. Зеленым цветом обозначены агенты в свободном состоянии F , красным – в активном A , а черным – агенты в заключительном состоянии T .

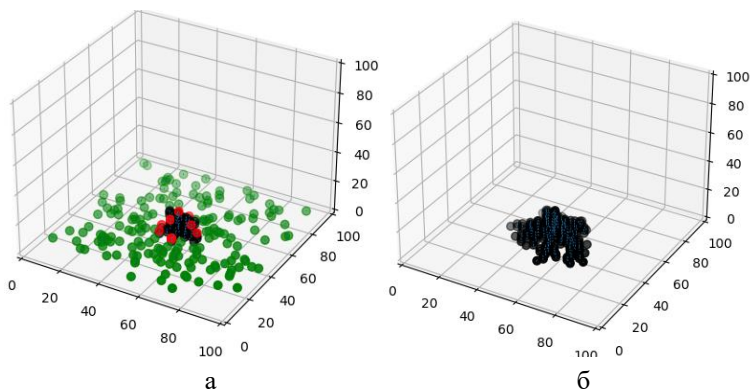


Рис. 3. а – начальный этап строительства, б – заключительный этап. Количество агентов – 200

Отметим, что здесь и ниже вычислительные эксперименты носили существенно качественный, иллюстративный характер. Их цель – лишь продемонстрировать возможность формирования тех или иных структур.

4. Формирование тоннелей

Следующей задачей является прокладывание путей (тоннелей) в некоторой неоднородной области, которая содержит множество препятствий, мешающих движению робота. Для решения этой задачи следует определить условия окончания работы агентов, а также направление их движения. Далее будут рассмотрены два варианта расположения этих областей. Важно, что во всех случаях агенту достаточно иметь два состояния: активное A и терминальное T . Алгоритм работы агентов тоже будет одинаков за исключением условия окончания и значений параметров K закона движения (2.5).

Горизонтальные тоннели. Рассмотрим ситуацию, когда эта область находится на плоскости, а положение препятствий, размещаемых случайным образом, ограничено кругом. Принцип работы системы заключается в том, что в центр области помещается агент в состоянии T . Этот агент задает направление движения остальных. Обобщенный закон (2.5) определяется вектором K , задающим движение по направлению к области интереса, например:

$$K = \begin{bmatrix} k_{brown} \\ k_{interest} \\ k_{bear} \end{bmatrix} = \begin{bmatrix} 0.05 \\ 0.5 \\ 0 \end{bmatrix}.$$

Здесь коэффициенты определяют значимость броуновского движения (k_{brown}), движения к области интереса ($k_{interest}$) и движения в определенном направлении, по пеленгу (k_{bear}).

Область интереса определяется множеством наблюдаемых агентов, находящихся в состоянии T . Алгоритм поведения агента a приведен ниже.

```

1: function ThinkTunnel(a)
2:   if  $q_a = T$  then return endif
3:   if eoj(a) then -- Условие завершения работы
4:      $q_a \leftarrow T$ 
5:   else
6:      $\langle v_a, A_a \rangle \leftarrow F_{ctl}$  -- Закон группового движения
7:   endif

```

Условие завершения работы агента заключается в обнаружении соседей в состоянии T :

```

1: function eoj(a)
2:    $R_T \leftarrow \{r_i \mid \text{dist}(a, r_i) < \varepsilon, z_{ri} = z_a, q_{ri} = T\}$ 
3:   return  $R_T \neq \emptyset$ 

```

На рис. 4 изображены начальная и конечная фазы прокладывания тоннелей.

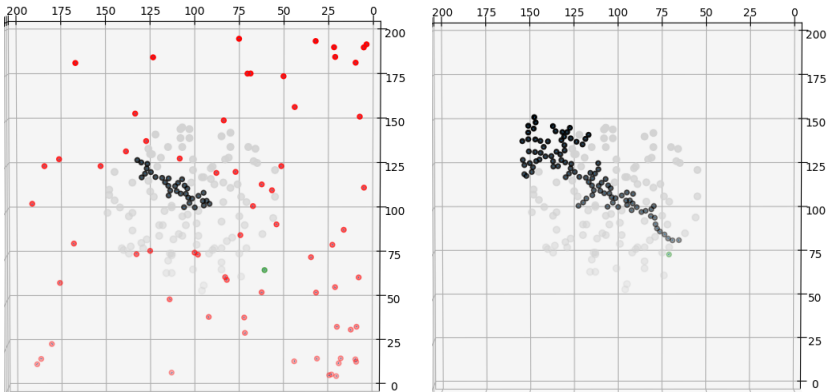


Рис. 4. Этапы прокладывания тоннелей по направлению к центру области. Красные точки – агенты в активном состоянии А, черные – в состоянии Т, светло-серые точки – «грунт» (препятствие). Количество агентов – 100

Вертикальные тоннели. Здесь область представлена полуплоскостью. "Рытье" вглубь заключается в том, что характер движения определяется пеленгом, направленным, условно, вглубь. Условие завершения заключается в наличии соседей в состоянии T или достижении "максимальной глубины" h_{max} :

```

1: function eoj(a)
2:    $R_T \leftarrow \{r_i \mid dist(a, r_i) < \varepsilon, z_{ri} = z_a, q_{ri} = T\}$ 
3:   return  $R_T \neq \emptyset$  or  $y_a > h_{max}$ 

```

Направление движения вглубь определяется неким пеленгом, все остальные факторы не учитываются (только $k_{bear} > 0$):

$$K = \begin{bmatrix} k_{brown} \\ k_{interest} \\ k_{bear} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0.5 \end{bmatrix}.$$

Алгоритм такого поведения по-прежнему описывается процедурой *ThinkTunnel*.

Это означает, что все агенты независимо начинают рытье в глубину. Количество возможных ходов и их вид определяется насыщенностью препятствиями и характером их расположения. В том случае, когда агент не может продолжить движение и находится на одном месте некоторое продолжительное время, он заново начинает процедуру поиска возможного хода. На рис. 5,а представлены результаты такой работы агентов, причем некоторые тоннели не достроены.

Можно ввести некий иницирующий фактор начала строительства, и тогда мы будем иметь дело с агентами, имеющими три состояния: F – свободное, A – активное и T – терминальное. Все агенты находятся изначально в состоянии F , а в состояние A переходят тогда, когда вокруг есть соседи (в природе одним из иницирующих факторов начала работы является скученность, нехватка пространства). Результаты работы будут теми же, что и для агентов с двумя состояниями, см. рис. 5,б.

Временная сложность процедур построения тоннелей обоих видов для N агентов может быть оценена как $O(N \cdot L)$, где L – насыщенность препятствиями области строительства.

Выше упоминалось, что реальные тоннели могут обладать весьма разветвленной структурой, содержать камеры, отводы и т.д. Как эти камеры и отводы появляются, на основании чего насекомые принимают решение рыть именно туда и именно так, биологов не интересует, равно как и ответ на вопрос, когда и на какой глубине останавливается процесс копания. На самом деле, есть основания полагать, что структура тоннелей определяется свойствами грунта (копаем туда, куда копаемся), а момент остановки определяется тем, что объема внутреннего пространства тоннелей доста-

точно для того, чтобы «выключить» строительную программу. Если структура, форма тоннелей подтверждается результатами имитационного моделирования, то условия начала и останова работы в рассматриваемой модели, конечно, весьма искусственны.

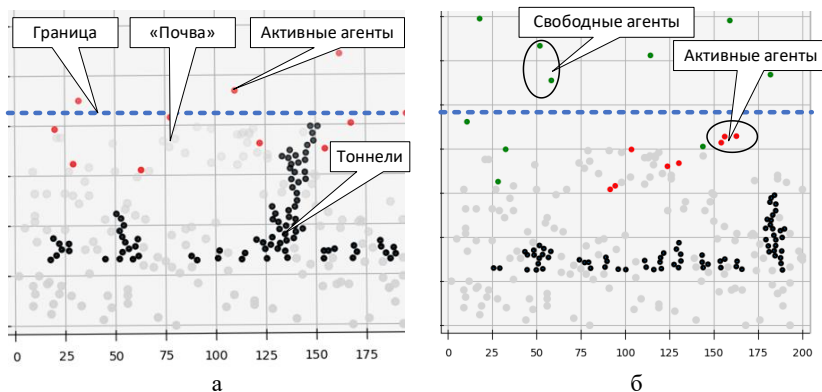


Рис. 5. Рытье тоннелей в глубину: а – агенты с двумя состояниями, б – с тремя состояниями. Количество агентов – 100

Заключение

В этой работе не рассматривались вопросы реализации собственно строительных операций, которые, безусловно, и важны, и интересны. Задача заключалась в ином: определить стратегию строительной деятельности, позволяющей агентам выполнять массовые операции, приводящие к созданию сложных структур. И при этом, подобно общественным насекомым, обходиться без плана, централизованного управления и используя косвенное межагентное взаимодействие.

Ключевыми факторами этой деятельности является определение условий ее начала и окончания. Относительная "простота" поведения при возведении купола объясняется тем, что имеется единственный центр строительства – агент, по каким-то причинам положивший туда строительный материал. Остальные просто подражают ему, переходя в активное состояние. Так же естественно определяется и окончание строительства (хотя в природе купол строится постоянно, а его рост ограничен только имеющимися ресурсами).

Со строительством тоннелей получается гораздо сложнее. Здесь факторы начала/окончания работ не столь очевидны. Кроме того, рытье тоннелей подразумевает "параллельную" работу многих временных "бригад". Особенно остро этот вопрос касается прокладки вертикальных тоннелей.

Повторим, что рассмотренные модели можно назвать частично биоинспирированными, т.к. представления о характере деятельности объектов для подражания – насекомых, муравьев – носят зачастую весьма фрагментарный характер. Это вполне естественно: вряд ли биологи изучают поведение насекомых с целью описания алгоритма поведения для робота. Но это же заставляет упрощать модели поведения, привносить в них дополнительные факторы-гипотезы, т.е. делать все то, за что критикуют биоинспирированные подходы.

Список литературы

- [Брайен, 1986] Брайен М. Общественные насекомые. Экология и поведение. – М.: Мир, 1986. – 396 с.
- [Длусский, 1967] Длусский Г.М. Муравьи рода Формика. – М.: Наука, 1967. – 233 с.
- [Длусский, 1981] Длусский Г.М. Муравьи пустынь. – М.: Наука, 1981. – 230 с.
- [Захаров, 2018] Захаров А.А. Муравей. Семья. Колония. – М.: Фитон XXI, 2018. – 192 с.
- [Захаров и др., 2015] Захаров А.А., Захаров Р.А., Федосеева Е.Б. Использование параметров гнезда рыжих лесных муравьев в мониторинге муравейников // Проблемы экологического мониторинга и моделирования экосистем. – 2015. – Т. 26, № 1. – С. 68-90.
- [Czyzowicz et al., 2021] Czyzowicz J., Dereniowski D., Pelc A. Building a Nest by an Automaton. // Algorithmica. – 2021. – Vol. 83, No. 1. – P. 144-176.
- [Faria Dias et al, 2021] Faria Dias P.G. и et al. Swarm Robotics: A Perspective on the Latest Reviewed Concepts and Applications // Sensors. – 2021. – Vol. 21, No. 6. – P. 1-30.
- [Huang et al, 2022] Huang L.R. et al. Construction and Excavation by Collaborative Double-Tailed SAW Robots // IEEE Robot. Autom. Lett. – 2022. – Vol. 7, No. 2. – P. 3742-3748.
- [Parker et al., 2003] Parker C.A.C., Zhang H., Kube C.R. Blind bulldozing: Multiple robot nest construction. – 2003. – Vol. 2. – P. 2010-2015.
- [Reynolds, 1987] Reynolds C.W. Flocks, herds, and schools: A distributed behavioral model // Proc. 14th Annu. Conf. Comput. Graph. Interact. Tech. SIGGRAPH 1987. – 1987. – Vol. 21, No. 4. – P. 25-34.
- [Werfel, 2012] Werfel J. Collective Construction with Robot Swarms // Morphogenetic Engineering, Toward Programmable Complex Systems. – Springer, 2012. – P. 115-140.

ИНТЕРАКТИВНЫЙ СЕРВИС ВИЗУАЛИЗАЦИИ ОБРАЗОВ СЛОЖНЫХ ОБЪЕКТОВ МЕТОДОМ УНИФИЦИРОВАННОГО ГРАФИЧЕСКОГО ВОПЛОЩЕНИЯ АКТИВНОСТИ

М.В. Кузнецов (*kurt21.89@mail.ru*)^A

В.А. Углев (*uglev-v@yandex.ru*)^{A,B}

^A Кафедра прикладной физики и космических технологий
Сибирского федерального университета, Железногорск

^B Отделение интеллектуальных систем в гуманитарной сфере
Российского государственного гуманитарного университета, Москва

В работе описывается инструментальное решение одного из подходов Data Visualization из Data Mining – метода унифицированного графического воплощения активности (UGVA). Приводится обобщённая модель формализации исходных данных и кратко описывается технология их отображения на части антропоморфного образа. Обсуждаются интерактивные возможности сервиса и возможности его интеграции в аналитические системы. На примере формирования образов работников научно-образовательной организации показано то, как метод UGVA может быть применен в интеллектуальных системах поддержки принятия решений.

Ключевые слова: Data Mining, когнитивная визуализация, метод унифицированного графического воплощения активности, интеллектуальная система поддержки принятия решений.

Введение

Комплексное представление объектов в виде графических образов – это базовое назначение всех методов Data Visualization из такого раздела искусственного интеллекта, как интеллектуальный анализ данных (Data Mining [Witten, 2025]). При этом если сами данные описываются с позиции структуры (параметрические модели), то для этого используются хорошо зарекомендовавшие себя подходы (кластеризация [Han, 2011], упругие карты [Зиновьев, 2000], самоорганизующиеся карты [Kohonen, 2013] и пр.). Но не только нужно на образе показать динамику, активность или развитие,

164

все эти методы становятся недостаточно выразительны. При этом всегда остаётся потребность в усилении объясняющей способности результата визуализации, соответствуя тренду ХАI-подхода [Argieta, 2020].

Фокусировка на структуре объекта отображения является слабым местом многих методов когнитивной визуализации. Частично эту проблему решает использование пиктографирования. Оно заключается в том, что образ формируется в виде сложной композиции элементов, имеющих отвлечённое отношение к исходным данным. Классическими примерами таких методов являются или. При этом совместить возможность отображения динамики (активности) образа (например, дашборды [Few, 2006]) и возможности множественного сравнения объектов (например, лица Чернова [Chernoff, 1973]) объединить в рамках одного подхода достаточно сложно [Uglev, 2024a].

Целью данной статьи является краткое описание метода унифицированного графического воплощения активности (UGVA), ориентированного на отражение динамики сложных объектов, а также интерактивного сервиса, реализующего этот метод. Для этого опишем особенности антропоморфных образов в Data Visualization, дадим краткую характеристику методу UGVA, опишем сервис по построению образов, а также приведём результаты практического применения метода при реализации интеллектуальных систем поддержки принятия решений.

1. Антропоморфные образы как инструмент Data Mining

Антропоморфные образы в пиктографике изначально делятся на две группы: отображение эмоции (концентрация на лице) или всей фигуры человека (концентрация на пропорциях тела). Первый подход развивался в работах Джоунса [Jones, 1982] и Чернова [Chernoff, 1973] (см. пример на рис. 1,а). Его достоинства – быстрое выделение отличий, передача общей оценки состояния объекта визуализации. Но, с другой стороны, он требователен к симметрированию на уровне данных, их полноте и низкой степени детализации (подробнее см. в [Uglev, 2024a]).

Второй подход, опирающийся на целостную фигуру человека, наоборот, не имел интегральной зоны и результативнее отображал сложные наборы исходных данных. При этом уровень детализации мог быть существенно повышен, а скорость восприятия образа не сильно уступала тем же лицам. В работах Да-Винчи (*Homo vitruvianus*), Ле Корбюзье [Le Corbusier, 1948] (см. пример на рис. 1,б) и других этот метод стал применяться больше для иллюстрации, чем в задачах принятия решений.

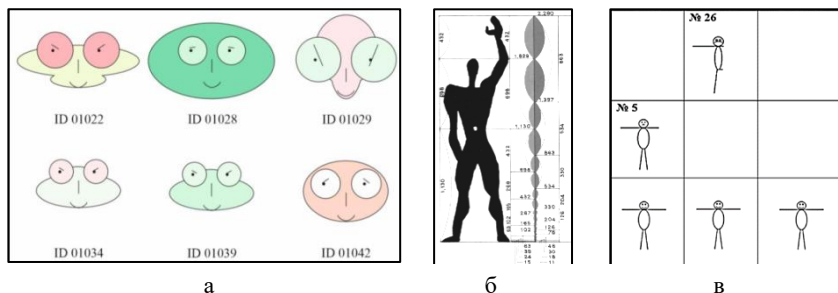


Рис. 1. Примеры антропоморфных образов в пиктографике

Объединить оба подхода удалось в работах В.А. Филимонова, предложившего сначала приложение бодикодер [Филимонов, 2015], а затем и введя образ модулятора Тризкина (см. пример на рис. 1,в, [Филимонов, 2021]). Совмещающая возможность передачи эмоции (интегральной оценки) и большого разнообразия элементов образа, данный подход позволил не только визуализировать сложный объект, но и сравнивать объекты между собой (по аналогии с [Chernoff, 1973]).

Таким образом, решение проблемы концентрации данных, их отображения и сравнения была в пиктографике решена за счёт использования легко воспринимаемых контуров – антропоморфных фигур. Но осталась проблема отображения динамики. Попытки использования анимации, кодирования неопределённости данных и цветовых шкал (см. обзор в [Nuñez, 2010]) не дали существенного улучшения. В первую очередь это связано с тем, что на уровне модели исходных данных активность объекта визуализации не рассматривалась как важная компонента [Uglev, 2022]. Введем модель, учитывающую динамику.

Описание динамики показателей сложного объекта, как активности, требующей визуализации, может быть представлено как совокупность четырёх планов описания: совокупность прецедентов активности A (цифровой след); совокупность признаков P (структурная составляющая); совокупность целей и аспектов деятельности G (классификатор активности) и множество планов анализа Q . Тогда P и G можно представить в виде графа типа дерево (граф с вертикальным типом связей), компонент Q будет реализовывать смысловое объединение элементов графа за счёт горизонтальных связей, отражающих семантику и экспертные оценки), а данные из A будут представлять из себя временной ряд [Uglev, 2024a]. Модель активности объекта u_k будет записана в виде кортежа (1)

$$u_k = \langle A|P, G, Q \rangle, \quad (1)$$

где k – индекс объекта, подлежащего визуализации.

Если провести аналогию с семиотическим представлением [Поспелов, 1999], то тройка $\langle P, G, Q \rangle$ будет соответствовать синтактике, прагматике и семантике соответственно. Денотатом будет сущность u_k в памяти аналитической системы, детализирующейся за счёт компонента A , формируя квадрат Поспелова.

Совокупность компонентов u_k из (1) будет отображать активность в том случае, если основой для формирования образа послужит граф G , т.е. будет использован функционально-структурный подход к анализу объектов [Балашов, 1985]. Это значит, что ключевые оси на образе, будут отражать группы активностей, принадлежащих одной ветви графа и имеющей схожую целевую обусловленность. Это заставляет поставить вопрос о рассмотрении такой графической нотации антропоморфного образа, которая будет результативно отображать предложенную модель из (1) относительно G , но инвариантно Q .

2. Метод UGVA

Метод Unified Graphic Visualization of Activity – это такой подход к визуальному представлению профиля сложного многопараметрического объекта, выраженного в виде антропоморфного образа, с целью показать особенности его функциональных (activity) и зависимых от них структурных параметров, а также результативного сравнения таких профилей между собой [Uglev, 2022b].

В основе антропоморфного образа лежит параметрическая модель той предметной области, для которой требуется произвести концентрацию данных. На образе выделяются оси, страты, определяется тип симметрирования [Weyl, 2015] и прочие параметры. Предполагается, что последовательно формируются следующие типы образов, усложняющиеся и дополняющиеся по мере решения задачи анализа: каркас, архитектурный образ, базовый образ, индивидуализированный образ, активный образ, групповая карта. Два примера групповых карт в нотации UGVA приведены на рис. 2.

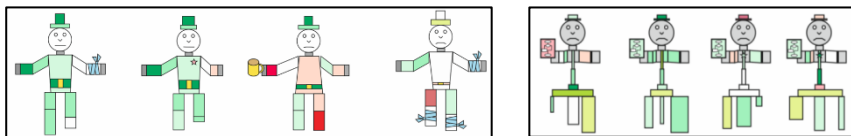


Рис. 2. Примеры групповых карт антропоморфных образов для работников предприятия (слева) и для группы учащихся (справа)

Формируется антропоморфный образ за 9 шагов, которые были подробно изложены в [Углев, 2023]. Для интерпретации картируемой группы объектов требуется не только рассматривать соотношение линейных раз-

меров частей образов, но и симметричность, «выражение лиц», наличие артефактов, характер тепловой карты (раскраски). Таким образом, на основании системно проработанной модели активности, формируется карта экземпляров объектов, позволяющая решать задачи диагностики, мониторинга и сравнения (выявления особенностей/аномалий).

Метод UGVA, по аналогии с лицами Чернова, применяется как для визуализации активных объектов (людей, машин, организаций и пр.), так и для сложных процессов (проектов, оперативной обстановки и пр.). Далее рассмотрим интерактивный сервис по формированию карт и обратимся к одному из примеров.

3. Сервис построения антропоморфных образов

Автоматизированное построение образа в нотации UGVA было реализовано в виде сервиса в 2020 году. В его возможности входило формирование и раскраска образа по вектору входных данных (без возможности интерактивного взаимодействия).

В 2024 году был разработан новый сервис, позволяющий интегрировать образы на страницы с профилями интеллектуальной систем поддержки принятия решений. Поддерживается возможность загрузки пакетных данных из баз данных активностей (см. пример описания события на рис. 3,а), выделение мышью элементов образа (с выводом уточняющей информации), автоматическое построение групповых карт по данным из базы проекта, настройка шаблонов отображения (рис. 3,б).

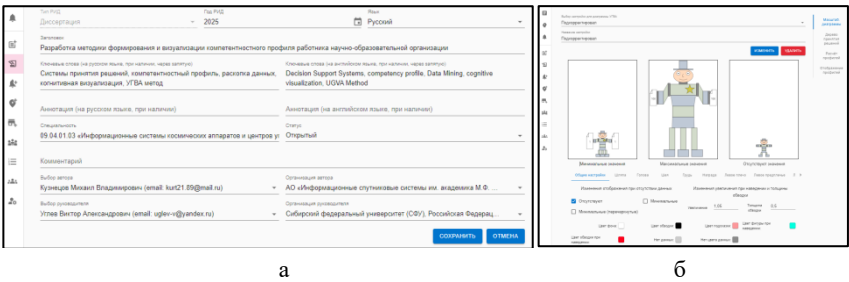


Рис. 3. Примеры интерфейса ввода первичных данных и формирования шаблона образа в нотации UGVA

Для разработки серверной части приложения выбран язык программирования PHP, в качестве серверного фреймворка использован Laravel (с поддержкой шаблонов проектирования Model-View-Controller). Клиентская часть реализована на JavaScript (библиотека React), CSS и HTML. В качестве технической основы для синтеза интерактивных образов в нотации UGVA выбран элемент HTML5 – canvas, предоставляющий воз-

возможности для программной отрисовки разнообразных графических примитивов, таких как линии, многоугольники, кривые и текстовые элементы. Использование Konva.js обеспечивает высокую производительность и гибкость визуализации, что критически важно для динамического и интерактивного отображения данных компетентностного профиля.

Визуальный образ активного объекта реализован в прямоугольной области фиксированных размеров. Поддерживает масштабирования под размеры окна приложения и динамически пересчитывает размеры элементов в зависимости от значений всех показателей в базе цифровых следов. Это делается для того, чтобы вычислить комплексную нормализованную оценку каждого образа в диапазоне от 0 до 1 с учётом распределения оценок всех образов, что позволит объективно позиционировать каждый образ в составе групповой карты. Нормализация данных, применяемая при $I > I_{min}$, преобразует все оценки в диапазон от 0 до 1 по формуле (2).

$$I_{\text{норм}} = (I - I_{\min}) / (I_{\max} - I_{\min}), \quad (2)$$

где $I_{\text{норм}}$ – нормализованное значение интегральной характеристики; I – значение характеристики образа; I_{\min} – минимальное значение критерия для параметра; I_{\max} – максимальное значение критерия для параметра.

Цветовая маркировка на образах используется в шкале от зеленого до красного с переходом в центре через белый: красный цвет обозначает низкое качество активностей в текущей группе показателей, зеленый обозначает высокое качество, белый отражает нейтральное значение, а переходы от белого к красному или зеленому показывают степень отклонения оценок признаков от нейтральных.

4. Пример экспериментального применения

Возьмём в качестве примера использования интерактивного сервиса по построению антропоморфных образов в нотации UGVA задачу анализа активности работников подразделения научно-образовательной организации. Сама проблема выявления «точек роста» при управлении подразделениями подобного типа упирается в отсутствии общепризнанной номенклатуры работ и норм показателей [Гаврилова, 2003 и 2024]. Проведя исследование [Uglev, 2024b], мы остановились на следующей структуре показателей цифрового следа сотрудника:

- классификатор активностей P был представлен тремя типами следов: научными (научные интересы, результаты интеллектуальной деятельности, публикационная активность и участие в научно-исследовательских работах и пр.), педагогическими (научной работой с аспирантами и магистрантами, читаемыми дисциплинами, учебно-методическими материалами, выпускными работами подопечных, и прочей активностью в рамках университетской работы), личностно-профессиональными (цели, образование, курсы повышения квалификации, роли на работе вне кафедры, экспертная работа и пр.).

- цели и аспекты деятельности G включали личные цели и планы, развитые относительно образовательному, профессиональному или научному аспектам;
- планами анализа Q стали кафедра точка зрения, университетская точка зрения, точка зрения индустриального предприятия-партнёра (работодателя) и обобщённый план анализа.

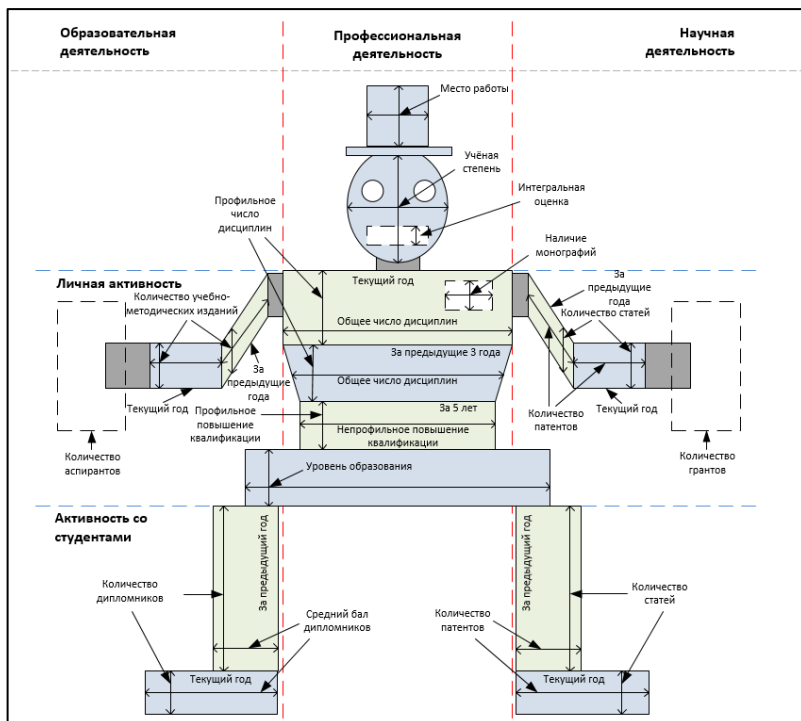


Рис. 4. Соотнесение параметров профиля элементам образа в нотации UGVA

В результате, получилась параметрическая модель из 18 обобщённых параметров, подавляющая часть из которых была представлена в делении на текущие (за этот год) и наработанные ранее (за предыдущие 5 лет). Распределение параметров по вертикальным стратам (аспекты) и горизонтальным (личная активность, активность в работе с учащимися и интегральная зона – голова) приведена на рис. 4. Согласно классификации из [Углев, 2023], данный образ имеет тип $\lambda 5Eit$.

Для примера возьмём научно-педагогический состав работников кафедры «Прикладная физика и космические технологии» Сибирского федерального университета. Исходная информация по сотрудникам кафедры

и приписанной к ней научно-учебной лаборатории вводилась в интеллектуальную систему поддержки принятия решений (данные хранятся в базе данных MySQL). Опиерируя совокупностью преобразованных и нормализованных по (2) данных из базы активностей сотрудников (компонент A из формулы (1)), сервис автоматически формирует антропоморфный образ, актуальный на дату запроса.

Антропоморфные образы, в качестве примера, для трёх научно-педагогических работников приведены на рис. 5 за 2024 и 2025 года: слева направо приведен пример профессора, доцента и аспиранта. Здесь выбран обобщённый план анализа, минимизирующий оценочную раскраску.

Если оценивать качества отдельных работников, то сразу бросается в глаза несимметричность образов: каждый работник имеет более «привычные» сферы активности, которые следует учитывать при формировании годовых планов. Но если давать оценку подразделения в целом (по всем образам работников, а не только трём приведённым на рис. 5), то в качестве рекомендаций сразу можно выделить необходимость увеличения публикационной активности и перераспределение нагрузки по научному руководству диссертациями магистрантов в сторону тех, кто более результативен для планов подразделения.

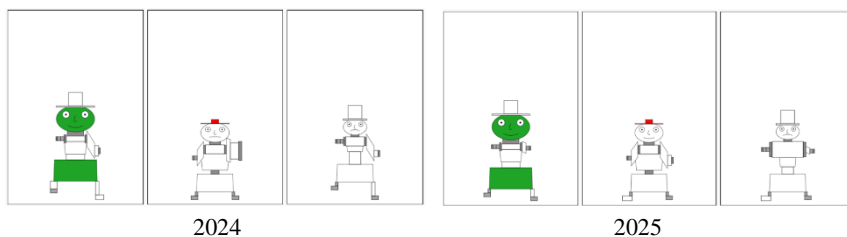


Рис. 5. Пример образов трёх научно-педагогических работников кафедры в виде групповой карты в нотации UGVA, построенные с интервалом в год

Иные примеры использования образов в нотации UGVA, хотя и реализованные на предыдущей версии сервиса, опирались на модели научных работников Сибирского отделения РАН [Uglev, 2024a], сотрудников предприятий Росатома [Мешков, 2024], учебных планов подготовки специалистов в области информатики и вычислительной техники [Uglev, 2024b] и системного анализа [Проценко, 2021]. Обобщение опыта использования метода UGVA приведено в работе [Uglev, 2024a].

Заключение

Метод унифицированного графического воплощения активности (UGVA) позволяет методически обосновать, а затем и реализовать визуализацию активности сложных систем. Приведённое описание интерактивного

сервиса по формированию и визуализации антропоморфных образов поддерживает интеграцию в сетевые проекты систем поддержки принятия решений. При этом поддерживаются не только единичные профили, но и построение карт образов, что отвечает главному назначению метода – обеспечение мониторинга и контроля, выявления аномалий и закономерностей.

Использование метода UGVA сейчас реализуется как в рамках киберсоциальных систем, так и в виде простого средства визуализации. Наибольшее число моделей активности нами было проработано для сопровождения электронного обучения на базе интеллектуальных автоматизированных обучающих систем. В данный момент осуществляется интеграция механизма оценки параметров образов с технологией рекомендательных (экспертных) систем с элементами нечёткой логикой [Uglev, 2024c].

Список литературы

- [Arrieta, 2020] Arrieta A., Díaz-Rodríguez N., Del Ser et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI // Information fusion. – 2020. – Vol. 58.
- [Chernoff, 1973] Chernoff H. The use of faces to represent points in k-dimensional space graphically // Journal of the American Statistical Association. – 1973. – Vol. 68. – doi: 10.1080/01621459.1973.10482434.
- [Few, 2006] Few S. Information dashboard design: The effective visual communication of data. O'Reilly Media, Inc., 2006.
- [Han, 2011] Han J., Kamber M., Pei J. Data Mining: Concepts and Techniques Third Edition // The Morgan Kaufmann Series in Data Management Systems. – 2011. – Vol. 5, No. 4.
- [Jones, 1982] Jones D. The Inventions of Daedalus: A Compendium of Plausible Schemes. WH Freeman, 1982.
- [Kohonen, 2013] Kohonen T. Essentials of the self-organizing map // Neural networks. – 2013. – Vol. 37. – P. 52-65.
- [Le Corbusier, 1948] Le Corbusier (C.E. Jeanneret). Le Modulor. Boulogne-sur-Seine: Editions de l'Architecture d'Aujourd'hui, Groupe Expansion, Paris, 1948.
- [Nuñez, 2010] Nuñez J. Ideas para el uso de las caras de Chernoff en la cartografía escolar // Boletim de Geografia. – 2010. – 28(1). – P. 5-15.
- [Uglev, 2022] Uglev V. Unified Graphic Visualization of Activity (UGVA) Method. Novel and Intelligent Digital Systems. LNNS, 2022. – 556. – P. 255-265. – doi: 10.1007/978-3-031-17601-2_25.
- [Uglev, 2024a] Uglev V., Meshkov S., Kuznetsov M. Methodology of Complex Visual Representation of Human Activity Using Cognitive Visualization // Journal of Integrated Design and Process Science. – 2024. – Vol. 27(3-4). – P. 184-199. – doi: 10.1177/10920617241289768
- [Uglev, 2024b] Uglev V., Kuznetsov M., Meshkov S. An approach to the formation and visualization of the competency profile of the staff of organizations using the UGVA method // Computational Science and Its Applications – ICCSA 2024 Workshops. LNCS. – Vol 14814. – Springer, Cham. – P. 170-185. – doi: 10.1007/978-3-031-64608-9_11.

- [Ughev, 2024c] Ughev V.A. Implementation of Decision-Making Mechanism in the Intelligent Tutoring System Based on the Expert Systems Module // Pattern Recognition and Image Analysis. – 2024. – Vol. 34, No. 3. – P. 744-750. – doi: 10.1134/S1054661824700615.
- [Weyl, 2015] Weyl H. Symmetry. Princeton University Pres, 2015.
- [Witten, 2025] Witten I.H. et al. Data Mining: Practical machine learning tools and techniques. – Elsevier, 2025.
- [Балашов, 1985] Балашов Е.П. Эволюционный синтез систем. – М.: Радио и связь, 1985.
- [Гаврилова, 2003] Гаврилова Т.А., Кузнецова А.В. Лещева И.А. К вопросу о разработке онтологии научной и учебной работы // XXI национальная конференция по искусственному интеллекту с международным участием (КИИ-2023). В 2 т. Т 1. - Смоленск: Принт-Экспресс, 2023. – С. 60-67.
- [Гаврилова, 2004] Гаврилова Т.А., Алканова О.Н., Гринберг Э.Я., Кузнецова А.В. Визуализация компетенций сотрудников с помощью карт знаний // Российский журнал менеджмента. – 2024. – № 1. – С. 86-112.
- [Зиновьев, 2000] Зиновьев А.Ю., Питенко А.А. Визуализация произвольных данных методом упругих карт // Радиотехника, информатика, управління. – 2000. – № 1(3).
- [Мешков, 2024] Мешков С.С. Методика визуального мониторинга проектных работ первичных трудовых коллективов на базе метода UGVA // Прогрессивные научные исследования – основа современной инновационной доктрины: Материалы Международной конференции. – Уфа: OMEGA SCIENCE, 2024. – С. 50-54.
- [Поспелов, 1999] Поспелов Д.А., Осипов Г.С. Прикладная семиотика // Новости искусственного интеллекта. – 1999. – № 1. – С. 9-35.
- [Проценко, 2021] Проценко Д.А. Сравнение учебных планов магистратуры направления «Системный анализ и управление» методом UGVA // Преподавание информационных технологий в Российской Федерации: сборник научных трудов; материалы Девятнадцатой открытой Всероссийской конференции. – М.: АПКИТ, 2021. – С. 173-174.
- [Углев, 2022] Углев В.А. Визуальная поддержка принятия решений при синтезе учебных планов с помощью метода UGVA // Искусственный интеллект и принятие решений. – 2022. – № 2. – С. 51-61. – doi: 10.14357/20718594220205.
- [Углев, 2023] Углев В.А. Поддержка процесса принятия решений с использованием нотации унифицированного графического воплощения активности (UGVA) // Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение. – 2023. – № 3(144). – С. 125-140. – doi: 10.18698/0236-3933-2023-3-125-140.
- [Филимонов, 2015] Филимонов В.А. Сума технологии: три пятилетки междисциплинарного проекта // Знания-Онтологии-Теории (ЗОНТ-2015): Материалы Всероссийской конференции с международным участием. – Новосибирск: Ин-т математики им. С.Л. Соболева СО РАН, Т. 2. – С. 181-188.
- [Филимонов, 2021] Филимонов В.А. Способ когнитивной визуализации многопараметрических компонентов системы // Робототехника и искусственный интеллект: Материалы XIII Всероссийской конференции с международным участием. – Красноярск: Литера-принт, 2021. – С. 109-113. – URL: https://aesfu.ru/local/conference/_docs/2021/RAI-21_print.pdf.

АЛГОРИТМЫ МНОГО-АГЕНТНОГО ПЛАНИРОВАНИЯ ТРАЕКТОРИИ С УЧЕТОМ ВРЕМЕНИ, ЗАТРАЧИВАЕМОГО НА ПОВОРОТЫ

А.И. Луценко (*ailutsenko@edu.hse.ru*)^А

К.С. Яковлев (*yakovlev@isa.ru*)^{А,В}

^А Национальный исследовательский университет

«Высшая школа экономики», Москва

^В Федеральный исследовательский центр

«Информатика и управление» РАН, Москва

В работе рассматривается задача много-агентного планирования путей на графе регулярной декомпозиции. В частности, исследуется один из возможных вариантов постановки этой задачи, а именно, вариант, когда агенты затрачивают на поворот время, равное времени, затрачиваемому на перемещение в смежную вершину. Предлагается несколько оригинальных алгоритмов решения задачи в этой постановке, являющихся модификациями известного вычислительно-эффективного алгоритма много-агентного планирования РИВТ. Приводятся результаты экспериментальных исследований.

Ключевые слова: много-агентное планирование, граф, робот, кратчайший путь, поиск пути в графе.

Введение

Задача много-агентного планирования путей – это задача, в которой по графу и множествам начальных и целевых вершин необходимо построить совокупность путей для нескольких агентов таким образом, что агенты во время следования вдоль этих путей не сталкиваются между собой.

Много-агентное планирование путей имеет множество приложений, например, в организации беспилотного транспорта, но ее самое частое применение сегодня – организация совместной работы роботов на автоматизированных складах, когда им надо перевозить товары из одного места в другое.

Задача уже широко изучена и существует множество алгоритмов, ее решающих. Так, например, популярным оптимальным алгоритмом является CBS [Sharon et al., 2015]. На его основе разработаны алгоритмы

ECBS [Barer et al., 2014] и EECBS [Li et al., 2021], которые находят решение, отличающееся от оптимального в фиксированное число раз, но при этом данное решение находится быстрее, чем в CBS.

Популярной идеей является построение алгоритмов на правилах: для агентов составляются правила, по которым они могут действовать (например, правило, обменивающее агентов местами) и с помощью этих правил задача решается. Такая идея применяется в статьях [De Wilde et al., 2014], [Sajid et al., 2012] и [Khorshid et al., 2011]. Решение находится быстро, но часто сильно отличается от оптимального в смысле качественных характеристик.

Предложенные алгоритмы хороши в своих областях, однако, они пренебрегают некоторыми физическими параметрами системы и решения получаются неприменимыми на практике. В данной работе рассматривается усложненная задача много-агентного планирования – много-агентное планирование с поворотами. Необходимость рассмотрения этой задачи диктуется практикой, т.к. в реальности большинство складских роботов затрачивают время на поворот на месте. Следовательно если пути для подобных роботов конструировались без учета этого обстоятельства, то безопасное следование по ним невозможно.

Для много-агентного планирования с поворотами уже был предложен оптимальный алгоритм CBS [Zhang et al., 2023], однако данный алгоритм из-за того, что является оптимальным, решает задачу за длительное время даже для небольшого числа агентов. В работе предлагаются алгоритмы PIBT_T, Turned PIBT и WinPIBT_T, основанные на алгоритме PIBT [Okumura et al., 2022], который пошагово решает задачу много-агентного планирования без поворотов. Он выбран за основу данных алгоритмов, так как решает задачу быстро, получая при этом не слишком плохие по качеству решения.

1. Постановка задачи

Определим задачу MAPF_T – задачу много-агентного планирования с поворотами.

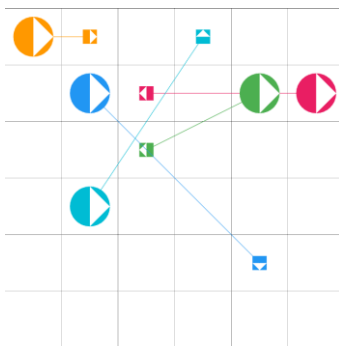


Рис. 1. Пример постановки задачи: круги – начальные положения агентов, квадраты – конечные их положения

Ее вход можно задать тройкой $\langle G, s, f \rangle$, где $G = (V, E)$ – неориентированный граф, $s = (s_1, s_2, \dots, s_n) \in V^n$ – стартовые позиции агентов, а $f = (f_1, f_2, \dots, f_n) \in V^n$ – конечные позиции агентов.

Граф в этой задаче задается на квадратной решетке, то есть $V \subset \mathbb{N}_{\geq 0}^2$, а множество соседей вершины $v = (x, y)$ является подмножеством $\{(x+1, y), (x-1, y), (x, y+1), (x, y-1)\}$, т.е. E состоит из ребер вида $((x, y), (x+1, y))$ и $((x, y), (x, y+1))$

Определим множество возможных направлений $D = \{X^+, X^-, Y^+, Y^-\}$. Ответ задается набором π из n путей. π_i – набор из $|\pi_i|$ пар $\langle v, d \rangle \in V \times D$ – текущая вершина и направление агента.

Правила для π :

1. Для любого $i = 1, \dots, n$ и $j = 1, \dots, |\pi_i| - 1$ верно одно из:
 - 1.1. $\pi_i[j].d = \pi_i[j+1].d = d$ и переход происходит в направлении d , то есть если $d = X^+$, то $\pi_i[j].v.y = \pi_i[j+1].v.y$, а $\pi_i[j].v.x + 1 = \pi_i[j+1].v.x$, для других направлений правила аналогичны
 - 1.2. $\pi_i[j].v = \pi_i[j+1].v$, но при этом направления $\pi_i[j].d$ и $\pi_i[j+1].d$ отличаются на 90° , то есть одно имеет вид X_γ , а другое – Y_γ .
 - 1.3. $\pi_i[j] = \pi_i[j+1]$, то есть агент не двигается
2. Для любого $i = 1, \dots, n$ верно, что $\pi_i[1] =$, а $\pi_i[t] = \langle f_i, d_i \rangle$ (d_i – любое направление из D), то есть s_i задает начало пути i -ого агента, а f_i – конец его пути.
3. Для любых двух агентов $a = 1, \dots, n$, $b = 1, \dots, n$ ($a \neq b$) в любой момент времени $j = 1, \dots, t$ верно, что $\pi_a[j].v \neq \pi_b[j].v$, то есть никакие два агента не находятся в одной вершине графа одновременно
4. Для любых двух агентов $a = 1, \dots, n$, $b = 1, \dots, n$ ($a \neq b$) в любой момент времени $j = 1, \dots, t - 1$ не могут одновременно выполняться условия $\pi_a[j].v = \pi_b[j+1].v$ и $\pi_a[j+1].v = \pi_b[j].v$, то есть никакие два агента не могут одновременно проходить по одному ребру (даже в разных направлениях)

При этом задачей алгоритмов является минимизация следующих метрик:

- *Makespan* – минимальная длина пути, то есть t ;
- *Sum-of-costs* – суммарная длина всех путей, то есть $\sum_{i=1}^n |\pi_i|$.

2. Алгоритмы

В основе предложенных алгоритмов лежит алгоритм PIBT [Okumura et al., 2022]. В ходе работы данного алгоритма происходит пошаговое планирование положений агентов. Агентам сопоставляются случайные приоритеты, распределенные равномерно на отрезке $[0, 1]$. Каждый ход

приоритет агента возрастает на 1, а когда он достигнет своей цели, целая часть приоритета отбрасывается (остается лишь изначально случайно сгенерированная величина). В его основе лежат техники Priority Inheritance и Backtracking. Ход выбирается следующим образом: Сначала с помощью алгоритма A* для агента a определяется следующая клетка на его кратчайшем пути до цели. Алгоритм пытается передвинуть a в эту клетку. Если она уже занята каким-либо другим агентом, то происходит Priority Inheritance: этот агент временно получает приоритет агента a и алгоритм также пытается найти ему клетку, в которую можно перейти, возможно, применяя Priority Inheritance. После этого, агент сообщает об успешности передвижения агенту a . Такое обратное взаимодействие агента и агента a называется Backtracking. В случае неуспешного завершения Priority Inheritance, a начинает искать себе другую клетку, в которую он может перейти. Данный алгоритм хорошо показывает себя на практике: решает задачу быстро и предъявляет хорошие по качеству решения

2.1. Turned PIBT

Чтобы получить алгоритм, который решает задачу в хорошем числе случаев, предлагается взять задачу MAPF_T, убрать из нее направления, получив тем самым задачу MAPF, решить ее любым решателем, например, PIBT, и вернуть в полученное решение направления.

Алгоритм 1 Преобразователь решения MAPF в решение MAPF_T

Вход: π — решение MAPF (набор путей, $\pi_i[t]$ — положение агента i в момент t)

Выход: π_T — решение MAPF_T ($\pi_{T_i}[t].v$ — положение агента i в момент t , $\pi_{T_i}[t].d$ — его направление в этот момент)

```

1: for  $i \in 0, \dots, |\pi| - 1$  do
2:    $\pi_{T_i}[0].v \leftarrow \pi_i[0]$ 
3:    $\pi_{T_i}[0].d \leftarrow X_+$  ▷ Считаем, что агенты изначально ориентированы вдоль
   положительного направления оси  $X$ 
4: end for
5: for  $t \in 0, \dots, \max |\pi_i| - 1$  do
6:    $\text{turns} \leftarrow [\emptyset, \dots, \emptyset]$ 
7:   for  $i \in 0, \dots, |\pi| - 1$  do
8:      $\text{next\_direction} \leftarrow \text{GET\_DIRECTION}(\pi_i[t], \pi_i[t + 1])$ 
9:      $\text{turns}[i] \leftarrow \text{ROTATE}(\pi_{T_i}[t - 1].d, \text{next\_direction})$ 
10:  end for
11:   $m \leftarrow \max |\text{turns}[i]|$ 
12:  for  $i \in 0, \dots, |\pi| - 1$  do
13:    for  $j \in 0, \dots, m - 1$  do
14:       $\pi_{T_i} \leftarrow \pi_{T_i} \cup (\pi_i[t], \text{turns}[i][j])$ 
15:    end for
16:  end for
17: end for
```

Алгоритм PIBT получает на выходе набор путей, не учитывающих повороты агентов. Далее, для получения корректного решения этот набор путей необходимо преобразовать в набор путей для задачи MAPF_T. Дан-

ную операцию выполняет Алгоритм 1. Он пошагово изменяет все пути следующим образом: берется каждый ход решения PIBT и для всех агентов, которые должны совершить поворот в этот ход, в ответ добавляется новое состояние с поворотом, а для всех остальных агентов добавляется последнее их состояние (агент остается на месте). Затем операция повторяется, если каким-либо агентам необходимо совершить разворот. После этого все агенты направлены в направлении из текущей клетки решения PIBT в следующую и в ответ добавляются состояния, переводящие нужных агентов вперед.

2.2. PIBT_T

Данный алгоритм является простейшей доработкой PIBT до алгоритма для задачи MAPF_T.

Агентам сопоставляются приоритеты, аналогичны приоритетам из алгоритма PIBT. Выбор следующей клетки происходит похожим образом: Сначала с помощью алгоритма A* для агента a определяется следующая клетка на его кратчайшем пути до цели. Если клетка находится не прямо перед агентом, то алгоритм поворачивает агента в направлении этой клетки. Иначе, алгоритм проверяет возможность продвижения агента a вперед. Если клетка свободна, то PIBT_T помечает клетку как зарезервированную для агента a . Если же клетка уже была зарезервирована для другого агента, то алгоритм пропускает данную клетку, выбирает кратчайший путь, не проходящий через эту клетку, и поворачивает агента к этой клетке, либо, оставляет агента на месте, если кратчайший путь через другие клетки оказывается длиннее. Также, алгоритму необходимо учесть случай, в котором клетку в данный момент занимает агент с более низким приоритетом. В этом случае алгоритм использует техники Priority inheritance и Backtracking аналогично PIBT.

2.3. WinPIBT_T

В ходе тестирования алгоритма PIBT_T было выявлено много проблем. Ключевой проблемой стало то, что алгоритм в некоторых случаях алгоритм не способен обойти одним агентом другого. Из этого стало ясно, что алгоритму при планировании необходимо принимать в расчет не только предыдущий ход, но и несколько ходов, предшествующих предыдущему. WinPIBT_T решает данную проблему, планируя пути агентов на несколько ходов вперед, как это делает алгоритм WinPIBT [Okumura et al., 2019]

2.3.1. Устройство алгоритма WinPIBT_T

Алгоритм устроен аналогично алгоритму PIBT_T. Агенты резервируют для себя положения вида $\langle t, v \rangle$ – время и вершина в будущем. Если в текущий момент в вершине v находится агент, то происходят Priority Inheritance и Backtracking.

2.3.2. Техника откатов в WinPIBT_T

Из-за того, что алгоритм планирует пути на несколько ходов в будущее могут возникать особые конфликты (рис. 2)

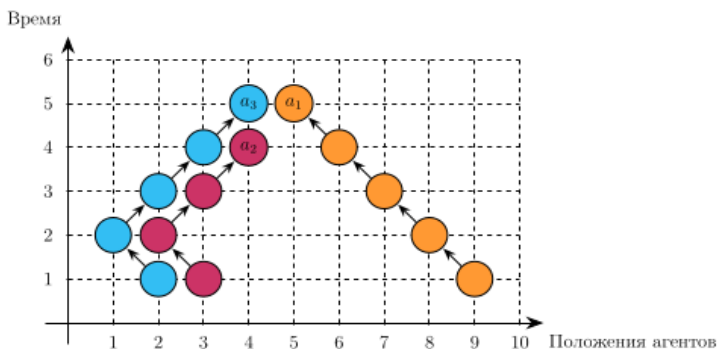


Рис. 2: Конфликт в алгоритме WinPIBT_T

Так, на рис. 2 представлена гипотетическая ситуация, в которой агенты могут планировать ходы на 4 хода вперед, а граф – полоска 1×10 . Агенты a_1 и a_3 успешно спланировали для себя пути, а агент a_2 не может продвинуться далее. Для обработки таких ситуаций в алгоритм добавлена система откатов: если алгоритм не может спланировать агенту ход, то он начинает постепенно удалять последние ходы этого агента до тех пор, пока не найдется клетка, в которой агент может простоять до конца планируемого периода. При нахождении такой клетки, агент остается в ней до конца планируемого периода. Если же такой клетки не нашлось, то агенту восстанавливаются все удаленные клетки из его пути, удаляются из остальных путей все клетки, которые зарезервированы в ходы, позже последнего хода рассматриваемого агента. Рассматриваемый агент получает самый высокий приоритет и алгоритм продолжается. Самый высокий приоритет позволяет данному агенту гарантированно пройти вперед на следующем ходу.

3. Эксперименты

Предложенные в работе алгоритмы реализованы на языке C++. Эксперименты проводились на нескольких картах из бенчмарков из известного набора MAPF-бенчмарков [Stern et al., 2019].

Каждый тест состоит из сценария – карты и набора положений агентов. Для сбора статистики по алгоритмам перебирается текущее число агентов (a) и всем алгоритмам на вход подаются первые a агентов из сценария. Алгоритмы запускаются на переданных данных, у них замеряется

время работы, собираются метрики полученных решений (процент успешных запусков – Success rate, сумма длин путей агентов – Sum of costs и максимальный пройденный путь – Makespan). Эксперименты проводятся на трех картах: Random (на квадратной карте препятствия расставлены случайным образом), Warehouse (складское помещение) и Den312d (карта из игры Dragon Age: Origins. На картах этой игры часто проводят замеры для анализа алгоритмов MAPF).

Запуск тестов происходит на процессоре Intel(R) Core(TM) i5-1038NG7 CPU @ 2.00GHz, 16 ГБ ОЗУ. Исходный код доступен в репозитории https://github.com/LutsenkoAnton/pibt_t.

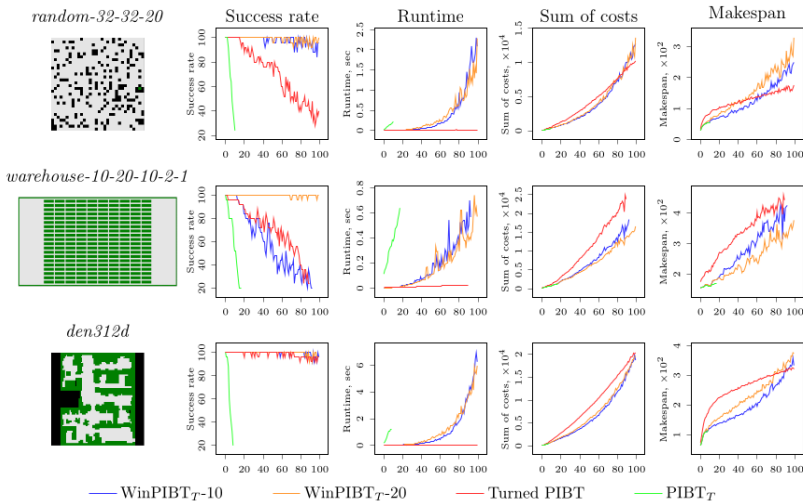


Рис. 3: Результаты экспериментов

На рис. 3 представлены результаты экспериментов. PIBT_T показывает неплохие Makespan и Sum-of-costs, но его success rate падает очень быстро.

Сравним теперь Turned PIBT и WinPIBT_T. Графики времени работы алгоритмов выглядят одинаково на разных картах: с ростом числа агентов время работы WinPIBT_T начинает резко возрастать. Действительно, это довольно предсказуемый результат: WinPIBT_T пытается спланировать больше ходов, обработка конфликтов для него дороже, а с ростом числа агентов число конфликтов возрастает. Однако, для 100 агентов видим, что время выполнения занимает несколько секунд, что вполне приемлемо для применения на практике.

На всех предложенных графиках WinPIBT_T-20 получает success rate, близкий к 100%. Это является его несомненным преимуществом над Turned PIBT: как видим, Success rate у Turned PIBT на двух из трех графиков падает почти до 20% – границе, после которой запуск алгоритма в данных экспериментах заканчивался.

Как можно видеть, метрики решения на карте Random у WinPIBT_T очень хорошие: Sum of costs примерно такой же, как у Turned PIBT, а Makespan в итоге оказывается в несколько раз больше. Вероятно, такой рост в Makespan обусловлен стратегией откатов: агент, не нашедший дальнейшего пути может потерять 20 ходов, если не найдет выхода.

Графики для Den312d похожи на графики для Random. Для небольшого числа агентов WinPIBT_T дает выигрыш, но в конце получается решение, схожее по характеристикам Turned PIBT.

Перейдем к рассмотрению ключевого случая: Warehouse. Алгоритм служит для планирования путей агентов на складах, поэтому этот тест особенно важен для данной работы. В этих тестах мы видим, что метрики у WinPIBT_T получаются меньше, чем у Turned PIBT. Также легко видеть, что Success rate у WinPIBT_T-20 примерно равен 100%, а у Turned PIBT и WinPIBT_T-10 постепенно падает. Это связано с тем, что карта, на которой проходит тестирование имеет много узких коридоров. Агентам, чтобы в них разойтись, надо пройти большое расстояние (объехать полки с товарами). Очевидно, Turned PIBT с такой задачей не справляется. WinPIBT_T-10 тоже не хватает планирования для объезда таких препятствий, а WinPIBT_T-20 спокойно справляется с задачей.

Заключение

В работе представлено три алгоритма, решающих задачу много-агентного планирования с поворотами: PIBT_T, Turned PIBT и WinPIBT_T. Проведено сравнение алгоритмов, в ходе которых показаны достоинства и недостатки этих алгоритмов. В ходе сравнения было подтверждено, что WinPIBT_T работает лучше альтернатив на основной карте: карте складского помещения, при этом стабильно предоставляя решения для других карт, не сильно уступающие другим алгоритмам.

Список литературы

- [Barer et al., 2014] Barer M., Sharon G.; Stern R., Felner A. Suboptimal Variants of the Conflict-Based Search Algorithm for the Multi-Agent Pathfinding Problem // In: Proceedings of the Annual Symposium on Combinatorial Search (SoCS). – 2014. – Vol. 5(1). – P. 19-27. – doi: 10.1609/socs.v5i1.18315.
- [De Wilde et al., 2014] De Wilde B., Ter Mors A. W., Witteveen C. Push and rotate: a complete multi-agent pathfinding algorithm // In: Journal of Artificial Intelligence Research. – 2014. – Vol. 51. – P. 443-492. – doi: 10.1613/jair.4447.

- [Khorshid et al., 2011] Khorshid M., Holte R., Sturtevant N. A polynomial-time algorithm for non-optimal multi-agent pathfinding // In: Proceedings of the International Symposium on Combinatorial Search. – 2011. – Vol. 2(1). – P. 76-83. – doi: 10.1609/socs.v2i1.18205.
- [Li et al., 2021] Li J., Ruml W., Koenig S. Eecbs: A bounded-suboptimal search for multi-agent pathfinding // In: Proceedings of the AAAI conference on artificial intelligence. – 2021. – Vol. 35(14). – P. 12353-12362. – doi: 10.1609/aaai.v35i14.17466.
- [Sajid et al., 2012] Sajid Q., Luna R., Bekris K. Multi-agent pathfinding with simultaneous execution of single-agent primitives // In: Proceedings of the International Symposium on Combinatorial Search. – 2021. – Vol. 3(1). – P. 88-96. – doi: 10.1609/socs.v3i1.18243.
- [Sharon et al., 2015] Sharon G., Stern R., Felner A., Sturtevant N. R. Conflict-based search for optimal multi-agent pathfinding // In: Artificial intelligence. – 2015. – Vol. 219. – P. 40-66. – doi: 10.1016/j.artint.2014.11.006.
- [Stern et al., 2019] Stern R., Sturtevant N.R., Felner A., Koenig S., Ma H., Walker T.T., Li J., Atzmon D., Cohen L., Kumar T.K.S., Boyarski E., Bartak R. Multi-Agent Pathfinding: Definitions, Variants, and Benchmarks // In: Symposium on Combinatorial Search (SoCS). – 2019. – Vol. 10(1). – P. 151-158. – doi: 10.1609/socs.v10i1.18510.
- [Okumura et al., 2019] Okumura K., Tamura Y., Défago X. winpibt: Extended prioritized algorithm for iterative multi-agent path finding // arXiv preprint arXiv:1905.10149. doi: 10.48550/arXiv.1905.10149. – 2019.
- [Okumura et al., 2022] Okumura K., Machida M., Défago X., Tamura Y. Priority inheritance with backtracking for iterative multi-agent path finding // In: Artificial Intelligence. – 2022. – Vol. 310, 103752. – doi: 10.1016/j.artint.2022.103752.
- [Zhang et al., 2023] Zhang Y., Harabor D., Le Bodic P., Stuckey P. J. Efficient Multi Agent Path Finding with Turn Actions // In: Proceedings of the International Symposium on Combinatorial Search. – 2023. – Vol. 16(1). – P. 119-127. – doi: 10.1609/socs.v16i1.27290.

УДК 004.896

doi: 10.15622/rcai.2025.048

НАВИГАЦИЯ ГРУППЫ ВЗАИМОЗАМЕНЯЕМЫХ АГЕНТОВ В НЕПРЕРЫВНОЙ СРЕДЕ

А.В. Микрюкова (*avmikryukova@edu.hse.ru*)^A

С.А. Дергачев (*sadergachev@hse.ru*)^{A,B}

^A Национальный исследовательский университет

«Высшая школа экономики», Москва

^B Федеральный исследовательский центр

«Информатика и управление» РАН, Москва

В данной работе рассматривается задача планирования путей для группы взаимозаменяемых агентов в непрерывной среде. В отличие от классического случая, в рассматриваемой постановке целевые позиции не закреплены за конкретными агентами. Проведён обзор современных методов много-агентного планирования, в ходе которого выделен алгоритм *GAP*, который использует непрерывное представление пространства и предъявляет минимальные требования к входным данным. Алгоритм был экспериментально сравнен с актуальными дискретными подходами. Результаты показали, что переход к непрерывному представлению существенно повышает качество планирования. Вместе с тем, ограничения существующих алгоритмов могут снижать долю успешно решённых задач при увеличении плотности агентов.

Ключевые слова: много-агентная навигация, много-агентный поиск путей, навигация групп взаимозаменяемых агентов, немаркированная навигация.

Введение

Проблема координации действий групп мобильных агентов является одной из важных исследовательских задач, встречающихся во многих практических областях, среди которых автоматизация логистики и доставки грузов [Wurman et al., 2008], создание мобильных робототехнических платформ для экстренного реагирования, а также разработкой видеоигр и симуляторов транспортных потоков. Ключевое направление исследований заключается в создании алгоритмов, способных эффективно планировать безопасные перемещения десятков или сотен роботов/агентов в динамических средах с учетом ограниченных ресурсов.

Задачу координации движения зачастую формулируют как задачу много-агентного планирования путей (*Multi-Agent Path Finding, MAPF*). Она заключается в поиске совокупности траекторий, обеспечивающих перемещение каждого робота/агента из начальной позиции в целевую без столкновений. Вариативность постановок задачи *MAPF* порождает множество исследовательских направлений. Например, могут рассматриваться различные представления рабочего пространства: перемещения могут ограничиваться дискретной сеткой или происходить в непрерывном пространстве [Stern et al., 2019].

В классической постановке задачи *MAPF* каждому агенту назначена конкретная цель, однако, существуют и альтернативные формулировки. Например, задача немаркированного/анонимного много-агентного поиска путей (*Unlabeled/Anonymous MAPF, AMAPF*), в котором агенты взаимозаменяемы и цели не закреплены за конкретными агентами, главное, чтобы все целевые позиции были достигнуты.

Аналогично классической задаче, при планировании для группы взаимозаменяемых агентов так же может быть использовано как дискретное представление пространства, так и непрерывное. При этом существует лишь небольшое количество работ, посвященных этому направлению, причем все полные методы, работающие в непрерывном пространстве, опираются на различные ограничения, накладываемые на входные данные. Данная работа сфокусирована на задаче с непрерывным представлением среды, что более точно отражает реальные условия применения, например, в роботизированных складах или при координации автономных транспортных средств. Для исследования был выбран алгоритм *GAP* [Turpin et al., 2014], работающий в непрерывном пространстве и накладывающий наименьший набор ограничений на входные данные. В рамках работы впервые приводятся результаты экспериментального сравнения непрерывного метода *GAP* с дискретными алгоритмами *TSWAP* и *FlowNetwork*, что позволяет выявить практические преимущества и ограничения методов, работающих в непрерывном пространстве.

1. Обзор литературы

Нахождение оптимального решения в задаче много-агентного поиска пути (*MAPF*) в её классической формулировке относится к классу NP-трудных проблем, что существенно затрудняет разработку эффективных алгоритмов и их применение в системах с большим количеством агентов [Okumura et al., 2022]. В случае немаркированного планирования путей (*Unlabeled/Anonymous MAPF, AMAPF*) задача (с точки зрения времени достижения всеми агентами целевых позиций) становится разрешимой за полиномиальное время [Yu et al., 2013].

Рассмотрим подробнее методы решения задачи немаркированного много-агентного планирования траектории. Большинство существующих методов предполагает наличие заранее заданного дискретного представления среды в виде сетки или графа специального вида, что не всегда возможно на практике и может приводить к снижению качества решений. Например, в работе [Yu et al., 2013] представлен алгоритм *FlowNetwork*, работающий на безопасных графах единичных расстояний, гарантирующий нахождение *makespan*-оптимальных решений за полиномиальное время, благодаря сведению исходной задачи к проблеме поиска максимального потока в сети. На практике, при использовании графов больших размеров, указанных алгоритм требует существенных вычислительных ресурсов. Альтернативой являются эвристические подходы, такие как алгоритм *TSWAP* [Okumura et al., 2022], который жертвует оптимальностью ради снижения вычислительных затрат. *TSWAP* комбинирует локальные перестановки целей с поэтапным планированием, демонстрируя гораздо большую скорость работы по сравнению с алгоритмом *FlowNetwork*.

С другой стороны, существует ограниченное число подходов, позволяющих находить решение в непрерывном пространстве. Алгоритмы, подходящие для агентов, движение которых происходит в непрерывной среде, представлены в статьях [Turpin et al., 2014], [Solovey et al., 2015], [Kloder et al., 2006], [Banyassady et al., 2022], [Le et al., 2021]. Алгоритм в работе [Banyassady et al., 2022] является полным, он предназначен для среды сложной формы, однако не предусматривает наличие отдельстоящих препятствий, что делает его непригодным для большинства практических сценариев. Подход, представленный в статье [Solovey et al., 2015], так же является полным, авторы доказывают, что сумма длин всех путей их решения не превышает значения $OPT + 4m$, где OPT – стоимость оптимального решения, а m – число агентов. Однако, для корректной работы алгоритма необходимо ввести сильные ограничения: минимальное расстояние между стартами/целями и препятствиями $\sqrt{5}R$, а минимальное расстояние между позициями стартов и между позициями целей равно $4R$, где R – радиус агента. Алгоритм *GAP*, предложенный в [Turpin et al., 2014], гарантирует нахождение решения, не накладывая при этом условий на расстояние от агентов до препятствий, ограничивая только расстояния между стартами и между целями ($4R$). Авторы статьи [Kloder et al., 2006] используют комплексные многочлены для решения задачи, что позволяет им строить путь между целью и агентом максимально приближенным к кратчайшему, однако данный алгоритм не учитывает возможные столкновения между агентами, так как они представляются точками. Стоит отметить, что существует алгоритм, учитывающий различные особенности агентов (их форму, модель движения) [Le et al., 2021], однако он не гарантирует нахождение решений.

Ещё одним активно развивающимся подходом к решению задачи является использование нейронных сетей и обучения с подкреплением [Khan et al., 2021], [Chu et al., 2024]. Однако эти не обеспечивают теоретических гарантий безопасности и полноты, а также не рассматривают препятствия сложной структуры.

На основе проведенного анализа были выбраны следующие критерии сравнения алгоритмов: возможность работы в непрерывном пространстве, ограничения, накладываемые на входные данные, теоретические гарантии полноты и безопасности, оптимальность с точки зрения одной из метрик. Указанная информация для ключевых алгоритмов представлена в табл. 1. В дополнение к выбранным критериям для всех методов также был указан год публикации работы и доступность открытой реализации.

Таблица 1

Сравнительный анализ ключевых методов решения задачи *AMAPF*

Метод	Непрерывная среда	Полнота и безопасность	Ограничения (старты/цели)	Ограничения (позиции-препятствия)	Оптимальность	Реализация
<i>FlowNetwork</i> [Yu et al., 2013]	—	+	Н/П	Н/П	+	+
<i>TSWAP</i> [Okumura et al., 2022]	—	+	Н/П	Н/П	—	+
Метод из [Banyassady et al., 2022]	+	+	—	—	—	—
Метод из [Kloder et al., 2006]	+	—	+	+	—	—
Метод из [Solovey et al., 2015]	+	+	—	—	—	—
<i>GAP</i> [Turpin et al., 2014]	+	+	—	+	—	—
Метод из [Le et al., 2021]	+	—	+	+	—	+
<i>GPG</i> [Khan et al., 2021]	+	—	+	—	—	—
<i>PROFILE</i> [Chu et al., 2024]	+	—	+	—	—	—

Как можно видеть из результатов анализа, ни один из алгоритмов, работающих в непрерывном пространстве, не гарантирует оптимальность найденного решения. При этом алгоритмы [Turpin et al., 2014], [Solovey et al., 2015], [Banyassady et al., 2022] гарантируют нахождение безопасного решения при выполнении ряда условий. Среди них можно выделить алгоритм *GAP* [Turpin et al., 2014], накладывающий ограничения только на расстояния между стартами/целями. Отдельно отметим, что ни для одного алгоритма планирования в непрерывной среде нет открытой реализации для проведения экспериментов.

Таким образом, по результатам анализа был выбран алгоритм [Turpin et al., 2014], так как он гарантирует нахождение решений и накладывает наименьшее число ограничений на среду.

2. Формальная постановка задачи

Обозначим агентами натуральными числами от 1 до N , а множество агентов как $I_N = \{1, 2, \dots, N\}$. Дано рабочее пространство $X \subseteq \mathbb{R}^2$ – непрерывная среда с препятствиями, заданными как замкнутые множества. Обозначим $X_{obs} \subset X$ пространство препятствий, тогда $X_{free} = X \setminus X_{obs}$ – область, доступная для движения агентов. Каждый из агентов моделируется диском радиуса $R \in \mathbb{R}^+$ и может двигаться с постоянной скоростью v . За $x_i(t) \in X_{free}$ обозначим состояние агента в момент t , а за $B(x_i) \subset \mathbb{R}^2$ – множество всех точек, занимаемых агентом.

Для каждого агента i задана начальная позиция $s_i = x_i(0) \in X_{free}$. Также заданы позиции целей: $g_j \in X_{free}$, $\forall j \in I_N$. Позиции стартов и целей задаются так, чтобы не возникало пересечений агентов с препятствиями и между собой:

$$\forall i \neq j \in I_N: \|s_i - s_j\|_2 \geq 2R, \quad \|g_i - g_j\|_2 \geq 2R,$$

$$\forall i \in I_N: B(s_i) \subset X_{free}, B(g_i) \subset X_{free}$$

За множество всех точек, занимаемых агентом, который движется по траектории/пути $\gamma : t \rightarrow X_{free}$ обозначим $B(\gamma) \subset \mathbb{R}^2$, при этом для корректности траектории также должно выполняться $B(\gamma) \subset X_{free}$.

Таким образом, задача немаркированного много-агентного планирования путей заключается в нахождении:

1. Функции назначения целей всем агентам $\phi : I_N \rightarrow I_N \cup 0$, где

$$\phi_i = \begin{cases} j, & \text{если агент } i \text{ должен достигнуть цель } g_j \\ 0, & \text{иначе} \end{cases}$$

2. Набора траекторий $\{\gamma_i(t)\}_{i=1}^N$ для каждого агента от его стартовой позиции до соответствующей цели т.ч. не возникает столкновений с препятствиями и между агентами:

$$\gamma_i(0) = s_i, \quad \gamma_i(T_i) = g_{\phi_i},$$

$$\forall t \in [0, \max_i T_i], \forall i \neq k : \|\gamma_i(t) - \gamma_k(t)\|_2 \geq 2R,$$

$$\forall i \in I_N, \forall t \geq 0 : B(\gamma_i(t)) \subset X_{free},$$

где $T_i = \inf\{t \geq 0 | x_i(t) = g_{\phi_i}\}$ – время, к которому агент i достигает цель.

Для оценки качества решения введём метрики **makespan** – максимальное время достижения целевой позиции среди всех агентов и **flowtime** – сумма времен достижения целевых позиций всеми агентами.

$$\text{makespan} = \max_{i \in I_N} T_i, \quad \text{flowtime} = \sum_{i \in I_N} T_i.$$

3. Алгоритм немаркированного много-агентного планирования пути в непрерывном пространстве

3.1. Общее описание алгоритма

В работе рассматривается алгоритм *GAP* [Turpin et al., 2014], который решает задачу планирования траекторий для групп взаимозаменяемых агентов в непрерывных средах с препятствиями. Основная идея заключается в декомпозиции проблемы на три ключевых этапа:

1. Поиск кратчайших путей: для каждого агента к каждой цели независимо от других агентов находится оптимальный маршрут.
2. Оптимальное назначение целей: решается задача лексикографического назначения, минимизирующая максимальную стоимость траекторий.
3. Приоритизация и параметризация:
 - Агенты упорядочиваются по приоритету на основе пересечений их траекторий.
 - Вводятся временные смещения \hat{t}_i для гарантии отсутствия столкновений между агентами.

Далее будут более подробно рассмотрены отдельные компоненты алгоритма.

3.2. Планирование траекторий и построение графа

Для работы алгоритма нам необходимо построить путь для каждого агента от его начальной позиции до всех целевых. Одним из наиболее распространенных и эффективных методов заключается в решении этой задачи, как задачи поиска пути на графе.

Итак, вводится граф общего вида $G = (V, E)$. Граф может формироваться путём дискретизации пространства X_{free} с использованием детерминированных или вероятностных методов [Kavraki L.E. et al., 1996]. Каждая вершина $v_i \in V$ представляет допустимое, не допускающее столкновений состояние в X_{free} . Кроме того, множество вершин V должно содержать все начальные и целевые состояния:

$$s_i \in V \forall i \in I_N, g_j \in V \forall j \in I_N.$$

Ребра $e = (v, u) \in E$ соответствуют безопасным траекториям $\gamma_{v,u}(t)$ ($B(\gamma_{v,u}) \subset X_{free}$) между соответствующими вершинами: $\gamma_{v,u}(t_0) = v$, $\gamma_{v,u}(t_f) = u$. Каждому ребру ставится в соответствие стоимость перехода $w(\gamma_{v,u}) > 0$.

Для гарантии полноты должны выполняться следующие свойства:

$$\begin{aligned} v_k \neq s_i \wedge v_l \neq s_i &\Rightarrow B(\tau_{kl}) \cap B(s_i) = \emptyset, \\ v_k \neq g_j \wedge v_l \neq g_j &\Rightarrow B(\tau_{kl}) \cap B(g_j) = \emptyset. \end{aligned}$$

Также для всех стартов и целей должны выполняться следующие условия:

$$\forall i \neq j \in I_N: \|s_i - s_j\|_2 \geq 4R, \quad \|g_i - g_j\|_2 \geq 4R.$$

После построения графа G для каждого $i, j \in I_N$ вычисляется оптимальная траектория $\gamma_{ij}(t)$ между позициями s_i и g_j , минимизирующая суммарную стоимость ребер. Этот этап реализуется с помощью алгоритма Дейкстры [Dijkstra, 1959]. В результате формируется матрица $C \in \mathbb{R}^{N \times N}$, где элемент C_{ij} соответствует стоимости пути $P^*(s_i, g_j)$ в графе:

$$C_{ij} = \sum_{e=(v,u) \in P^*(s_i, g_j)} w(\gamma_{v,u}).$$

Если цель g_j недостижима из s_i , полагаем $C_{ij} = \infty$.

Стоит отметить, что в отличие от дискретных методов, в данном алгоритме граф может иметь свободную форму и строиться на основе различных подходов, например, на основе графа видимости или методом PRM [Kavraki et al., 1996].

3.3. Назначение целей

Назначение целей $\phi^*: I_N \rightarrow I_N \cup 0$ определяется путем решения задачи *LexBAP* (*Lexicographic Bottleneck Assignment Problem*). Решение гарантирует, что:

$$\max_i C_{i\phi^*(i)} \leq \max_i C_{i\phi(i)}, \quad \forall \phi \in \Phi,$$

где Φ – множество допустимых биекций.

Для снижения вычислительной сложности *LexBAP* аппроксимируется задачей:

$$\phi^* = \arg \min_{\phi} \left(\sum_{i=1}^N C_{i\phi(i)}^p \right)^{1/p},$$

где $p \gg 1$ (например, в текущей реализации используется $p = 50$). При $p \rightarrow \infty$ решение совпадает с *LexBAP*.

Модифицированная задача решается за $O(N^3)$ с помощью Венгерского алгоритма и дополнительной проверкой условия:

$$\max (C_{i\phi^*(i)}, C_{j\phi^*(j)}) \leq \max (C_{j\phi^*(j)}, C_{i\phi^*(i)}), \quad \forall i, j.$$

3.4. Приоритизация и предотвращение столкновений

Для обеспечения предотвращения столкновений вводятся следующие правила частичного упорядочивания между агентами, основанный на их траекториях и взаимном расположении. Условия приоритизации задаются следующим образом:

1. Правило стартовых пересечений – если стартовая позиция агента i лежит на оптимальном пути агента j , последний получает более высокий приоритет

$$s_i \in P^*(s_j, g_{\phi^*(j)}) \Rightarrow j > i.$$

2. Правило целевых пересечений – приоритет агента j больше, чем приоритет агента i , если его траектория проходит через цель агента i

$$g_{\phi^*(i)} \in P^*(s_j, g_{\phi^*(j)}) \Rightarrow j > i.$$

На основе установленных правил вычисляется полный порядок ψ и выполняется поиск смещений времени начала движения \hat{t} для каждого агента. Далее представлен алгоритм определения временных смещений:

1. Агенту с наивысшим приоритетом (ψ_1) назначается $\hat{t}_{\psi_1} = 0$.
2. Для каждого последующего агента ψ_k вычисляется минимальное смещение:

$$\hat{t}_{\psi_k} = \min \{t \geq 0 : \forall m < k, B(\gamma_{\psi_k}(t - \hat{t}_{\psi_k})) \cap B(\gamma_{\psi_m}(t - \hat{t}_{\psi_m})) = \emptyset\}.$$

В результате получается набор траекторий $\{\gamma_i(t - \hat{t}_i)\}$, гарантирующий:

$$\forall t, i \neq j \in I_N: \|\gamma_i(t) - \gamma_j(t)\|_2 \geq 2R + \epsilon.$$

где ϵ – параметр алгоритма, задающий дополнительный безопасный зазор.

4. Экспериментальное исследование

Для экспериментов были использованы три карты из широко используемого в *MAPF* сообществе набора бенчмарков *MovingAI* [Stern et al., 2019]: *random-32-32-20* (32 x 32), *maze-32-32-2* (32 x 32), *den312d* (256 x 257). Иллюстрация карт представлена на рис. 1.

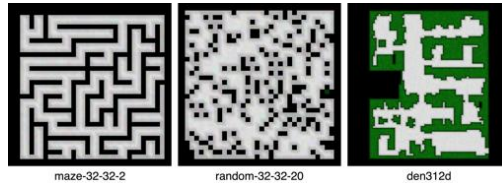


Рис. 1. Карты, используемы в экспериментах

Для каждой карты было случайно сгенерировано по 50 сценариев, содержащих по 100 пар стартовых и целевых позиций, удовлетворяющих ограничениям алгоритма *GAP*. Число агентов варьировалось от 10 до 100 с шагом 10; для запуска с k агентами выбирались первые k пар из сценария.

Для проведения экспериментальной оценки были реализованы несколько вариантов алгоритма GAP^1 . Так, были реализованы варианты алгоритма, использующие начальное назначение, минимизирующее сумму длин путей ($p = 1$) и назначение, полученное на основе аппроксимации решения задачи $LexBAP$ ($p = 50$). Кроме того, были реализованы два варианта построения графа. В первом варианте использовалось более близкое к дискретному представление в виде 8-связной сетки. Во втором варианте использовался стохастический алгоритм PRM [Kavraki et al., 1996] с числом вершин равным 2000 и радиусом соединения 5.

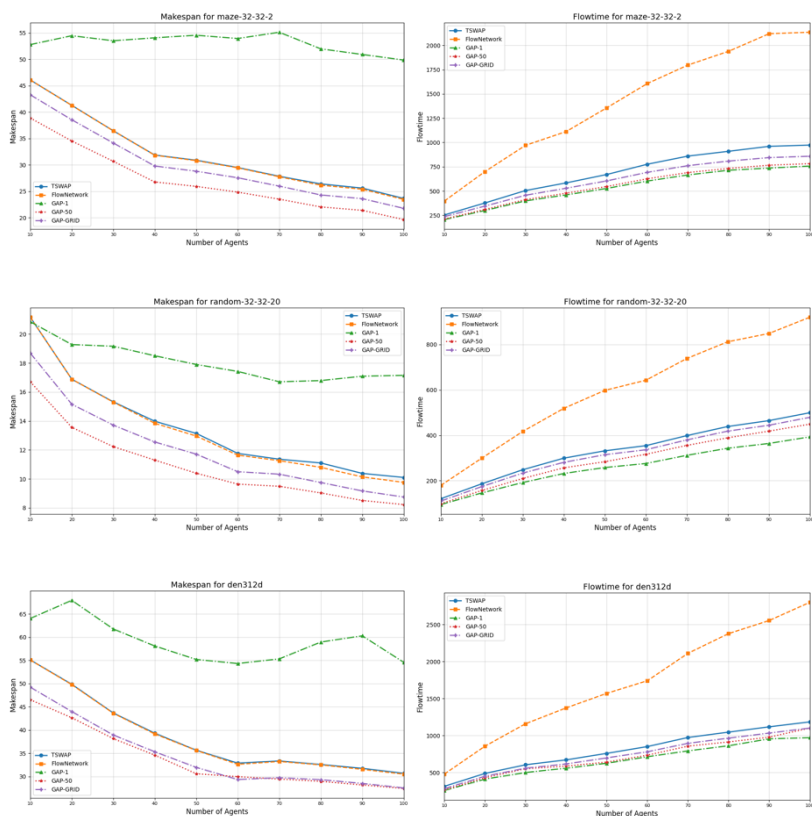


Рис. 2. Средние показатели метрик *flowtime* и *makespan* для различного числа агентов. Меньше – лучше

¹ Исходный код: <https://github.com/a-v-mikryukova/course-project-2>.

В экспериментальном исследовании участвовали следующие варианты алгоритма: два варианта использующих для формирования графа метод *PRM*, один из которых в задаче назначения использует $p = 1$ (обозначен как ***GAP-1***), другой использует $p = 50$ (обозначен как ***GAP-50***), третий вариант, обозначенный как ***GAP-GRID*** использует 8-связную сетку и $p = 50$. Кроме того, в рамках экспериментов для проведения сравнения были проведены запуски дискретных алгоритмов *TSWAP* [Okumura et al., 2022] и *FlowNetwork* [Yu et al., 2013].

Алгоритмы оценивались по средним значениям метрик *makespan* и *flowtime*. Результаты, представленные на рис. 2, оказались схожими для всех карт. Алгоритм *GAP-1* показал лучшие значения по *flowtime*, однако уступает другим методам по *makespan*. При этом, результаты *GAP-50* превосходят показатели дискретных методов (включая оптимальный по *makespan* алгоритм *FlowNetwork*) по обоим метрикам, приближаясь по *flowtime* к *GAP-1*. *GAP-GRID* занял промежуточное положение между дискретным алгоритмом *TSWAP* и непрерывным *GAP-50*, а на карте *den312d* его результаты почти совпадают с *GAP-50*.

Таким образом, алгоритмы, работающие в непрерывной среде, могут обеспечивать более высокое качество планирования даже при использовании представления близкого к полностью дискретному.

Также было проведено исследование о влиянии ограничений, накладываемых алгоритмом *GAP*, на процент успешно решенных заданий на карте *maze-32-32-2*. Стартовые и целевые позиции выбирались аналогично предыдущим экспериментам, однако при построении сценариев не учитывали ограничения алгоритма *GAP*. Результаты представлены в табл. 2. Так, можно видеть, что при числе агентов свыше 70 доля успешных решений упала ниже 80%, а при 100 агентах – до 65%. Такие результаты в большой степени связаны со структурой карты, где небольшое число свободных клеток образуют узкие проходы, что не позволяет выполнить ограничения алгоритма *GAP*. При этом отметим, что указанные задания могут быть решены рассмотренными дискретными алгоритмами в 100% случаев.

Таблица 2

Доля успешно решенных заданий алгоритмом *GAP-50* при полностью случайном расположении агентов

Число агентов	10	20	30	40	50	60	70	80	90	100
Процент успеха	99%	96%	92%	89%	85%	83%	79%	74%	70%	65%

Заключение

Таким образом, в работе была рассмотрена задача немаркированного много-агентного планирования путей в непрерывной среде. Проведённый анализ существующих подходов позволил выделить алгоритм *GAP* как наиболее универсальный благодаря минимальным ограничениям на входные данные. Экспериментальное сравнение с дискретными алгоритмами показало значительное преимущество *GAP* по качеству получаемых решений. Однако было установлено, что ограничения, накладываемые на входные данные, могут существенно снижать долю успешно решённых задач при увеличении плотности агентов. Таким образом, важным направлением будущих исследований является модификация алгоритма *GAP* для устранения или смягчения ограничений на входные данные при сохранении теоретических гарантий. Другим важным направлением работ является сравнение непрерывных алгоритмов с дискретными аналогами с точки зрения времени решения.

Список литературы

- [Banyassady et al., 2022] Banyassady B. et al. Unlabeled multi-robot motion planning with tighter separation bounds // 38th international symposium on computational geometry (SoCG 2022). – 2022.
- [Chu et al., 2024] Chu W. et al. Perception field based imitation learning for unlabeled multi-agent pathfinding // Science China Information Sciences. – Springer, 2024. – Vol. 67(5). – P. 152107.
- [Dijkstra, 1959] Dijkstra E.W. A note on two problems in connexion with graphs // Numerische mathematik. – 1959. – Vol. 1(1). – P. 269-271.
- [Kavraki et al., 1996] Kavraki L.E. et al. Probabilistic roadmaps for path planning in high-dimensional configuration spaces // IEEE transactions on Robotics and Automation. – IEEE, 1996. – Vol. 12(4). – P. 566-580.
- [Khan et al., 2021] Khan A., Kumar V., Ribeiro A. Large scale distributed collaborative unlabeled motion planning with graph policy gradients // IEEE Robotics and Automation Letters. – 2021. – Vol. 6(3). – P. 5340-5347.
- [Kloder et al., 2006] Kloder S., Hutchinson S. Path planning for permutation-invariant multirobot formations // IEEE Transactions on Robotics. – IEEE, 2006. – Vol. 22(4). – P. 650-665.
- [Le et al., 2021] Le D., Plaku E. Multi-robot motion planning with unlabeled goals for mobile robots with differential constraints // 2021 IEEE international conference on robotics and automation. – IEEE, 2021.
- [Okumura et al., 2022] Okumura K., Défago X. Solving simultaneous target assignment and path planning efficiently with time-independent execution // Proceedings of the international conference on automated planning and scheduling (ICAPS 2022). – 2022. – Vol. 32. – P. 270-278.
- [Solovey et al., 2015] Solovey K. et al. Motion planning for unlabeled discs with optimality guarantees // Proceedings of robotics: Science and systems (RSS 2015). – 2015.

- [**Stern et al., 2019**] Stern R. et al. Multi-agent pathfinding: Definitions, variants, and benchmarks // Proceedings of the 12th annual symposium on combinatorial search (SoCS 2019). – 2019. – P. 151-158.
- [**Turpin et al., 2014**] Turpin M. et al. Goal assignment and trajectory planning for large teams of interchangeable robots // Autonomous Robots. – 2014. – Vol. 37(4). – P. 401-415.
- [**Wurman et al., 2008**] Wurman P.R., D’Andrea R., Mountz M. Coordinating hundreds of cooperative, autonomous vehicles in warehouses // AI magazine. – 2008.
- [**Yu et al., 2013**] Yu J., LaValle S.M. Multi-agent path planning and network flow // Proceedings of the workshop on the algorithmic foundations of robotics (WAFR 2013). – 2013. – P. 157-173.

УДК 004.89

doi: 10.15622/rcai.2025.049

АНСАМБЛЬ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ ИЗОБРАЖЕНИЙ В РАМКАХ СИСТЕМ ПОДДЕРЖКИ ПРИНЯТИЯ ВРАЧЕБНЫХ РЕШЕНИЙ

И.И. Мищенко (*sombra74@yandex.by*)

А.Е. Мисник (*anton@misnik.by*)

Белорусско-Российский университет,
Республика Беларусь, Могилев

В статье рассматривается модуль предварительной обработки рентгеновских изображений, разработанный для использования в составе систем поддержки принятия врачебных решений, ориентированных на диагностику Hip-Spine синдрома. Основу модуля составляет ансамбль сверточных нейросетей, реализованных по иерархической архитектуре: управляющая супервайзер-сеть анализирует характеристики входного изображения и активирует специализированные подмодули, отвечающие за коррекцию яркости, контрастности и восстановление локальных признаков. Такой подход обеспечивает адаптивность обработки в зависимости от качества визуальных данных. Проведённые эксперименты показали прирост точности классификации на 13% по сравнению с базовыми методами обработки. Представлены подробная архитектура модуля, принципы его функционирования, результаты оценки эффективности и перспективы дальнейшего развития в направлении интеграции трёхмерной реконструкции.

Ключевые слова: система поддержки принятия врачебных решений, Hip-Spine синдром, предварительная обработка, нейронные сети, рентгеновские снимки.

Введение

Системы поддержки принятия врачебных решений (СППВР) становятся неотъемлемой частью современного здравоохранения, особенно в условиях растущего объема медицинских данных и необходимости быстрого реагирования на клинические ситуации. Они позволяют автоматизиро-

вать основные этапы анализа информации, сократить влияние человеческого фактора и обеспечить более объективную и обоснованную постановку диагноза.

СППВР применяются в различных отраслях медицины, включая кардиологию, онкологию, офтальмологию, а также в области травматологии и ортопедии. В последнем случае особую роль играют методы медицинской визуализации, в частности анализ рентгеновских снимков, которые являются основным источником диагностической информации при патологиях опорно-двигательного аппарата [Гонсалес, 2015].

Для эффективной работы СППВР необходимо обеспечить высокое качество исходных данных, что возможно только при применении многоступенчатой системы предварительной обработки изображений. Без качественной фильтрации, коррекции контрастности и адаптивного масштабирования даже самые передовые алгоритмы машинного зрения не смогут достичь необходимой точности [Прэтт, 2022]. Особое значение это приобретает при диагностике сложных состояний, таких как Hip-Spine синдром, где ошибка в интерпретации визуальных данных может привести к неправильному выбору стратегии лечения.

Современные системы обработки медицинских изображений широко применяются в задачах диагностики. В международных бенчмарках, таких как MURA и SpineWeb лучшие результаты показывают гибридные архитектуры, а именно U-Net для сегментации и ResNet для классификации [Ahmed et al., 2025]. Однако применительно к Hip-Spine синдрому эти подходы требуют модификации из-за необходимости совместного анализа тазобедренного сустава и поясничного отдела позвоночника, а также компенсации специфических артефактов (наложение органов, металлические импланты и т.п.).

Анализ существующих решений показывает, что задачи обработки медицинских изображений для диагностики разрабатываются в нескольких направлениях. В работе автоматическая сегментация и детекция позвонков и межпозвоночных дисков при помощи нейронных сетей авторов Guinebert S., Petit E., Bousson V., Bodard S. предложен метод семантической сегментации позвонков на основе модифицированного U-Net, демонстрирующий точность 84% на датасете SpineWeb. Отечественные исследования предлагают эффективную модификацию U-Net архитектуры для сегментации костных структур, демонстрирующую точность 87% на рентгенограммах позвоночника [Попов и др., 2022]. Однако, существующие подходы фокусируются на изолированном анализе отдельных анатомических областей, тогда как диагностика Hip-Spine синдрома требует комплексной обработки изображений тазобедренного сустава и позвоночника, что и является целью данного исследования.

Предлагаемая архитектура ансамбля нейронных сетей развивает идеи ранее предложенных решений для СППВР за счет введения сети супервайзера с механизмом раннего завершения, что на 15% сокращает время обработки без потери точности [Мищенко и др., 2024а].

Архитектура СППВР

Архитектурно СППВР представляет собой совокупность взаимосвязанных модулей, каждый из которых выполняет строго определенные функции и взаимодействует с другими компонентами. Архитектура включает в себя четыре последовательно взаимодействующих компонента: модуль загрузки и предварительной обработки изображений, модуль сегментации и анализа позвоночника с применением нейросетевых алгоритмов, блок автоматической постановки диагноза и модуль визуализации и сохранения данных. На рис. 1 показана блок-схема основных модулей системы поддержки принятия врачебных решений.

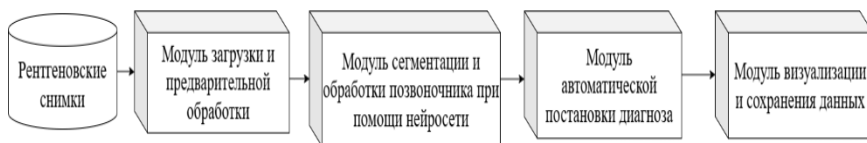


Рис. 1. Блок-схема основных модулей системы поддержки принятия врачебных решений

Все модули работают в единой среде и обеспечивают адаптивную, модульную и масштабируемую структуру СППВР, которая легко адаптируется под конкретные клинические задачи.

Проблематика диагностики Hip-Spine синдрома

Hip-Spine синдром представляет собой совокупность клинических состояний, при которых поражения в тазобедренном суставе и поясничном отделе позвоночника взаимно усиливают болевой синдром и затрудняют постановку диагноза. Эти состояния часто маскируют друг друга, что требует особой точности и внимательности при интерпретации визуальных данных. Классические методы диагностики, основанные на субъективной оценке рентгеновских изображений, недостаточно надежны в условиях перекрывающихся симптомов. Рентгеновские снимки пациентов с подозрением на Hip-Spine синдром часто содержат шумы, артефакты, искажения геометрии и перепады яркости. Это делает необходимым применение сложной и многоуровневой системы предварительной обработки изображений, которая бы позволила устранить эти проблемы до начала анализа.

Дифференциальная диагностика Hip-Spine синдрома требует решения трех взаимосвязанных задач. Во-первых, необходимо точно идентифицировать источник патологии с определением первичного очага поражения (позвоночник L1-L5 или тазобедренный сустав) и степени компенсаторных изменений, включая углы смещения, с точностью не менее 90%. Во-вторых, важным является структурированный анализ изображений, предполагающий совместную обработку парных снимков в фронтальной и сагиттальной проекциях с вычислением интегральных признаков: гистограммных характеристик (энтропия, контраст) и геометрических параметров (углы, расстояния между структурами). В-третьих, требуется оптимизированный алгоритм обработки, обеспечивающий автоматическую адаптацию к качеству снимка на основе QoI-метрики, минимизацию артефактов (металлические импланты, наложения органов) и минимальное среднее время анализа на изображение. Предлагаемая архитектура СППВР реализует эти требования через ансамбль из пяти специализированных нейросетей, иерархическую систему принятия решений и динамическую маршрутизацию задач обработки [Мищенко и др., 2024b].

Архитектура и реализация модуля предварительной обработки

Для решения проблемы классификации Hip-Spine синдрома главным компонентом системы поддержки принятия врачебных решений стал модуль предварительной обработки рентгеновских изображений. Разработка данного модуля для СППВР преследует две основные цели: сокращение диагностических ошибок и повышение воспроизводимости результатов. В отличие от классических фильтров, встроенных в рентгеновские аппараты, программный модуль предоставляет многоуровневую обработку, которая адаптируется к характеристикам конкретного снимка [Kheirandish, 2023]. Основание для такой адаптации – статистическое разнообразие клинических данных: различия в моделях оборудования, в анатомических особенностях пациентов и даже в цифровых форматах хранения. Без учёта этих факторов любая универсальная схема обработки либо переусложнена, либо даёт усреднённый, но не оптимальный результат [Yang et al., 2021]. Предлагаемая архитектура строится вокруг принципа разделения обязанностей: каждая подсистема отвечает за устранение строго определённого класса артефактов, а центральная управляющая сеть синхронизирует их работу. Архитектура модуля разработана на основе ансамбля сверточных нейронных сетей (CNN), которые взаимодействуют по принципу master-slave.

Основными дополнениями и изменениями архитектуры, примененной в предыдущих исследованиях, являются:

1. Иерархическое управление. Так вместо линейного конвейера применен динамический выбор обработчиков. Супервайзер-сеть на базе ResNet-50 анализирует DICOM-метаданные (kVp, mAs) и генерирует бинарную маску активации подмодулей.

2. Оптимизация ресурсов. Для снимков с $QoI > 0.92$ обработка прекращается после этапа шумоподавления (экономия 20-40 мс на изображение).

В основе модуля лежит иерархическая структура, сочетающая ансамбли моделей и принципы сервис-ориентированной архитектуры (SOA). Каждый подмодуль развёрнут как независимый микросервис, что облегчает масштабирование и обновление. На верхнем уровне расположена супервайзер-сеть, приобретающая статус «скелета» архитектуры. Её задача – сформировать вектор метаданных, отражающих текущее состояние входного изображения: медианный уровень яркости, спектральную плотность шума, статистику контраста, показатели анизотропии и кривизны краёв. На основании этого вектора сеть генерирует маску активации, определяя, какие подмодули будут запущены, в каком порядке и с какими параметрами. Такая динамическая маршрутизация экономит вычислительные ресурсы: для качественных снимков активируется только лёгкая нормализация, для сложных – цепочка из пяти-семи преобразований.

После получения цифрового кадра DICOM парсер извлекает матрицу пикселей и базовый набор метаданных: kVp, mAs, расстояние источник-приёмник, позицию пациента по DICOM-тегу. Эти сведения передаются супервайзеру, который добавляет к ним внутрисетевые оценки качества. Далее формируется запрос на распределение задач. Каждому подмодулю предшествует буфер состояния: если кадр уже был обработан, сервис мгновенно возвращает результат [Nguyen, 2022].

В зависимости вектора метаданных активируются одна или несколько подчиненных нейросетей:

- 1 Сеть усиления контраста сочетает классические и нейросетевые методы для адаптивной обработки рентгеновских снимков. На первом этапе применяется алгоритм CLAHE (Contrast Limited Adaptive Histogram Equalization) с параметрами $clip_limit=2.0$ и $tile_grid_size=(8 \times 8)$, обеспечивающий глобальную коррекцию контраста. Затем специализированная сверточная сеть (3 слоя с ядрами 5×5 и ReLU-активацией) анализирует локальные особенности изображения: для костных структур (пиксели с интенсивностью < 150 HU) применяется оператор Лапласа с коэффициентом усиления $\alpha=0.3$, повышающий резкость, а для металлических имплантов (интенсивность > 220 HU) используется пороговая фильтрация, снижающая засветку.

- 2 Сеть нормализации яркости или же сеть выравнивания освещения реализована на основе модифицированной архитектуры U-Net с четырьмя уровнями дискретизации. Сеть принимает на вход изображение размером 512×512 пикселей и генерирует карту яркости $L \in [0, 1]^{H \times W}$, где значения близкие к 0 соответствуют затемненным областям, а к 1 – переэкспонированным. Особенностью архитектуры являются пропущенные соединения

(skip-connections) между энкодером и декодером, сохраняющие мелкие детали, и функция потерь, обеспечивающая сохранение структурной информации, которая находится по формуле 1:

$$L = \alpha \cdot MSE(I_{out}, I_{ref}) + \beta \cdot SSIM(I_{out}, I_{ref}), \quad (1)$$

где I_{ref} – эталонные снимки, $\alpha = 0.7$, $\beta = 0.3$ – весовые коэффициенты.

3 Сеть шумоподавления реализует гибридный подход к устранению артефактов: на первом этапе применяется адаптивный фильтр Винера с окном 5×5 пикселей для подавления высокочастотного шума, сохраняющий структуру костных тканей. Затем трехслойный сверточный автоэнкодер (ядра 3×3 , stride=1) с симметричной архитектурой восстанавливает потерянные детали, обучаясь на синтетически зашумленных данных.

4 Сеть геометрической коррекции построена с двумя ключевыми блоками: локализационная сеть (3 полносвязных слоя по 256 нейронов) предсказывает параметры аффинного преобразования (угол поворота $\theta \in [-15^\circ, +15^\circ]$, смещения $\Delta x, \Delta y \in [-10\%, +10\%]$), а слой с билинейной интерполяцией применяет коррекцию.

5 Сеть масштабирования обеспечивает унификацию разрешения с последующей оптимизацией резкости. Модифицированная FSRCNN-архитектура (4 сверточных слоя с активацией PReLU) обучалась на парных изображениях с разным разрешением (исходное/ $0.5x/2x$), минимизируя комбинированную функцию потерь. Система поддерживает приведение к стандартному разрешению 512×512 px.

Все нейросети работают согласованно, автоматически подбирая оптимальные параметры обработки для каждого конкретного снимка.

Управляющая сеть построена на модифицированной ResNet-50, где стандартные 3×3 свертки заменены на depthwise-операторы, что сократило количество параметров на 30% без потери точности. Ключевым компонентом является двухуровневый маршрутизатор дефектов: первый уровень (LSTM на 256 нейронов) анализирует DICOM-метаданные (kVp, mAs), второй (полносвязная сеть) обрабатывает статистики изображения (энтропию, градиенты, гистограмму). На выходе формируется вектор активации $a \in \{0, 1\}^5$, определяющий, какие подмодули необходимо задействовать. При оценке качества изображения $QoI > 0.92$ активируется механизм раннего завершения, пропускающий избыточные этапы обработки и сокращающий время вычислений на 15-20%. [Ahmed et al., 2025].

Финальное изображение формируется путем адаптивного взвешивания выходов активированных подмодулей. Для каждого модуля k супервайзер-сеть вычисляет вес w_k по формуле 2:

$$w_k = \frac{\exp(E_k / T)}{\sum_{i=1}^5 \exp(E_i / T)}, \quad (2)$$

где E_k – оценка полезности преобразования, $T=0.5$ – параметр температуры. Результирующее изображение получается по формуле 3:

$$I_{final} = \sum_{k=1}^5 a_k \cdot w_k \cdot f_k(I_{input}), \quad (3)$$

где a_k – бинарный флаг активации из вектора a .

На заключительном этапе применяется кусочно-линейное выравнивание гистограммы и гауссова фильтрация ($\sigma=0.8$) для устранения артефактов слияния. Эксперименты показали, что такой подход обеспечивает плавные переходы между зонами влияния разных модулей при сохранении диагностически значимых деталей [Panse et. al., 2021].

На рис. 2 представлена схема взаимодействия нейронных сетей.



Рис. 2. Схема взаимодействия нейросетей для модуля предварительной обработки

Данная схема взаимодействия обеспечивает модульность, масштабируемость и высокую точность, позволяя адаптировать обработку под особенности каждого конкретного случая.

Для обучения сформирована база данных, состоящая из 6317 снимков, включающая материал: ФЦТОЭ г. Смоленска, БСМП г. Могилева, европейского открытого датасета MURA и собственных анонимизированных снимков. Каждый кадр прошёл процедуру первичной обработки: удалены DICOM-теги, несущие персональные данные. Итоговый датасет разделён по пациентам в отношении 8:1:1 на обучение, валидацию и тест.

Каждый компонент системы обучался отдельно: модуль коррекции яркости – на восстановлении равномерного освещения, модуль контраста – на оптимизации распределения тонов, модуль шумоподавления – на ми-

нимизации искажений при сохранении детализации. После первоначального обучения параметры компонентов фиксировались, а управляющий модуль дообучался методом пошаговой оптимизации с подкреплением. В качестве критерия эффективности использовался составной показатель качества обработки, учитывающий контрастность, уровень шумов, резкость и равномерность изображения. Применялся алгоритм постепенной оптимизации с ограниченными шагами корректировки. В результате система научилась адаптивно выбирать необходимые этапы обработки.

Оценка эффективности и прирост точности

Для количественной оценки эффективности предложенного подхода проведен ряд экспериментов, включающих различные анатомические области (поясница, таз, бедро). Использовалась ранее сформированная база данных.

Проведено сравнение трёх конфигураций: без обработки, с обработкой при помощи локального метода выравнивания гистограммы яркости, используемый для улучшения контрастности изображений – CLAHE и различных фильтров, а также с предлагаемым ансамблем нейронных сетей.

Метрики вычислялись следующим образом:

- Точность распознавания: доля, верно, идентифицированных анатомических ориентиров относительно экспертной разметки.
- Полнота локализации: процент обнаруженных патологических зон (переломы, остеопиты).
- Комплексная оценка: сумма, состоящая из 40% точности распознавания и 60% полноты локализации.

База данных составляла 6317 снимков (ФЦТОЭ г. Смоленска, БСМП г. Могилева, европейского открытого датасета MURA). Для объективной оценки использовалась 10-кратная кросс-валидация: датасет случайно разделялся на 10 подмножеств, где каждое подмножество по очереди выступало в роли тестовой выборки, одно в роли валидационной, а остальные – обучающей. Итоговые метрики усреднялись по всем подмножествам.

В табл. 1 представлены результаты исследований.

Таблица 1

Метод обработки	Точность распознавания ориентиров, %	Полнота локализации ключевых зон, %	Комплексная оценка качества, %
Без обработки	68	59	63
CLAHE и фильтры	81	77	79
Ансамбль нейросетей	93	90	92

Таким образом, интеграция модуля ансамбля нейросетей позволила увеличить точность распознавания на 12%, а общую производительность – более чем на 13% по сравнению с предыдущими реализациями. Особенно заметно прирост проявился в сложных случаях с плохим качеством изображений и множеством артефактов.

Заключение

Разработка модуля предварительной обработки рентгеновских изображений на основе ансамбля нейронных сетей позволяет совершенствовать интеллектуальные системы поддержки врачебных решений, ориентированных на диагностику заболеваний опорно-двигательного аппарата. Предложенная архитектура сочетает в себе гибкость и масштабируемость, позволяя адаптироваться к различным входным условиям, включая изображения с низким качеством, шумами, неравномерной экспозицией и искажёнными геометрическими пропорциями.

В рамках модуля используется принцип иерархического управления, где управляющая супервайзер-сеть анализирует характеристики входного изображения и определяет оптимальный путь его обработки. Такое решение демонстрирует высокую точность и устойчивость по сравнению с традиционными методами, поскольку задействует специализированные подмодули, нацеленные на устранение конкретных дефектов: пониженную яркость, недостаточную контрастность, артефакты и искажения локальных признаков.

Результаты экспериментальной оценки, проведенной на созданной базе данных, подтверждают эффективность предложенного подхода. Повышение комплексной оценки качества на 13% и рост общей точности до уровня 92% свидетельствуют о высоком потенциале предлагаемой технологии. Особенно важно, что прирост качества сохраняется и в наиболее сложных случаях, что важно для систем, предназначенных для клинической практики.

Дополнительно стоит отметить модульную структуру системы, которая позволяет легко интегрировать дополнительные этапы анализа, такие как сегментация, а также обучение с участием врача по принципу активного обучения. Благодаря этому СППВР может не только выполнять роль вспомогательного инструмента, но и служить основой для построения гибридных моделей взаимодействия человека и искусственного интеллекта.

В перспективе планируется расширение функциональности модуля с возможностью обработки изображений, полученных с других модальностей (например, КТ, МРТ), а также внедрение механизмов самообучения модели при изменении условий съёмки или состава оборудования. Это обеспечит дополнительную устойчивость и применимость системы в реальных клинических условиях с разнообразием диагностических сценариев.

Таким образом, внедрение ансамбля нейронных сетей на этапе предварительной обработки медицинских изображений позволяет не только повысить качество входных данных, но и существенно улучшить итоговую диагностическую точность СППВР. Разработанный подход может быть масштабирован и адаптирован для различных задач медицинской визуализации, представляя собой универсальное решение в контексте цифрового здравоохранения.

Список литературы

- [Гонсалес, 2015] Гонсалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2015. – 1072 с.
- [Мищенко и др., 2024a] Мищенко И.И., Александров А.В., Шалухова М.А. Способ интеллектуального распознавания Hip-Spine-синдрома // Интегрированные модели и мягкие вычисления в искусственном интеллекте: Сб. науч. тр. XII междунар. науч.-практ. конф. «ИММВ-2024». – 2024. – С. 245-254.
- [Мищенко и др., 2024b] Мищенко И.И., Мисник А.Е., Александров А.В. Применение технологий компьютерного зрения и предварительной обработки изображений в системах поддержки принятия решений // Вестник Самарского государственного технического университета. – 2024. – № 4. – С. 6-26.
- [Попов и др., 2022] Попов А.Н., Волков К.Д., Семенова Е.А. Сегментация костных структур на рентгеновских снимках с помощью модифицированной U-Net архитектуры // Искусственный интеллект и принятие решений. – 2022. – № 1. – С. 34-45.
- [Прэтт, 2022] Прэтт У. Цифровая обработка изображений. – М.: Мир, 2022. – 998 с.
- [Ahmed et al., 2025] Ahmed M.K., Havugimana F., Moinudin K.A. MU-Net: Modified U-Net for Precise Localization and Segmentation of Lumber-Spine Regions from Sagittal Views // Recent Advances in Soft Computing and Data Mining. – Springer, 2025.
- [Guinebert et al., 2022] Guinebert S., Petit E., Bousson V., Bodard S. Automatic semantic segmentation and detection of vertebrae and intervertebral discs by neural networks // Intelligent Imaging. – 2022.
- [Kheirandish, 2023] Kheirandish F. Evaluation of U-Net model in the detection of cervical spine fractures: thesis. – Laurentian University, 2023. – 103 p.
- [Nguyen, 2022] Nguyen H.Y. Scoliosis Detection using Deep Neural Network: thesis. – Hanoi: University of Science and Technology of Hanoi, 2022. – 86 p.
- [Panse et al., 2021] Panse V., Gupta R. Medical Image Enhancement Using Global Dynamic Histogram Equalization // Int. Conf. on Communication and Networking. – 2021. – P. 45-52. – doi: 10.1109/CSNT51715.2021.9509670.
- [Yang et al., 2021] Yang R., Yu Y. Artificial convolutional neural network in object detection and semantic segmentation for medical imaging analysis // Frontiers in Oncology. – 2021. – Vol. 11. – Article ID 638182. – doi: 10.3389/fonc.2021.638182.

УДК 004.932.2

doi: 10.15622/rcai.2025.050

РЕГРЕССИОННАЯ НЕЙРОННАЯ СЕТЬ ДЛЯ ОЦЕНКИ ПОЛОЖЕНИЯ БПЛА НА ОСНОВЕ РЕГУЛЯРИЗАЦИИ ФУНКЦИИ ПОТЕРЬ И ВЕСОВЫХ КОЭФФИЦИЕНТОВ¹

Я.С. Пикалёв (*i@pikaliov.ru*)

Б.В. Павленко (*bogdanpavl2000@mail.ru*)

Институт проблем искусственного интеллекта, Донецк

В работе представлена архитектура регрессионной нейронной сети RegrUAV, разработанной для точного определения углов поворота камеры беспилотного летательного аппарата (БПЛА) по входным аэрофотоснимкам. Предложена модифицированная функция потерь Focal Huber Loss (FHL), объединяющая свойства Huber Loss и Focal Loss, что позволяет повысить робастность модели, а также компенсировать дисбаланс данных. Для улучшения качества регрессии интегрирован метод Feature Distribution Smoothing (FDS), обеспечивающий сглаживание распределения признаков и калибровку статистики в процессе обучения. Проведено тестирование на наборах данных GTA-UAV и VisLoc с использованием различных архитектур Backbone (StripNet-small, InceptionNext-tiny). Результаты показали, что применение FHL и FDS обеспечивает снижение ошибки MAE и повышение устойчивости модели к дисбалансу и вариативности данных, при этом на данных GTA-UAV архитектура StripNet-small продемонстрировала наилучшие показатели точности.

Ключевые слова: свёрточные нейронные сети, регрессия, БПЛА.

Введение

Определение угла поворота камеры беспилотного летательного аппарата (БПЛА) на основе входного изображения представляет собой важную задачу в области компьютерного зрения и обработки изображений [Пикалёв, 2024]. Эта задача имеет множество направлений применения и значительно влияет на эффективность работы БПЛА в различных сферах [Степанов, 2011].

¹ Работа выполнена в рамках федерального проекта «Развитие человеческого капитала в интересах регионов, отраслей и сектора исследований и разработок» национального проекта «Наука и университеты».

Угол наклона камеры (рис. 1) влияет на интерпретацию сцены, что критично для задач распознавания объектов, сбора данных и анализа изображений. Использование нейронных сетей и методов регрессии для определения угла поворота камеры помогает в создании более точных моделей, обученных на больших объемах данных, что улучшает их производительность, позволяя адаптировать действия БПЛА в реальном времени.

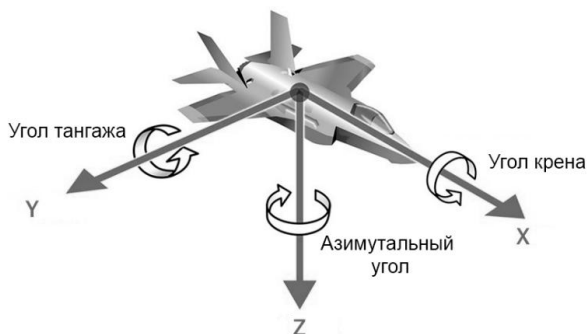


Рис. 1. Схема поворота углов БПЛА

Цель данной работы заключается в разработке и реализации модели, способной точно определять углы поворота камеры БПЛА на основе входных изображений, захваченных во время полета.

Стоит отметить, что отдельной проблемой является дисбаланс данных [Павленко, 2024]. Вместо сохранения идеального равномерного распределения по каждой категории данные часто демонстрируют искаженное распределение с длинным хвостом, когда для определенных целевых значений наблюдается значительно меньше наблюдений. Это явление ухудшает качество моделей глубокого обучения.

Таким образом, в работе были поставлены следующие задачи.

1. Спроектировать архитектуру нейронной сети RegrUAV, состоящую из таких составных частей, как Backbone, Neck, Head.
2. Изменить процесс обучения нейронной сети, использовав технику сглаживания распределения признаков (Feature Distribution Smoothing, FDS).
3. Разработать функцию потерь Focal Huber Loss (FHL), учитывающую дисбаланс данных.

Провести численные эксперименты и сравнить эффективность применения FDS и FHL.

1. Архитектура RegrUAV

1.1. Общая схема

На рис. 2 изображена общая схема RegrUAV, состоящая из:

- 1) backbone – модуль, который извлекает начальные признаки из входных изображений;
- 2) neck – модуль, который манипулирует и объединяет признаки, извлечённые из backbone;
- 3) head – модуль, который преобразует признаки в финальный выход (значения углов ориентации).

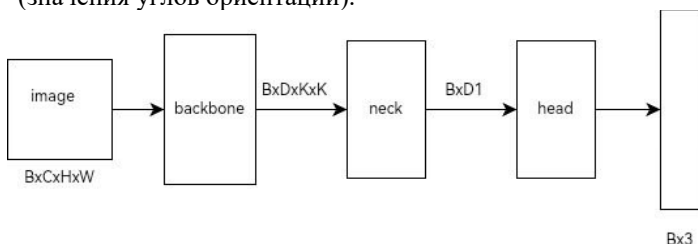


Рис. 2. Общая схема нейросети для определения положения углов наклона БПЛА (image – входное изображение, В – размер батча, С – количество каналов, Н – высота изображения, W – ширина, D – размер вектора визуальных признаков, К – количество ядер, D1 – размер одномерного тензора)

1.2. Backbone

Модуль Backbone является важнейшим компонентом в архитектурах нейронных сетей, преобразуя входные данные в сжатое скрытое представление и фиксируя существенные признаки, отбрасывая нерелевантные детали относительно поставленных целей. Предлагаемый backbone основан на модели StripNet (конфигурации StripNet-S) [Yuan, 2025].

Объекты небольшого масштаба довольно часто встречаются на аэрофотоснимках. При этом эффективность обнаружения снижается по мере увеличения масштаба объектов. В StripNet используются ортогональные крупноядерные (large-kernel) свертки в качестве основных пространственных фильтров, которые хорошо показывают себя при работе с аэрофотоснимками.

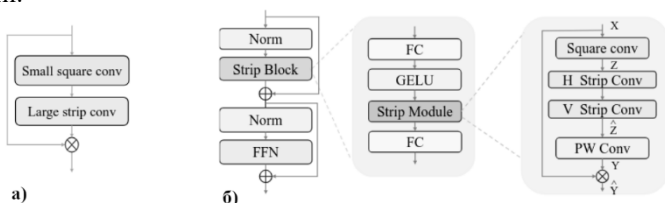


Рис. 3. Ключевые компоненты StripNet [Yuan, 2025]

StripNet состоит из базовых блоков, как показано на рис. 3,а: двух блоков с остаточной связью: strip-блока и полносвязного слоя. Как видно на рис. 3,б, strip-блок построен на стандартной свертке с малым ядром и двух свертках с большими ядрами в форме полосы (large strip convolutions) для захвата надежных признаков для объектов различного масштаба.

1.3. Neck

Модуль Neck (рис. 4) был спроектирован для захвата контекста при различных масштабах и обеспечения устойчивости (робастности) к изменениям масштаба и разрешения.

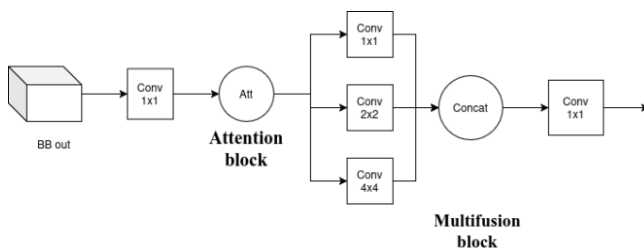


Рис. 4. Общая схема модуля Neck

Его основными блоками являются блок внимания (Attention Block, рис. 5), который улучшает точность регрессии и обеспечивает взаимодействие с разреженными слоями, улучшает различие признаков; и блок многомасштабного извлечения признаков (Multifusion Block), который объединяет признаки, собранные на разных уровнях, в единое представление, что позволяет учитывать как локальные, так и глобальные контексты.

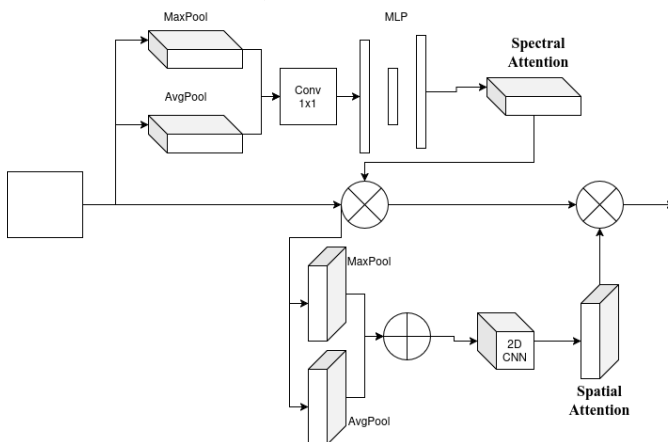


Рис. 5. Общая схема Attention Block

1.4. Head

Модуль Head (рис. 6) состоит из двух полносвязных (Fully-Connected, FC) для определения на выходе 3х углов: pitch (тангаж), yaw (рысканье), roll (крен).

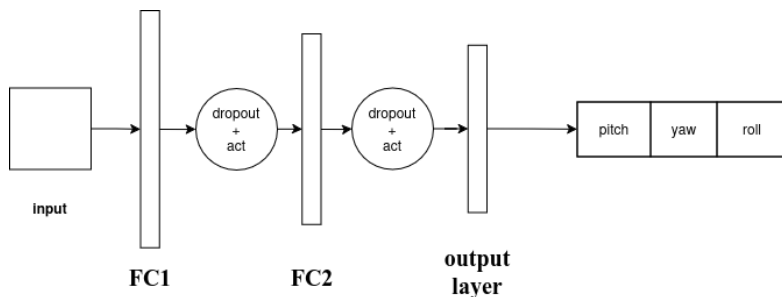


Рис. 6. Общая схема модуля Head

2. Сглаживание распределения признаков для задачи регрессии

Для решения проблемы дисбаланса данных используется FDS [Yang, 2021], которое выполняет сглаживание распределения в пространстве объектов, т.е. переносит статистику объектов между ближайшими целевыми ячейками. Эта процедура направлена на калибровку потенциально необъективных оценок распределения признаков в обучающих данных.

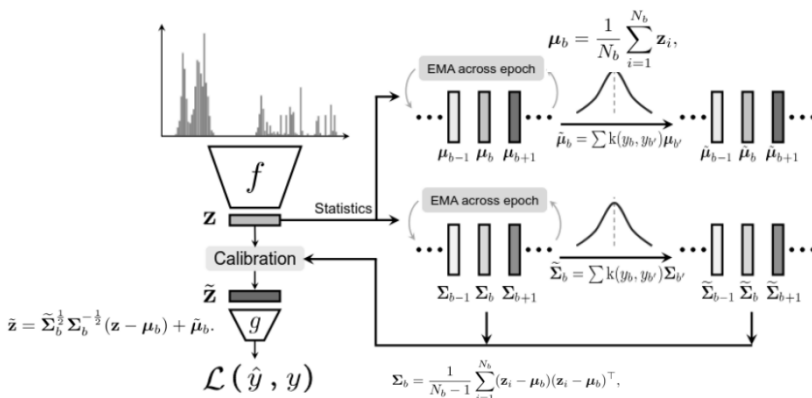


Рис. 7. Схема работы FDS в процессе обучения модели регрессии [Yang, 2021]

Для того, чтобы не ухудшить способность модели к обобщению признаков используется ковариация, чтобы отразить также взаимосвязь между различными функциональными элементами в пределах \mathbf{z} :

$$\mu_b = \frac{1}{N_b} \sum_{i=1}^{N_b} z_i, \quad (2.1)$$

$$\Sigma_b = \frac{1}{N_b - 1} \sum_{i=1}^{N_b} (z_i - \mu_b) (z_i - \mu_b)^T, \quad (2.2)$$

где N_b – общее количество образцов в ячейке. Учитывая статистику признаков, используется функция сглаживания в виде симметричного ядра $k(y_b, y_b)$, чтобы сгладить распределение среднего значения признака и ковариации по целевым ячейкам B . Это приводит к сглаженной версии статистики:

$$\mu_b = \sum_{b \in B} k(y_b, y_b) \mu_b, \quad (2.3)$$

$$\Sigma_b = k(y_b, y_b) \Sigma_b, \quad (2.4)$$

$$\tilde{z} = \tilde{z} \frac{1}{b^2} \frac{1}{b^2} (z - \mu_b) + \mu_b. \quad (2.5)$$

FDS интегрируется в RegrUAV путём добавления слоя калибровки характеристик после формирования признаков в Backbone. Для обучения модели используется динамическое обновление текущей статистики $\{\mu_b, \Sigma_b\}$ по каждой эпохе обучения. Соответственно, сглаженная статистика $\{\mu_b, \Sigma_b\}$ обновляется в разные периоды, но фиксируется в пределах каждого периода обучения.

В качестве функции обновления весов используется экспоненциальное скользящее среднее (Exponential Moving Average, EMA) [Morales-Brotons, 2024] для текущей статистики, что приводит к более стабильным и точным оценкам статистики признаков во время обучения. Затем откалиброванные признаки z передаются в регрессионную функцию и используются для вычисления потерь.

3. Focal Hubert Loss

В качестве основных функций потерь для задачи регрессии используются среднеквадратичная ошибка (Mean Square Error, MSE) и Huber Loss, отображённые на рис. 8.

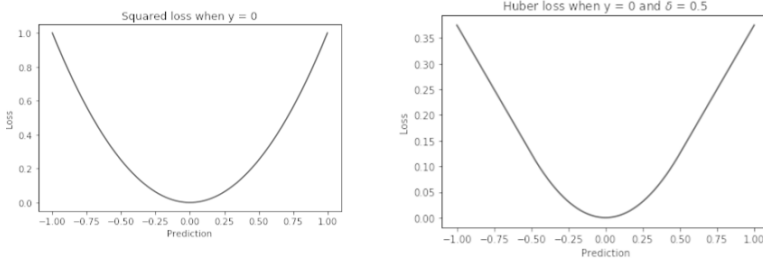


Рис. 8. Графики MSE Loss (слева) и Hubert Loss (справа)

MSE «взвешивает» большие ошибки больше, чем меньшие, в зависимости от величины ошибки. Это может приводить к выбросам в случае больших значений. Одним из решений является использование Huber Loss, который менее чувствителен к выбросам, чем MSE, поскольку существует только линейная зависимость между размером ошибки и потерей за ее пределами:

$$H_{\beta}(d) = \begin{cases} \frac{d^2}{2\beta}, & d < \beta \\ d - \frac{\beta}{2}, & d \geq \beta \end{cases} \quad (3.1)$$

$$d = y - \hat{y}. \quad (3.2)$$

В задаче классификации проблема дисбаланса данных (классов) решается при помощи Focal Loss. Эта функция уменьшает потери для «хорошо классифицированных примеров» или примеров, когда модель предсказывает правильные результаты с вероятностью p более 0.5. Дополнительно она увеличивает потери для «трудно классифицируемых примеров», когда модель предсказывает с вероятностью менее 0.5. Это свойство позволяет обращать «внимание» моделей на дисбаланс классов.

$$FL(p_t) = -\alpha_t(1 - p_t)^{\gamma} \log(p_t). \quad (3.3)$$

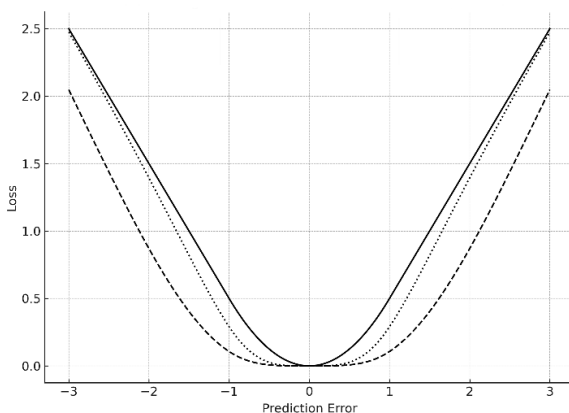


Рис. 9. Сравнение вариаций Huber Loss.

Чем выше значение γ , тем меньше потери для хорошо классифицированных примеров; α присваивает высокий вес редкому классу и низкий – распространенному.

Основная идея Focal Huber Loss (см. рис. 9) состоит в том, чтобы объединить ключевые преимущества Focal Loss с Huber Loss:

$$FHL = \frac{1}{N} \sum_{i=1}^N H_{\beta}(d_i) F_{\gamma}(d_i), \quad (3.4)$$

$$F_{\gamma}(d) = \begin{cases} (\tanh(\beta d))^{\gamma}, & \text{act} = \tanh \\ (2\sigma(\beta d) - 1)^{\gamma}, & \text{act} = \text{sigmoid} \end{cases} \quad (3.5)$$

$$\sigma(z) = \frac{1}{1+e^{-z}}. \quad (3.6)$$

Главным преимуществом FHL является то, что малые ошибки имеют пониженный вес, т.е. основной акцент делается на сложные примеры, т.к. некоторые углы обладают высоким уровнем шума. Т.е. FHL решает проблему дисбаланса данных. А экземпляры данных с небольшим уровнем шума перестают оказывать сильное влияние в обновлениях градиента после того, как они изучены.

4. Характеристика обучающего и тестового набора данных

В качестве наборов данных для обучения и тестирования были выбраны GTA-UAV [Ji, 2025] и VisLoc [Xu, 2025].

GTA-UAV создан с использованием видеоигры для моделирования и сбора большого количества аэрофотоснимков, полученных с разных высот (от 80 до 650 м) и сценариев полета. В общей сложности со всей игровой карты собрано 33763 снимка с БПЛА, охватывающих различные сцены, такие как город, горы, пустыня, лес, поле и побережье. Углы БПЛА имеют следующие диапазоны:

$$\begin{aligned} \text{roll}, \phi &\in [-10...10]; \\ \text{pitch}, \theta &\in [-100...-80]; \\ \text{yaw}, \psi &\in [-180...180]. \end{aligned}$$

Разделение на обучающие и тестовые данные проводилось посредством объединения тестовых и обучающих данных с последующим разделением, где процент обучающих данных был равен 80% (25281 для обучающих и 8427 для тестовых).

VisLoc состоит из данных, собранных из различных регионов Китая, и содержит разные географические объекты и местности: сёла, города, фермы, реки, склоны и леса. VisLoc содержит аэрофотоснимки, полученные при разной высоте полёта (400-2000 м) и углах наклона камеры, а также различную сезонность (лето и осень). Набор данных VisLoc состоит из 6742 снимков, полученных с БПЛА (с аннотацией в виде высоты, углов поворота и географических координат) и 11 спутниковых карт.

При подготовке наборов данных углы поворота приводились к положительным значениям.

5. Численные эксперименты

В ходе численных экспериментов проводится обучение и тестирование RegrUAV на основе Focal Huber Loss и средней абсолютной ошибки (Mean Absolute Error, MAE).

В данной задаче используется MAE вместо коэффициента детерминации R^2 из-за ряда недостатков: нестабильность при малой дисперсии и чувствительность к выбросам в данных.

В качестве Backbone использовались предобученные архитектуры: StripNet-small и InceptionNext-tiny [Yu, 2024].

Обучение проводилось со следующими параметрами:

- размер входного изображения: 384x384 пикселей;
- размер батча при обучении: 64;
- размер батча при тестировании: 16;
- количество эпох: 10;
- оптимизатор: Ranger [Wright, 2021].

Таблица 1

Сравнительная таблица улучшения количественных показателей
RegrUAV

Backbone	Набор данных	test loss	train loss	test MAE	train MAE
StripNet-small	GTA-UAV	0.05	0.05	0.1	0.11
InceptionNext-tiny	GTA-UAV	0.1	0.08	0.21	0.17
StripNet-small	VisLoc	0.11	0.10	0.22	0.22
InceptionNext-tiny	VisLoc	0.11	0.11	0.21	0.24

Исходя из результатов обучения в табл.1 можно сделать вывод, что Backbone StripNet-small показал минимальные значения ошибок, что указывает на высокую способность архитектуры извлекать информативные признаки с аэрофотоснимков и эффективно обучаться на данном наборе данных.

На наборе данных VisLoc рассматриваемые архитектуры показали более высокие ошибки, чем на GTA-UAV, что говорит о большей сложности задачи и о меньшем объеме данных. Разница между StripNet-small и InceptionNext-tiny минимальна по test MAE (0.22 и 0.21), но train MAE у Inception выше (0.24 и 0.22), что может указывать на чуть большую склонность к переобучению.

Таким образом, StripNet-small демонстрирует преимущество на более специфичных для аэрофотосъемки данных (GTA-UAV), где архитектура, оптимизированная под особенности снимков с БПЛА, лучше справляется с извлечением признаков.

InceptionNext-tiny на VisLoc показывает сопоставимые результаты по тестовой ошибке, что может объясняться тем, что этот набор содержит более разнообразные сцены, где универсальные архитектуры также работают хорошо. Ошибки на VisLoc примерно в два раза выше, чем на GTA-UAV, что указывает на необходимость дополнительной адаптации или применения методов доменной адаптации.

Из вышесказанного можно сделать вывод, что для задачи определения углов ориентации камеры на данных, подобных GTA-UAV, предпочтительно использовать StripNet-small. Для более сложных и разнообразных данных, подобных VisLoc, следует рассматривать улучшение методов регуляризации, увеличение объёма данных или комбинированные архитектуры, чтобы снизить ошибки.

Заключение

Спроектированная архитектура RegrUAV для задачи определения поворота углов по входному изображению с БПЛА обладает следующими ключевыми особенностями:

- использование backbone специализированного под данные формата аэрофотоснимков;
- использование Attention Block, который улучшает точность регрессии и обеспечивает взаимодействие с разреженными слоями, улучшает различие признаков;
- использование Multifusion Block, который объединяет признаки, собранные на разных уровнях, в единое представление, что позволяет учитывать как локальные, так и глобальные контексты;
- использование алгоритма сглаживания весовых коэффициентов по сле каждой эпохи обучения;
- использование функция потерь Focal Huber Loss.

RegrUAV показала на 10 эпохах следующие результаты на:

1) наборе данных GTA-UAV:

- по MAE: 0.11 на обучающих данных; 0.1 на тестовых данных;
- по FHL: 0.05 на обучающих данных; 0.05 на тестовых.

2) наборе данных VisLoc:

- по MAE: 0.22 на обучающих данных; 0.22 на тестовых данных;
- по FHL: 0.1 на обучающих данных; 0.11 на тестовых.

Список литературы

- [Пикалёв, 2024] Пикалёв Я.С. Обнаружение ключевых объектов и перекрёстная геолокализация: Анализ наборов данных и методологические перспективы // Проблемы искусственного интеллекта. – 2024. – Т. 35. – №. 4. – С. 25-37.
- [Степанов, 2011] Степанов Д.Н., Тищенко И.П. Задача моделирования полета беспилотного летательного аппарата на основе системы технического зрения // Программные системы: теория и приложения: электрон. научн. журн. – 2011. – № 4(8). – С. 33-43. – URL: http://psta.psiras.ru/read/psta2011_4_3343.pdf.

- [**Павленко, 2024**] Павленко Б.В., Пикалёв Я.С. Методика создания набора аэрофотоснимков для задачи перекрёстной геолокализации // Проблемы искусственного интеллекта. – 2024. – Т. 35, № 4. – С. 101-112.
- [**Yuan, 2025**] Yuan X. et al. Strip R-CNN: Large Strip Convolution for Remote Sensing Object Detection. arXiv 2025 //arXiv preprint arXiv:2501.03775.
- [**Yang, 2021**] Yang Y. et al. Delving into deep imbalanced regression // International conference on machine learning. – PMLR, 2021. – P. 11842-11851.
- [**Morales-Brotons, 2024**] Morales-Brotons D., Vogels T., Hendrikx H. Exponential moving average of weights in deep learning: Dynamics and benefits // arXiv preprint arXiv:2411.18704. – 2024.
- [**Ji, 2025**] Ji Y. et al. Game4loc: A uav geo-localization benchmark from game data // Proceedings of the AAAI Conference on Artificial Intelligence. – 2025. – Vol. 39, No. 4. – P. 3913-3921.
- [**Xu, 2025**] Xu W. et al. Uav-visloc: A large-scale dataset for uav visual localization // arXiv preprint arXiv:2405.11936. – 2024.19:06
- [**Yu, 2024**] Yu W. et al. Inceptionnext: When inception meets convnext // Proceedings of the IEEE/cvf conference on computer vision and pattern recognition. – 2024. – P. 5672-5683.20:16.
- [**Wright, 2021**] Wright L., Demeure N. Ranger21: a synergistic deep learning optimizer // arXiv preprint arXiv:2106.13731. – 2021.

УДК 004.85

doi: 10.15622/rcai.2025.051

ПЕРЕХОД С ОФФЛАЙН НА ОНЛАЙН ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ ТРАНСФОРМЕРНЫХ МОДЕЛЕЙ ДЛЯ РОБОТОТЕХНИЧЕСКОЙ МАНИПУЛЯЦИИ

Д.И. Поярков (*poiarkov.di@phystech.edu*)^{A,B}

А.В. Староверов (*alstar8@yandex.ru*)^{B,C}

А.И. Панов (*panov.ai@mipt.ru*)^{A,B,C}

^A Московский физико-технический институт, Долгопрудный

^B Федеральный исследовательский центр

«Информатика и управление» РАН, Москва

^C AIRI, Москва

Трансформерные модели показывают выдающиеся результаты в роботизированной манипуляции, используя для этого обучение на больших оффлайн наборах данных, но нуждаются в онлайн-дообучении для повышения надёжности стратегии. В этом исследовании сравниваются способы совместить онлайн и оффлайн методы для повышения эффективности обучения. Эксперименты показывают, что оффлайн-предобучение при помощи RL достигает целевой производительности на 32% быстрее, что подчеркивает его потенциал для снижения размера выборки при сохранении преимуществ трансформерных моделей для роботизированного управления. Исходный код доступен по адресу https://gitlab.com/cogmod_mr/isaacclab_mod.

Ключевые слова: машинное обучение, искусственный интеллект, трансформер, оффлайн-обучение, онлайн-обучение, обучение с подкреплением.

Введение

Недавние достижения в роботизированной манипуляции в значительной степени обусловлены архитектурами на основе трансформеров, обученных с помощью клонирования поведения (BC) или оффлайн-обучения с подкреплением (RL) на больших демонстрационных наборах данных [Kim et al., 2025], [Chebotar et al., 2023]. Хотя эти подходы показали многообещающие результаты с показателями успеха 70-90% (включая имею-

щиеся в наборе данных задачи) [Staroverov et al., 2023], [Li et al., 2025], многие реальные приложения требуют более высокой надежности и устойчивости.

Многие RL-алгоритмы, в частности, Proximal Policy Optimization (PPO) [Schulman et al., 2017], способны достигать высоких результатов в задачах управления роботами. Однако, эти методы обычно требуют большого количества взаимодействий со средой для сходимости, что делает их вычислительно затратными и потенциально непрактичными для реальных приложений робототехники. Это поднимает важный вопрос: можем ли мы использовать эффективное по выборке оффлайн-предобучения для ускорения сходимости RL-алгоритмов?

Комбинация оффлайн- и онлайн-обучения успешно применялось в различных областях [Ramrakhya et al., 2023], [Baker et al., 2022], включая большие языковые модели [Minaee et al., 2024]. Однако применение этого подхода к роботизированной манипуляции с трансформерными моделями является особой проблемой. Трансформеры, хотя и очень эффективны во многих приложениях, имеют ряд ограничений в работе с частично наблюдаемыми Марковскими процессами (POMDP) [Lu et al., 2024], и их совместимость с онлайн-RL остается открытым вопросом.

В этой работе мы исследуем эффективность различных стратегий оффлайн-предобучения в ускорении сходимости PPO для роботизированной манипуляции при помощи трансформеров. В частности, мы сравниваем два подхода к предобучению: традиционное клонирование поведения (BC) и оффлайн-RL с использованием алгоритма ArCHer [Zhou et al., 2024]. Наши результаты показывают, что правильное предобучение может значительно сократить количество взаимодействий со средой, необходимых для достижения PPO высокой производительности, при этом предобучение ArCHer обеспечивает сходимость до 90% процентов успеха за 32% меньше итераций по сравнению с обучением с нуля.

Наш основной вклад:

- Систематическая оценка стратегий оффлайн-предобучения для роботизированной манипуляции на основе трансформеров, демонстрирующая, что предварительное обучение ArCHer обеспечивает на 32% более быструю сходимость к 90% успешности по сравнению с обучением с нуля.
- Унифицированная архитектура трансформерной модели, которая служит как актером, так и критиком, с использованием группировки действий и отдельными выходными сегментами для оценки стратегии, что обеспечивает эффективное обучение в задачах непрерывного управления.
- Структура для перехода между оффлайн- и онлайн-фазами обучения, которая поддерживает стабильность модели с помощью заморозки актора и плавного графика скорости обучения (learning rate).

- Реализация с открытым исходным кодом, полностью совместимая с фреймворком IsaacLab, что обеспечивает воспроизводимость и модифицируемость проведённых экспериментов (https://gitlab.com/cogmod_mr/isaacclab_mod).

1. Связанные работы

1.1. Робототехника с трансформерными моделями

Трансформерная архитектура [Vaswani et al., 2017] сегодня является основой для большинства систем робототехнической манипуляции. Недавние работы, такие как RT-2 [Brohan et al., 2023], OpenVLA [Kim et al., 2024] и Octo [Octo et al., 2024] продемонстрировали эффективность трансформерных архитектур в задачах манипуляции, достигая показателей успеха 70-90% (включая новые задачи и задачи в наборах данных). Эти системы в основном полагаются на клонирование поведения для обучения.

Недавняя теоретическая работа [Lu et al., 2024] вызвала обеспокоенность по поводу ограничений трансформерных моделей в решении POMDP.

1.2. Способы предобучения моделей для робототехнического управления

Текущие подходы к предобучению моделей роботизированной манипуляции можно разделить на две группы. Первая категория, включающая RT-1 [Brohan et al., 2023a], RT-2 and OpenVLA, фокусируется на работе “zero-shot”, используя клонирование поведения на разнообразных наборах данных [Collaboration et al., 2024], [Dalal et al., 2024]. Хотя эти методы эффективны для манипуляции общего назначения, они обычно достигают плато при 70-90% успеха.

Вторая категория, представленная PerAct [Shridhar et al., 2022] и ACT [Zhao et al., 2023], делает акцент на обучении, ориентированном на конкретные задачи, на ограниченном количестве демонстраций. Наша работа нацелена на объединение этих концепций, комбинируя офлайн-предобучение с онлайн-RL.

Оффлайн-онлайн обучение в робототехнике. Недавние работы показали многообещающие результаты в объединении офлайн- и онлайн-обучения для робототехнических задач [Ramrakhaya et al., 2023], [Baker et al., 2022]. Наш подход основывается на этих результатах, однако делает акцент на проблеме онлайн-RL с трансформерной архитектурой. Подобно ArCHer [Zhou et al., 2024], мы используем иерархический актор-критик, но адаптируем его для непрерывного управления в робототехнической манипуляции.

2. Метод

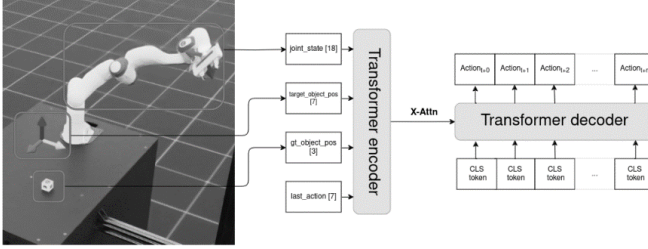


Рис. 1. Архитектура предлагаемой модели актора на основе трансформатора для непрерывного управления. Модель обрабатывает наблюдения за состоянием через энкодер и генерирует последовательность n действий через декодер

Формулировка задачи

Мы формулируем задачу роботизированной манипуляции как марковский процесс принятия решений (MDP), определяемый набором $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$, где \mathcal{S} представляет собой пространство состояний, \mathcal{A} – пространство действий, $\mathcal{P}: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ определяет динамику перехода, $\mathcal{R}: \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ – функция вознаграждения, а $\gamma \in [0, 1]$ – коэффициент дисконтирования.

Группировка действий (Action Chunking)

Вместо прогнозирования отдельных действий мы определяем группу действий длиной L [Zhao et al., 2023] как последовательность $a^{1:L} = (a^1, \dots, a^L)$, где каждое $a^i \in \mathcal{A}$. Формально, для состояния s_t в момент времени t стратегия имеет на выходе:

$$\pi_{\theta}(a^{1:L}|s_t): \mathcal{S} \rightarrow \mathcal{P}(\mathcal{A}^L),$$

где $\mathcal{P}(\mathcal{A}^L)$ обозначает распределение вероятностей для L действий.

Архитектура модели

Наша архитектура следует модели актор-критик, где оба компонента совместно используют общую трансформерную основу, но поддерживают отдельные выходные «головы».

Для непрерывного управления мы моделируем каждое действие в последовательности с помощью гауссовой стратегии. Получая на выходе модели вектор h_a^i , мы вычисляем среднее значение μ_i и стандартное отклонение σ_i для каждого действия:

$$\pi_{\theta}(a^i|s_t) = \mathcal{N}(\mu_i, \sigma_i^2).$$

Архитектура критика

Нейросеть критика использует ту же трансформерную основу, но использует разные выходные головы для оценки V и Q -значения. Для заданного состояния s_t и последовательности действий $a^{1:L}$ оценки значений вычисляются следующим образом:

$$\begin{aligned} V_\Psi(s_t) &= W_v h_e^{[CLS]} + b_v, \\ Q_\Theta(s_t, a^{1:L}) &= W_q h_d^L + b_q, \end{aligned}$$

где $h_e^{[CLS]}$ представляет собой первый токен энкодера, а h_d^L – последний токен декодера.

Стабилизация обновлений. В течение первых 100 итераций PPO мы замораживаем сеть актора, позволяя критику обучиться на новых онлайн-данных. Такое количество было взято как достаточное для адаптации критика к онлайн-данным.

Структура обучения

Наша структура обучения состоит из двух фаз: оффлайн-предобучение и онлайн-дообучение.

Мы исследуем два метода оффлайн-предобучения: ArCNeq и клонирование поведения (BC).

Предобучение ArCNeq. Алгоритм ArCNeq работает как на уровне последовательностей, так и на уровне отдельных токенов. Цели обучения Q -функции и функции значения (value-функции) следующие:

$$\begin{aligned} J_\Theta^Q &= E_{s,a,r,s' \sim \mathbb{D}} \left[\left(Q_\Theta(s, a^{1:L}) - r - \gamma V_\Psi(s') \right)^2 \right], \\ J_\Psi^V &= E_{s \sim \mathbb{D}} \left[E_{a^{1:L} \sim \pi_\Phi(\cdot|s)} \left[\left(V_\Psi(s) - Q_\Theta(s, a^{1:L}) \right)^2 \right] \right]. \end{aligned}$$

Чтобы решить проблемы оффлайн-обучения, используются цели IQL и AWR:

$$\begin{aligned} \mathcal{L}_\Psi^{IQL} &= E_{s \sim \mathbb{D}} \left[E_{a^{1:L} \sim \pi_\Phi(\cdot|s)} \left[L_2^r \left(V_\Psi(s) - Q_\Theta(s, a^{1:L}) \right) \right] \right], \\ \mathcal{L}_\Phi^{AWR} &= -E_{(s,a^{1:L}) \sim \mathbb{D}} \left[\exp(\beta A(s, a^{1:L})) \log \pi_\Phi(a^{1:L}|s) \right], \end{aligned}$$

где $L_2^r(u) = |u - 1| \mathbf{1}\{u < 0\} |u|^2$ – функция потерь на основе экспонент.

BC-предобучение. Для BC минимизируется среднеквадратичную ошибку между предсказанными и продемонстрированными действиями:

$$\mathcal{L}_\Phi^{BC} = E_{(s,a^{1:L}) \sim \mathbb{D}} \left[\sum_{i=1}^L |\hat{a}^i - a^i|^2 \right],$$

где \hat{a}^i – предсказанное действие, а a^i – использованное на шаге i действие.

Онлайн-дообучение. Для онлайн-обучения используется PPO, который оптимизирует обрезанную суррогатную цель:

$$J_{\phi}^{\text{CLIP}} = E_t[\min(r_t(\phi)\widehat{A}_t, \text{clip}(r_t(\phi), 1 - \epsilon, 1 + \epsilon)\widehat{A}_t)],$$

где $r_t(\phi) = \frac{\pi_{\phi}(a_t^{1:L}|s_t)}{\pi_{\phi_{\text{old}}}(a_t^{1:L}|s_t)}$ – отношение вероятностей, а \widehat{A}_t – оценка преимущества.

Полная функция цели онлайн-обучения объединяет потери стратегии и value-функции:

$$L_{\phi,\psi}^{\text{total}} = J_{\phi}^{\text{CLIP}} - c_1 L_{\psi}^V + c_2 H(\pi_{\phi})$$

где $L_{\psi}^V = E_t[(V_{\psi}(s_t) - R_t)^2]$ – это потеря value-функции, $H(\pi_{\phi})$ – это значение энтропии, а c_1, c_2 – весовые коэффициенты.

Комбинированное обучение. Полная процедура обучения состоит из двух фаз:

1. Оффлайн-фаза: модель предобучается с помощью ArCNeer или BC на демонстрационном наборе данных \mathcal{D} на фиксированном количестве шагов.

2. Онлайн-фаза: дообучение модели с использованием PPO со следующими изменениями:

- Сеть актора заморожена на первые 100 итераций, чтобы позволить критику адаптироваться.
- Коэффициент скорости обучения (learning rate) постепенно возвращается к целевому значению.

3. Описание и настройка задачи

Чтобы оценить наш подход, мы рассматриваем задачу роботизированной манипуляции, которая охватывает ключевые непрерывного управления. Рассматривается задача поднятия куба из фреймворка IsaacLab [Mittal et al., 2023].

3.1. Описание среды

Задача формулируется как процесс принятия решений Маркова (MDP) со следующими компонентами:

- **Пространство состояний \mathcal{S} :** Пространство состояний включает конфигурацию сочленений 7-DOF манипулятора Franka, куба в качестве объекта и плоского рабочего пространства.

- **Пространство действий \mathcal{A} :** Действия определяются как $a = [\Delta q, g] \in \mathbb{R}^7$, где $\Delta q \in \mathbb{R}^6$ представляет собой приращения положения сочленений, а $g \in \{0, 1\}$ задаёт бинарное состояние захвата.

- **Начальное состояние p_0 :** Эпизоды начинаются с манипулятора в фиксированной конфигурации. Начальное положение куба $p_0 \sim \text{Uniform}(\mathcal{W})$ случайным образом выбирается из рабочей области $\mathcal{W} \subset \mathbb{R}^2$ на поверхности стола.

- **Динамика перехода \mathcal{P} :** Окружающая среда развивается в соответствии с физическим моделированием в IsaacLab, которое обрабатывает динамику твердого тела и контактные взаимодействия между манипулятором, кубом и столом.

3.2. Формула вознаграждения

Функция вознаграждения r направлена на поощрение успешного выполнения задачи и обеспечение стабильного управления. Она состоит из нескольких слагаемых: за приближение к объекту (**reach**), поднятие объекта (**lift**), достижение цели (**goal**), точное размещение (**precise**), а также штрафов за высокую скорость сочленений (**vel**) и резкие действия (**action**).

3.3. Пространство наблюдения

Вектор наблюдения $s_t \in \mathcal{S}$ содержит:

$$s_t = [q_t, \dot{q}_t, p_c, p_g, a_{t-1}],$$

где

$q_t \in R^6$: Положения суставов

$\dot{q}_t \in R^6$: Скорости суставов

$p_c \in R^3$: Положение объекта (истинное значение)

$p_g \in R^3$: Положение цели

$a_{t-1} \in R^7$: Предыдущее действие

Это пространство наблюдения было выбрано, чтобы предоставить достаточно информации как для имитационного обучения, так и для обучения с подкреплением, сохраняя при этом относительно низкую размерность, что способствует эффективному обучению.

3.4. Настройка оффлайн-обучения

3.6.1 Набор данных экспертных демонстраций.

Для оффлайн-предобучения мы собрали набор данных \mathcal{D} , содержащий приблизительно 10 000 экспертных демонстраций. Каждая демонстрация $\tau_i \in \mathcal{D}$ представляет собой набор, содержащий состояние, действие, вознаграждения и метку о завершении эпизода:

$$\tau_i = \{(s_t, a_t, r_t, d_t)\}_{t=0}^{T_i},$$

где $T_i \leq 40$ шагов. Демонстрации были сгенерированы с использованием обученной модели, с траекториями, усеченными до 40 шагов, чтобы избежать проблем, связанных с горизонтом планирования для value-функции во время оффлайн-обучения.

4. Эксперименты

Наши эксперименты нацелены на сравнение трех стратегий обучения: (1) чистое онлайн-обучение с PPO (базовый эксперимент), (2) ВС-предобучение и PPO-дообучение и (3) предобучение иерархическим RL с последующим PPO-дообучением.

4.1. Экспериментальная установка

Все эксперименты используют одинаковый трансформерный актор-критик, как описано в разделе 3. Для справедливого сравнения мы поддерживаем согласованные гиперпараметры во всех конфигурациях обучения:

- Оффлайн-предобучение: 12 000 шагов модели со скоростью обучения $6e-5$.
- Онлайн-обучение: 5 000 итераций со скоростью обучения актера $4e-5$ и скоростью обучения критика $2e-3$.
- Количество шагов для вычисления процента успеха: 1 000 сред \times 5 эпизодов (всего 5 000 эпизодов).

Для экспериментов с предварительно обученными моделями мы замораживаем актор на первые 100 итераций онлайн-обучения.

4.2. Исследование длины выходной последовательности

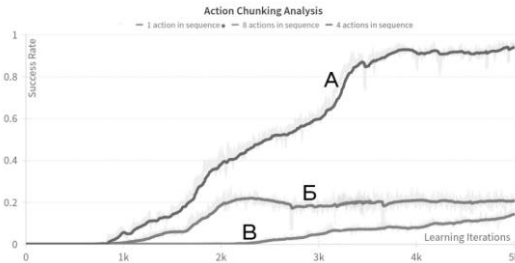


Рис. 2. Обучение моделей с разной длиной последовательности действий: 1 действие (Б), 4 действия (А), 8 действий (В)

Сперва мы исследуем влияние длины последовательности действий на эффективность обучения. Мы оцениваем длину последовательности из 1, 4 и 8 действий, используя базовый эксперимент с PPO. Результаты, показанные на рис. 2, показывают, что оптимальным числом действий для выбранной задачи является 4. Эта длина последовательности используется для всех последующих экспериментов.

4.3. Подходы к обучению

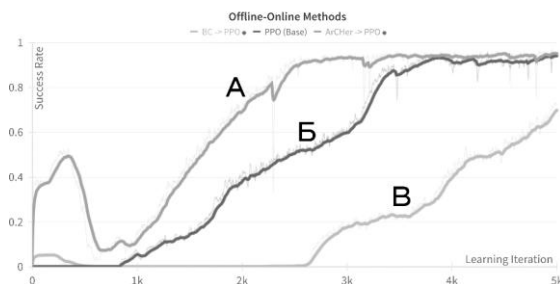


Рис. 3. Двухэтапные методы и базовый метод обучения.

РРО с нуля (Б), ВС + дообучение РРО (В), Оффлайн-RL + дообучение РРО (А)

Мы оцениваем три различных подхода к обучению. Все они имеют общую конфигурацию для инициализации сети, общие гиперпараметры и настройки симулятора.

РРО с нуля (базовый). Модель инициализируется случайным образом и обучается с использованием РРО. Процент успеха для этого эксперимента показан на рис. 3.

ВС + дообучение РРО. Модель предварительно обучена с использованием клонирования поведения на демонстрационном наборе данных, за которым следует РРО-дообучение. Критик инициализируется случайным образом в начале дообучения.

Оффлайн-RL + дообучение РРО. Модель предварительно обучена с использованием иерархического оффлайн-RL на демонстрационном наборе данных и дообучена при помощи РРО. И актер, и критик используют предобученные веса своих моделей.

5. Результаты

5.1. Производительность методов двухэтапного обучения

Эксперименты показывают, что переход между оффлайн- и онлайн-обучением создаёт видимый «провал» в производительности моделей в начале обучения. При этом, ВС-предобучение приводит к ухудшению производительности в целом.

Иерархический оффлайн-RL тоже изначально испытывает падение процента успеха, но, используя знания, полученные во время предварительного обучения, восстанавливается и в итоге достигает 90% успеха примерно на **32% быстрее**, чем базовый метод.

Экспериментальные результаты наших установок обучения приведены в табл. 1. Для индикации производительности для каждой установки мы взяли момент, в который модель достигла успешности 90%, который мы обозначили как итерацию успеха 90%.

Таблица 1

Медианный процент успеха для оффлайн → онлайн методов, выраженный в номере итерации, на которой достигнут 90% процент успеха (меньше – лучше)

Метод	90% SI
Random → PPO	3700
BC → PPO	>5000
Offline RL → PPO	2500

5.2. Производительность методов оффлайн-обучения

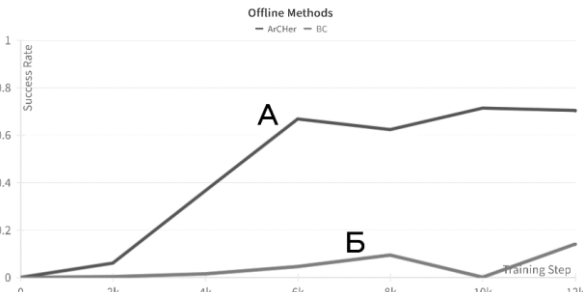


Рис. 4. Производительность оффлайн-предобучения: ArCHer (A) и BC (Б)

Производительность выбранных оффлайн алгоритмов с точки зрения показана в табл. 2 и демонстрирует прогресс по этапам обучения. Оффлайн обучение ArCHer показывает превосходную производительность по сравнению с простым BC.

Таблица 2

Успех оффлайн-методов после 12000 шагов обучения. Параметр сэмпинга показывает, применяется ли значение стандартного отклонения, взятое из модели, для рандомизации ее выходных данных

Метод	SR (с сэмпингом)	SR (без сэмпинга)
BC	14%	14%
ArCHer	70%	78%

Заключение

В работе исследуется объединение оффлайн-предобучения и онлайн-дообучения трансформеров для роботизированной манипуляции. Показано, что предварительное оффлайн-обучение с использованием RL ускоряет достижение целевой производительности на 32% по сравнению с обу-

чением с нуля, сохраняя высокую итоговую эффективность. Это подчеркивает значимость оффлайн-этапа для повышения эффективности онлайн-обучения.

В то же время выявлены сложности при переходе между фазами: простое клонирование поведения снижает качество обучения, что указывает на важность правильного задания функции цели. Полученные результаты особенно актуальны для робототехники, где критичны эффективность по выборке и надежность. Перспективным направлением является разработка более продуманных стратегий перехода и расширение подхода на более сложные задачи.

Список литературы

- [Baker et al., 2022] Baker B., Akkaya I., Zhokhov P., Huizinga J., Tang J., Ecoffet A., Houghton B., Sampedro R., Clune J. Video pretraining (vpt): Learning to act by watching unlabeled online videos // 36th Conference on Neural Information Processing Systems (NeurIPS 2022). – 2022. – doi: 2206.11795.
- [Brohan et al., 2023a] Brohan A. [et al.]. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control // Proceedings of The 7th Conference on Robot Learning. – PMLR, 2023. – doi: 10.48550/arXiv.2307.15818.
- [Brohan et al., 2023b] Brohan A. [et al.]. RT-1: Robotics Transformer for Real-World Control at Scale // Robotics: Science and Systems XIX. Daegu, Republic of Korea, 2023. – doi: 10.15607/RSS.2023.XIX.025.
- [Chebotar et al., 2023] Chebotar Y. [et al.]. Q-Transformer: Scalable Offline Reinforcement Learning via Autoregressive Q-Functions // Proceedings of The 7th Conference on Robot Learning. – PMLR, 2023. – doi: 10.48550/arXiv.2309.10150.
- [Collaboration et al., 2024] Collaboration E. [et al.]. Open X-Embodiment: Robotic Learning Datasets and RT-X Models // Proceedings of the 2024 IEEE International Conference on Robotics and Automation (ICRA). – IEEE, 2024. – doi: 10.1109/ICRA57147.2024.10611477.
- [Dalal et al., 2024] Dalal M. [et al.]. Neural MP: A Generalist Neural Motion Planner // arXiv. – 2024. – doi: 10.48550/arXiv.2409.05864.
- [Kim et al., 2025] Kim M. [et al.]. OpenVLA: An Open-Source Vision-Language-Action Model // Proceedings of The 8th Conference on Robot Learning. – PMLR, 2025. – doi: 10.48550/arXiv.2406.09246.
- [Li et al., 2025] Li X. [et al.]. Evaluating Real-World Robot Manipulation Policies in Simulation // Proceedings of The 8th Conference on Robot Learning. – PMLR, 2025. – doi: 10.48550/arXiv.2405.05941.
- [Lu et al., 2024] Lu C. [et al.]. Rethinking Transformers in Solving POMDPs // Proceedings of the 41st International Conference on Machine Learning. – 2024. – doi: 10.48550/arXiv.2405.17358.
- [Minaee et al., 2024] Minaee S. [et al.]. Large Language Models: A Survey // arXiv:2402.06196. – 2024. – doi: 10.48550/arXiv.2402.06196.
- [Mittal et al., 2023] Mittal M. [et al.]. Orbit: A Unified Simulation Framework for Interactive Robot Learning Environments // IEEE Robotics and Automation Letters. – 2023. – doi: 10.1109/LRA.2023.3270034.

- [**Ramrakhya et al., 2023**] Ramrakhya R., Batra D., Wijmans E., Das A. PIRLNav: Pretraining with Imitation and RL Finetuning for ObjectNav // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). – 2023.
- [**Schulman et al., 2017**] Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal Policy Optimization Algorithms // arXiv:1707.06347. – 2017. – doi: 10.48550/arXiv.1707.06347.
- [**Shridhar et al., 2022**] Shridhar M., Manuelli L., Fox D. Perceiver-Actor: A Multi-Task Transformer for Robotic Manipulation // Proceedings of the Conference on Robot Learning (CoRL). – 2022. – doi: 10.48550/arXiv.2209.05451.
- [**Staroverov et al., 2023**] Staroverov A., Gorodetsky A.S., Krishtopik A.S., Yudin D.A., Kovalev A.K., Panov A.I. Fine-tuning Multimodal Transformer Models for Generating Actions in Virtual and Real Environments // IEEE Access. – 2023. – Vol. 11. – P. 130548-130559. – doi: 10.1109/ACCESS.2023.3334791.
- [**Octo Team et al., 2024**] Octo Model Team, D. Ghosh, H. Walke, K. Pertsch [и др.]. Octo: An Open-Source Generalist Robot Policy // Proceedings of Robotics: Science and Systems (RSS). – 2024. – URL: <https://arxiv.org/abs/2405.12213>.
- [**Vaswani et al., 2017**] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser Ł., Polosukhin I. Attention Is All You Need // Advances in Neural Information Processing Systems. – 2017. – Vol. 30. – P. 5998-6008.
- [**Zhao et al., 2025**] Zhao T.Z., Tompson J., Driess D., Florence P., Ghasemipour S.K.S., Finn C., Wahid A. ALOHA Unleashed: A Simple Recipe for Robot Dexterity // Proc. of the 8th Conf. on Robot Learning (CoRL). PMLR 270, 2025. – P. 1910-1924.
- [**Zhou et al., 2024**] Zhou Y., Zanette A., Pan J., Levine S., Kumar A. ArCHer: Training Language Model Agents via Hierarchical Multi-Turn RL // Proceedings of the 41st International Conference on Machine Learning (ICML 2024). – PMLR, 2024.

УДК 004.896

doi: 10.15622/rcai.2025.052

НАВИГАЦИЯ МОБИЛЬНОГО РОБОТА ДО ЦЕЛЕВОЙ ТОЧКИ С ПОМОЩЬЮ ТОПОЛОГИЧЕСКОЙ КАРТЫ И ОБУЧАЕМЫХ МЕТОДОВ ДВИЖЕНИЯ

В.А. Романенко (*viromanenko_1@edu.hse.ru*)^A

К.Ф. Муравьев (*muraviev@isa.ru*)^B

^A Национальный исследовательский университет

«Высшая школа экономики», Москва

^B Федеральный исследовательский центр

«Информатика и управление» РАН, Москва

Необходимым условием автономного функционирования мобильного робота является способность выполнять автономную навигацию (т.е. достижение заданной целевой точки). Существует множество классических и обучаемых алгоритмов, эффективно решающих задачу навигации при наличии точной информации о положении целевой точки относительно робота. Как правило, такие алгоритмы апробируются в симуляционной среде с использованием точных или слабо зашумленных относительных позиций из симулятора. В данной работе представлена система навигации, использующая только наблюдения с робота и предварительно построенную топологическую карту среды. Система основана на обучаемых методах генерации действий по достижению целевой точки и на определении местоположения робота и планировании пути до цели по компактной топологической карте. Эксперименты, проведенные в симуляционных помещениях большой площади без использования информации о позиции, демонстрируют успешное достижение целевой точки.

Ключевые слова: автономная навигация, мобильные роботы, топологическое картографирование, обучение с подкреплением.

Введение

Автономная навигация является одним из ключевых аспектов функционирования автономного мобильного робота. Традиционно задача навигации формулируется как достижение роботом заданной целевой точки. Такая задача имеет множество формулировок, среди которых наиболее распространены

навигация до точки, заданной относительными координатами (PointNav) и навигация до целевой точки, заданной с помощью изображения (ImageNav). Обе формулировки имеют практическое применение и широко используются для исследования технологий автономной навигации.

В настоящее время задача PointNav, как правило, решается методами, основанными на обучении с подкреплением [Wijmans et al., 2019], [Ye et al., 2021], с использованием целеуказания, получаемого из симулятора по точным координатам робота и цели. Агент получает на вход изображение с камеры, а также расстояние до целевой точки и направление до нее. По полученным данным с помощью нейронных сетей выбирается один из возможных вариантов шагов. Задача ImageNav также решается в основном методами, основанными на обучении с подкреплением [Bono et al., 2023], [Yadav et al., 2023], однако успешность решения такой задачи ниже, чем у PointNav, за счет наличия большого количества визуально похожих друг на друга объектов и локаций. Приемлемое качество решения задачи ImageNav достигается, как правило, при небольших расстояниях от позиции робота до цели.

В реальном мире точные координаты робота, нужные для целеуказания методам решения задачи PointNav, недоступны. Также зачастую бывают недоступны внешние источники позиционирования наподобие GPS. В таком случае целеуказание невозможно, и задача навигации существенно усложняется, переходя в формулировку ImageNav или другую, не зависящую от координат. Успешность навигации при этом значительно снижается, особенно в средах большой площади, а пройденное роботом расстояние может многократно превышать длину оптимального маршрута до цели из-за необходимости поиска нужного объекта по всей среде.

Для выполнения многих практических задач может быть предварительно построена карта местности, по которой будет двигаться робот. В таких случаях задачу навигации можно привести к формулировке PointNav, решив задачу локализации, т.е. определения местоположения робота на карте по наблюдениям с его бортовых датчиков. Наиболее распространены на сегодняшний день методы локализации, использующие глобальную метрическую карту в виде облака точек [Caselitz et al., 2016], [Koide et al., 2019], а также метрические методы одновременного картирования и локализации (англ. Simultaneous Localization and Mapping, SLAM) [Labbé et al., 2019], [Koide et al., 2024]. Такие методы позволяют определять относительное положение цели для задачи PointNav, однако в помещениях большой площади возможны сбои локализации из-за высокого потребления вычислительных ресурсов. Также при большом расстоянии от робота до цели, подаваемой на вход методу решения задачи PointNav, возможна неэффективная навигация из-за отсутствия информации о структуре среды.

Альтернативой метрическим методам SLAM является топологическое картирование, т.е. представление среды в виде графа локаций. Такой подход дает компактное представление сред большой площади, а также обеспечивает быструю локализацию и быстрое планирование пути. Также использование топологической карты позволяет декомпозировать задачу достижения далеко расположенной целевой точки на несколько подзадач, заключающихся в достижении локаций, лежащих на пути от текущего положения робота до цели. Топологический подход уже реализован для решения задач ImageNav (например, в методе TSGM [Kim et al., 2023]). Однако метод TSGM соединяет ребрами далекие друг от друга локации, что приводит к снижению навигационной эффективности (показано в работе [Muravyev et al., 2023]).

В данной работе представлена система навигации по карте, основанная на обучаемом методе достижения целевой точки DD-PPO [Wijmans et al., 2019] и локализации в топологической карте с помощью метода PRISM-ТороМар [Muravyev et al., 2025]. Граф локаций, построенный методом PRISM-ТороМар, имеет малый размер в памяти робота и при этом обеспечивает надежную локализацию и быстрое планирование пути. Для подачи целеуказания методу DD-PPO между роботом и целью строится путь в графе локаций, затем в качестве цели для DD-PPO подается следующая локация в этом пути. Таким образом, обучаемый метод DD-PPO всегда получает на вход близко расположенную цель (соседняя локация), что повышает стабильность и эффективность навигации. С разработанной системой навигации проведено экспериментальное исследование в симуляционных помещениях большой площади. Исследование показало, что разработанная система обеспечивает стабильную и эффективную навигацию до целевой точки по данным с бортовых датчиков робота и предвдительно построенной карте.

1. Постановка задачи

Рассмотрим мобильную робототехническую систему, оснащенную датчиками восприятия (камера и лидар) и датчиком одометрии, движущуюся в некоторой плоской среде со свободным пространством $W \subset \mathbb{R}^2$. В момент времени t робот находится в позиции $p_t \in W$ и имеет ориентацию $r_t \in SE(3)$. Управление роботом осуществляется с помощью конечного набора управляющих действий $A = \{a^1, \dots, a^n\}$ (например, проехать вперед на 25 см, повернуть влево на 10 градусов, повернуть вправо на 10 градусов, остаться на месте). В каждый момент времени t робот получает управляющее действие $a_t \in A$. Модель движения робота $p_{t+1}, r_{t+1} = f(p_t, r_t, W, a_t)$ может быть неизвестна.

В каждый момент времени t робот получает наблюдение o_t с бортовых датчиков. Наблюдение включает изображение с камеры, облако точек с лидара, одометрию и, возможно, другие данные. Имеется предварительно построенная карта среды $M(W)$. По наблюдениям с бортовых датчиков робот может выполнять локализацию в карте. Задача локализации состоит в определении позиции робота p_t с как можно более высокой точностью:

$$Loc(M(W), o_t) = (\tilde{p}_t, \tilde{r}_t); \quad ||p_t - \tilde{p}_t||_2 + \alpha ||r_t - \tilde{r}_t||_2 \rightarrow \min.$$

Задача навигации состоит в достижении целевой точки с известными координатами $g \in W$. Для этого в каждый момент $t = 1, \dots, T$ алгоритм навигации должен генерировать управляющие воздействия a_t , приводящие робота к цели, пользуясь наблюдением с робота и данными локализации. При этом эффективность навигации оценивается по успешности достижения цели и по длине пройденного роботом пути (англ. Success weighted by Path Length, SPL):

$$a_t = Alg(\tilde{p}_t, o_t); \quad f(p_t, r_t, W, a_t) = (p_{t+1}, r_{t+1});$$

$$SPL = \frac{\sum_{i=1}^N ||p_i^* - p_{i-1}^*||_2}{\sum_{t=1}^T ||p_t - p_{t-1}||_2} I(||p_T - g||_2 < \varepsilon), \quad (1.1)$$

где $(p_0^* = p_0, p_1^*, \dots, p_N^* = g)$ – оптимальный путь от начального положения робота до целевой точки; $I(cond)$ возвращает 1, если условие $cond$ выполняется, и 0 в противном случае.

2. Предлагаемый подход

В данной работе предлагается подход к решению поставленной выше задачи, основанный на использовании предварительно построенной карты в виде графа локаций и обучаемых методах решения задачи PointNav. По графу локаций осуществляется локализация (т.е. определение локации, в которой находится робот, и его позиции в этой локации), а также построение пути до целевой точки. На построенном пути выбирается промежуточная целевая точка, которая подается на вход методу решения задачи PointNav в качестве цели. Подробное описание локализации, выбора промежуточной цели и ее достижения приведено ниже. Схема предложенного подхода изображена на рис. 1.

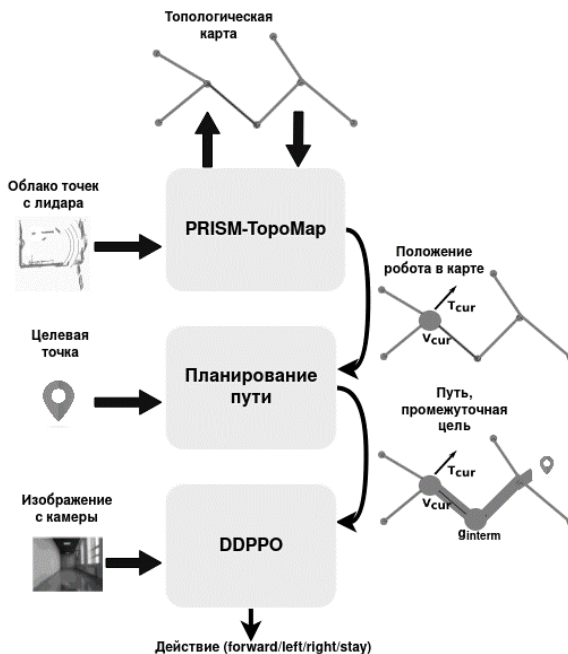


Рис. 1. Схема предложенного подхода

2.1. Локализация в топологической карте

Для построения топологической карты и последующей локализации в ней используется метод PRISM-ТopoMap [Muravyev et al., 2025]. Карта представляется в виде графа локаций, каждая из которых описывает некоторую область пространства, охватываемую одним наблюдением с робота. У каждой локации сохраняется точка, с которой сделано это наблюдение. Ребрами соединяются смежные локации, и на ребрах записываются относительные позиции между точками наблюдения локаций. Граф строится предварительно, при объезде среды в ручном режиме, либо с помощью алгоритмов исследования местности (например, [Muravyev et al., 2021]). По окончании проезда граф локаций сохраняется в постоянную память робота и для локализации при выполнении дальнейших навигационных заданий используется без изменений.

Процедура локализации метода PRISM-ТopoMap проводится в два этапа и принимает на вход предварительно построенную карту в виде графа локаций и текущее наблюдение с робота (облако точек с лидара и, опционально, изображения с передней и задней камер). Схема локализации показана на рис. 2. В начале из текущего наблюдения нейросетевой моделью извлекается дескриптор – вектор признаков небольшой размер-

ности. В графе выбираются k локаций, чьи дескрипторы наиболее близки к дескриптору текущего наблюдения. Затем по текущему облаку точек с робота строится двумерная проекция. Для каждой из k выбранных локаций вычисляется положение робота относительно ее центра путем сопоставления извлеченных из проекций особых точек. Локации, для которых сопоставление не найдено, удаляются из рассмотрения. Из оставшихся локаций выбирается та, которая имеет наиболее высокую оценку качества сопоставления.

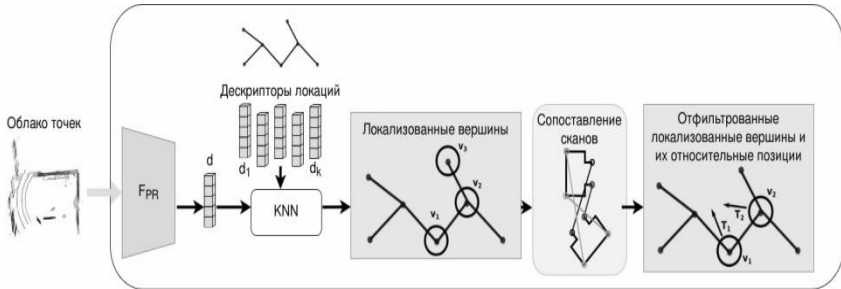


Рис. 2. Схема локализации в топологической карте

Метод PRISM-ТороМар на каждом шаге поддерживает текущее состояние робота в графе – локацию v_{cur} , в которой находится робот, и положение робота относительно ее центра T_{cur} . На старте состояние (v_{cur}, T_{cur}) определяется по результатам локализации либо задается пользователем. Далее состояние обновляется исходя из перекрытия текущего облака точек с робота и локацией v_{cur} , ребер графа и результатов локализации. Более подробное описание процедуры обновления графа доступно в работе [Муравьев, 2024].

2.2. Выбор промежуточной целевой точки

Для достижения целевой точки с заданными координатами g выбирается целевая локация v_{goal} как ближайшая к v_{cur} из всех локаций, содержащих целевую точку. Если $v_{cur} = v_{goal}$, то в качестве промежуточной целевой точки подается конечная цель: $g_{interm} = g$. В противном случае от v_{cur} до v_{goal} строится путь в графе локаций с помощью алгоритма Дейкстры. Обозначим найденный путь как $(v_0 = v_{cur}, v_1, \dots, v_n = v_{goal})$. В качестве промежуточной целевой точки выбирается точка наблюдения локации v_1 . Иллюстрация приведена на рис. 4.

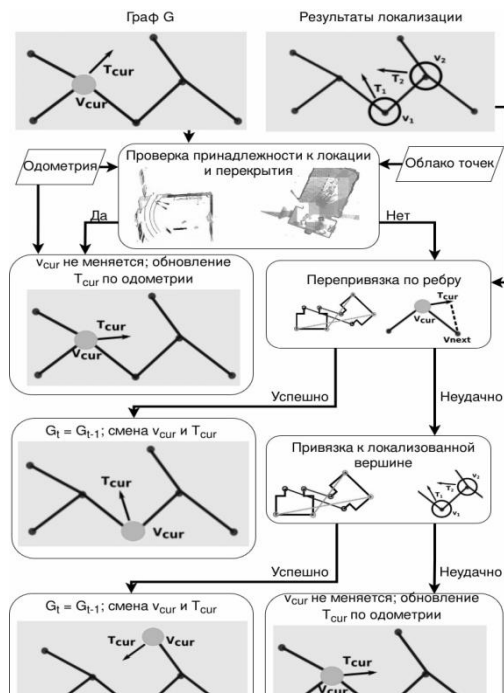


Рис. 3. Схема обновления текущего состояния робота в графе

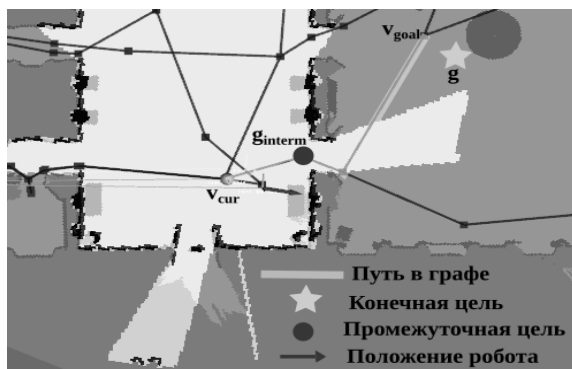


Рис. 4. Постановка промежуточной целевой точки по пути в графе локаций

2.3. Движение до промежуточной целевой точки

Данный этап базируется на классическом подходе к решению задачи PointNav, в котором агент получает на вход от среды следующие данные:

- относительное расстояние d до промежуточной точки;
- угол между направлением главной оптической оси камеры и направлением на цель;
- RGB изображение с передней камеры.

Эти данные подаются на вход нейросетевой модели на основе архитектуры SE-ResNeXt50 с рекуррентным модулем LSTM. На выходе модель выдает одно из дискретных действий: `move_forward`, `turn_left`, `turn_right` или `stop`. Модель была обучена с помощью алгоритма DDPPPO.

3. Экспериментальное исследование

3.1. Постановка эксперимента

Для оценки эффективности предложенной системы навигации было проведено экспериментальное исследование в различных симуляционных сценах с использованием и без использования «идеальных» координат.

Основные параметры экспериментов:

- **Среды и датасеты:** используются два набора сцен – MP3D и MIPТ.
- **Метрики качества:** в качестве основных метрик применяются Success Rate (SR) – доля эпизодов, в которых агент достиг целевой точки, и SPL (1.1) – эффективность пути с учетом длины пройденного пути.

Параметры сцен:

- **MP3D:** 5 сцен площадью от 100 до 700 м², каждая из которых представляет собой трехмерную модель дома.
- **MIPТ:** одна сцена, представляющая собой коридоры здания университета МФТИ. Площадь сцены – 1600 м².

В начале для оценки устойчивости моделей PointNav к шуму позиции и проводится серия экспериментов с использованием данных о позиции робота из симулятора – идеально точных и зашумленных. Затем для оценки применимости методов навигации по изображениям в условиях отсутствия точных координат робота проводятся эксперименты с решением задачи ImageNav. В конце проводится серия экспериментов с использованием локализации по наблюдениям с робота в топологической карте, предварительно построенной по данным из симулятора. На вход алгоритму PRISM-ТороМар подается облако точек и незашумленная одометрия из симулятора, однако стартовое положение робота остается неизвестным и определяется с помощью локализации на топологической карте.

3.2. Эксперименты с использованием внешних данных о позиции робота

В качестве модели для решения задачи PointNav была использована модель с архитектурой SE-ResNeXt50 + LSTM512. Всего на каждой сцене было создано 20 навигационных заданий, каждое из которых задавалось

стартовой и целевой точками. Среднее расстояние от старта до целевой точки на сценах датасета MP3D составило 2.78 м, на сцене MIPT – 5.38 м. В табл. 1 приведены результаты экспериментов проверки устойчивости с зашумлением входных данных модели на нескольких датасетах.

Из таблицы видно, что модель остается устойчивой к небольшому зашумлению входных данных. Однако, если дисперсия шума становится сравнимой с пороговым расстоянием до цели, по которому определяется успешность навигации, то успешность навигации неизбежно снижается. Если дисперсия шума ниже порогового расстояния до цели, то показатели успешности остаются высокими.

Таблица 1

Результаты экспериментов с моделью PointNav с использованием внешних данных о позиции робота

Датасет	Используемый шум	Порог расстояния до цели	SPL, %	SR, %
MP3D	-	0.2	94.2	98.9
MP3D	$N(0, 0.01)$	0.2	93.6	98.3
MP3D	$N(0, 0.03)$	0.2	91.9	97.1
MP3D	$N(0, 0.05)$	0.2	91.1	96.2
MP3D	$N(0, 0.1)$	0.2	80.0	86.9
MP3D	$N(0, 0.2)$	0.2	44.1	55.1
MIPT	-	0.2	51.6	56.3
MIPT	$N(0, 0.05)$	0.2	50.8	54.4
MIPT	-	0.3	86.8	95
MIPT	$N(0, 0.02)$	0.3	87.9	95

Эксперименты с решением задачи ImageNav

Для решения задачи ImageNav использовался тот же набор сцен и навигационных заданий, что и для задачи PointNav с использованием координат из симулятора. В качестве метода решения для тестирования был выбран метод DEBiT, так как он имеет лучшие метрики качества по сравнению с остальными методами решения задачи ImageNav. В табл. 2 приведены результаты экспериментов.

Таблица 2

Результаты экспериментов с моделью ImageNav

Датасет	Модель	SPL, %	SR, %
MIPT	DEBiT- B (Base)	40.5	63.8
MP3D	DEBiT- L (Large)	6.2	4.2

Наибольшие показатели по SR и SPL наблюдаются на датасете MIPT благодаря меньшему разнообразию сцен, тогда как для MP3D автономная визуальная навигация демонстрирует низкие результаты из-за сложности и обширности окружения. В целом эксперименты показывают низкую применимость современных методов навигации по изображениям без использования информации о координатах робота и цели.

3.3. Эксперименты с локализацией по наблюдениям с робота

Для проведения тестирования был создан новый набор из 20 навигационных заданий по первой сцене датасета MP3D. В каждом задании стартовая и целевая точки выбирались случайно. Среднее расстояние между точками старта и финиша составило 21 м. Результаты экспериментов с предложенным методом для задачи PointNav приведены в табл. 3.

Таблица 3

Результаты экспериментов с моделью PointNav с использованием незашумленного позиционирования из симулятора (GT) и локализации в топологической карте (PRISM-TopoMap)

Локализация	SPL	SR
GT	67.3	100.0
PRISM-TopoMap	59.8	95.0

За счет использования топологической карты для локализации и планирования пути значения SR и SPL незначительно снизились при переходе с незашумленного внешнего целеуказания на целеуказание по наблюдениям с робота. Такое снижение может объясняться отсутствием адаптации нейросетевой модели генерации действий к частым изменениям целевой точки. Такая адаптация может быть проведена путем дополнительного обучения нейросетевой модели.

Заключение

В данной работе представлена система навигации мобильного робота по наблюдениям с робота (видеоданные и данные одометрии) по предварительно построенной топологической карте. Для локализации и планирования маршрута используется алгоритм PRISM-TopoMap, характеризующийся надежной локализацией по легковесной топологической карте. Для навигации робота вдоль маршрута используется обучаемая нейросетевая модель DD-PPO, на вход которой подается следующая локация на маршруте. С представленной системой навигации проведено экспериментальное исследование в симуляционной среде большой площади. Исследование показало, что предложенная навигационная система позволяет успешно достигать целей при локализации по наблюдениям с датчиков робота без использования знаний о его положении. В дальнейшем плани-

руется адаптация моделей решения задачи PointNav к частой смене целевой точки, а также применения комбинаций моделей PointNav и ImageNav для повышения успешности и эффективности навигации. Планируется проведение экспериментов без использования данных из симулятора даже на этапе построения карты, а также на реальных робототехнических системах.

Список литературы

- [Муравьев, 2024] Муравьев К. Ф. Топологическое картирование помещений с использованием нейросетевой локализации и сопоставления сканов // Информационные технологии и вычислительные системы. – 2024. – № 3. – С. 28-38.
- [Bono et al., 2023] Bono G. et al. End-to-end (instance)-image goal navigation through correspondence as an emergent phenomenon // arXiv preprint arXiv:2309.16634. – 2023.
- [Caselitz et al., 2016] Caselitz T. et al. Monocular camera localization in 3D lidar maps // 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). – IEEE, 2016. – P. 1926-1931.
- [Chaplot et al., 2020] Chaplot D. S. et al. Neural topological SLAM for visual navigation // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2020. – P. 12875-12884.
- [Kim et al., 2023] Kim N. et al. Topological semantic graph memory for image-goal navigation // Conference on Robot Learning. – PMLR, 2023. – P. 393-402.
- [Koide et al., 2019] Koide K., Miura J., Menegatti E. A portable three-dimensional LIDAR-based system for long-term and wide-area people behavior measurement // International Journal of Advanced Robotic Systems. – 2019. – Vol. 16, No. 2. – P. 1729881419841532.
- [Koide et al., 2024] Koide K. et al. Glim: 3d range-inertial localization and mapping with gpu-accelerated scan matching factors // Robotics and Autonomous Systems. – 2024. – Vol. 179. – P. 104750.
- [Labbé et al., 2019] Labbé M., Michaud F. RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation // Journal of field robotics. – 2019. – Vol. 36, No. 2. – P. 416-446.
- [Muravyev et al., 2021] Muravyev K., Bokovoy A., Yakovlev K. Enhancing exploration algorithms for navigation with visual SLAM // Russian conference on artificial intelligence. – Cham: Springer International Publishing, 2021. – P. 197-212.
- [Muravyev et al., 2023] Muravyev K., Yakovlev K. Evaluation of topological mapping methods in indoor environments // IEEE Access. – 2023. – Vol. 11. – P. 132683-132698.
- [Muravyev et al., 2025] Muravyev K. et al. PRISM-TopoMap: online topological mapping with place recognition and scan matching // IEEE Robotics and Automation Letters. – 2025.
- [Wijmans et al., 2019] Wijmans E. et al. Dd-ppo: Learning near-perfect pointgoal navigators from 2.5 billion frames // arXiv preprint arXiv:1911.00357. – 2019.
- [Ye et al., 2021] Ye J. et al. Auxiliary tasks speed up learning point goal navigation // Conference on Robot Learning. – PMLR, 2021. – P. 498-516.
- [Yadav et al., 2023] Yadav K. et al. Ovrl-v2: A simple state-of-art baseline for imaganav and objectnav // arXiv preprint arXiv:2303.07798. – 2023.

ГИБРИДНЫЙ ПОДХОД ОБРАБОТКИ ОБЛАКОВ ТОЧЕК ПОМЕЩЕНИЙ НА ОСНОВЕ ГЕОМЕТРИЧЕСКОЙ СЕГМЕНТАЦИИ, КЛАСТЕРИЗАЦИИ И ГЛУБОКОГО ОБУЧЕНИЯ

О.И. Христодуло (*o-hristodulo@mail.ru*)

А.Я. Байчугуров (*trappistt@yandex.ru*)

Л.Р. Мухаметдинова (*2mllr4@mail.ru*)

Уфимский университет науки и технологий, Уфа

Предложен гибридный подход к обработке облаков точек внутренних помещений, интегрирующий алгоритмы геометрической сегментации и плотностной кластеризации с нейросетевой классификацией на основе архитектуры PointNet. Данная комбинация обеспечивает эффективное распознавание структурных элементов и объектов сцены при существенном сокращении требований к объему размеченных данных за счет перехода от поточечной семантической сегментации к классификации предварительно сегментированных кластеров.

Ключевые слова: облако точек, трёхмерная сцена, сегментация, кластеризация, нейронная сеть PointNet, классификация, семантическая сегментация.

Введение

В статье исследуется научно-практическая проблема несоответствия между возможностями автоматизированной обработки облаков точек внутренних пространств зданий и требованиями к точности пространственных моделей, что выражается в необходимости ручной коррекции значительного объема результатов автоматического распознавания объектов [Феттер и др., 2022]. Проблема особенно актуальна при построении планов помещений, где требуется доработка человеком из-за ограниченности алгоритмов в идентификации элементов внутренней структуры (стен, окон, дверей, мебели) из-за фрагментарности данных лазерного сканирования, шумов и наложения объектов.

Исследование выполняется в рамках проекта «Геоинформационная система Уфимского Университета науки и технологий». Учитывая, что университет состоит из 18 корпусов, разработка метода автоматизированной обработки облаков точек внутренних помещений является актуальной. Разрабатываемый метод оптимизирован для вузовских сред с ключевыми ограничениями: несовместимостью с современными нейросетевыми архитектурами (CUDA 9.0) и требованиями к ресурсоэффективности [Рахимова и др., 2013].

Актуальность работы обусловлена необходимостью преодоления фундаментального противоречия между возрастающими требованиями к точности пространственных моделей помещений и ограничениями существующих методов. Свёрточные нейронные сети неэффективны при обработке нативных 3D-данных из-за потери геометрических признаков при 2D-проекции [Терещенко и др., 2021]. Технические барьеры вузовских систем включают устаревшие версии CUDA, несовместимые с современными архитектурами, и требования к легковесности моделей для интеграции в ресурсоограниченные системы. Кроме того, алгоритмы демонстрируют низкую устойчивость к фрагментарности данных SLAM-сканирования и шумам кластеризации, характерным для мобильных съёмок помещений.

В данной работе предлагается гибридный подход, сочетающий геометрические алгоритмы сегментации, кластеризации (RANSAC и DBSCAN) и нейросетевую классификацию объектов с помощью модели PointNet, напрямую работающей с облаками точек. Метод работает поэтапно: RANSAC используется для выделения крупных плоскостных структур, DBSCAN – для плотностной кластеризации оставшихся объектов, а PointNet – для последующей классификации полученных кластеров по типу объекта (стол, стул, доска и др.) [Гура и др., 2023].

В отличие от существующих методов, основанных на end-to-end семантической сегментации (PointNet, PointNet++ [Qi et al., 2017]) или чисто алгоритмических подходах (например, [Xie et al., 2020]), предлагаемый гибридный подход впервые комбинирует геометрическую сегментацию, плотностную кластеризацию и изолированную классификацию объектов PointNet. Это позволяет устранить ключевые ограничения:

- Для нейросетевых методов: зависимость от огромных размеченных датасетов и чувствительность к шуму.
- Для алгоритмических методов: неспособность распознавать неметрические объекты сложной формы.
- Новизна подхода заключается в каскадной обработке данных, где кластеризация DBSCAN после RANSAC выполняет роль «предсегментации», снижая требования к данным для PointNet.

Разметка данных при этом всё ещё необходима, так как модель PointNet требует обучающего набора данных с метками классов. Однако, в отличие от задач семантической сегментации, где требуется размечать каждую точку в облаке, для задачи классификации кластеров достаточно обозначить одну метку на объект, что существенно упрощает процесс подготовки данных для обучения модели [Новикова и др., 2024].

Также проведено сравнение предложенного метода с прямой семантической сегментацией на базе PointNet, что позволяет оценить преимущества и ограничения гибридного подхода в реальных условиях. Полученные результаты обеспечивают снижение трудозатрат при создании цифровых планов помещений за счёт комбинированной обработки данных (геометрическая сегментация, кластеризация и классификации кластеров) и могут быть интегрированы в информационные, а также геоинформационные системы для автоматизации инвентаризации объектов внутренних пространств с точностью до 82.6% по взвешенной F_1 -мере.

1. Алгоритмы сегментации, кластеризации RANSAC и DBSCAN

1.1. Подготовка и фильтрация данных

В качестве тестового набора данных была выбрана трёхмерная сцена, представленная в виде облака точек в формате PLY. Исходные данные получены из открытого набора SketchUp 3D. Выбрано помещение, которое содержит 681270 облаков точек.

Перед применением сегментации данные проходят предобработку:

- Фильтрация выбросов – применяется метод Statistical Outlier Removal, основанный на анализе расстояний до ближайших соседей и удалении аномально удалённых точек [Волкова, 2013]. Это позволяет уменьшить шум и повысить точность последующей сегментации.
- Воксельная фильтрация – делит пространство на равномерные ячейки, оставляя в каждой только одну точку, что снижает плотность облака и вычислительные затраты при сохранении геометрической структуры трёхмерной сцены [Василенко, 2024]. В нашем случае, размер вокселя выбран 0,6 см на основе среднего расстояния до ближайшего соседа в облаке точек.

1.2. Сегментация плоскостей методом RANSAC

Алгоритм RANSAC (RANdom SAmple Consensus) позволяет устойчиво находить геометрические модели в данных, содержащих шум и выбросы. В контексте внутренних помещений RANSAC используется для выделения плоских структур: стен, пола, потолка.

Процедура работы RANSAC:

1. Случайный выбор минимального числа точек (обычно трёх) для построения плоскости.
2. Подсчет числа точек, находящихся вблизи этой плоскости (инлайеров).
3. Итерации повторяются фиксированное количество раз.

Найденные инлайеры удаляются из облако, и RANSAC повторяется на остатке – это называется многошаговой RANSAC-сегментацией (Multi-Order RANSAC) [Соколова и др., 2014].

Однако RANSAC выделяет только плоскости без учёта связности точек, что может приводить к ошибкам.

1.3. Кластеризация DBSCAN

Для устранения недостатков RANSAC и кластеризации оставшихся точек применяется алгоритм кластеризации DBSCAN (Density-Based Spatial Clustering of Applications with Noise), который не требует заранее заданного числа кластеров и устойчив к выбросам [Шафроненко и др., 2011].

Алгоритм работает на основе двух параметров:

- Радиус поиска соседей;
- Минимальное число кластеров для формирования кластера.

DBSCAN используется на двух этапах:

- После RANSAC: точки, принадлежащие найденной плоскости, дополнительно кластеризуются, и сохраняется только самый плотный связный сегмент, остальные – исключаются или передаются в повторную обработку.
- На остатке облака: DBSCAN выделяет кластеры точек сложной формы, например, объекты мебели, оборудования и др.

Такой каскадный подход позволяет гибко выделять как плоскости, так и произвольные объекты в помещении. Параметры алгоритмов в нашей работе подбирались вручную на основе эмпирической оценки качества сегментации с помощью визуализации. Есть методы для автоматического подбора параметров, например, радиус поиска DBSCAN можно подобрать с помощью графика расстояний до k -го соседа, выбрав значение радиуса в точке перегиба [Шафроненко и др., 2011].

Для визуального отображения результатов сегментации и кластеризации на рис. 1 каждому кластеру присвоен цвет из циклической палитры «tab20». Вследствие конечности цветового пространства (20 дискретных значений) при количестве кластеров, превышающем 20, происходит повторное использование цветов.



Рис. 1. Сегментированные и кластеризованные облака точек с помощью методов RANSAC и DBSCAN

2. Архитектура PointNet

PointNet – это нейросетевая архитектура, способная напрямую обрабатывать неупорядоченные облака точек [Qi et al., 2017]. Её ключевая особенность – использование общих (shared mlp) слоёв и симметричной функции агрегации (max pooling), что позволяет получить инвариантные к порядку точек признаки.

Общая структура сети включает следующие компоненты:

- Input transform (T-Net) – это обучаемый модуль для выравнивания облака точек.
- Shared MLP: извлекает локальные признаки.
- Feature transform (T-Net): оптимизирует пространство признаков.
- Max Pooling: симметричная функция агрегации, извлекающая глобальные признаки.

На рис. 2 представлена классическая архитектура PointNet (для классификации и семантической сегментации).

В качестве нейросетевого классификатора выбрана архитектура PointNet. Альтернативные архитектуры, такие как PointNet++ [Qi et al., 2017], не применялись в исследовании ввиду их **технической несовместимости** с требованиями проекта:

- Системы обработки данных университета работают на устаревшей версии CUDA (9.0), несовместимой с фреймворками для PointNet++.
- Техническое задание предусматривало использование легковесных моделей для интеграции в геоинформационные системы с ограниченными вычислительными ресурсами.

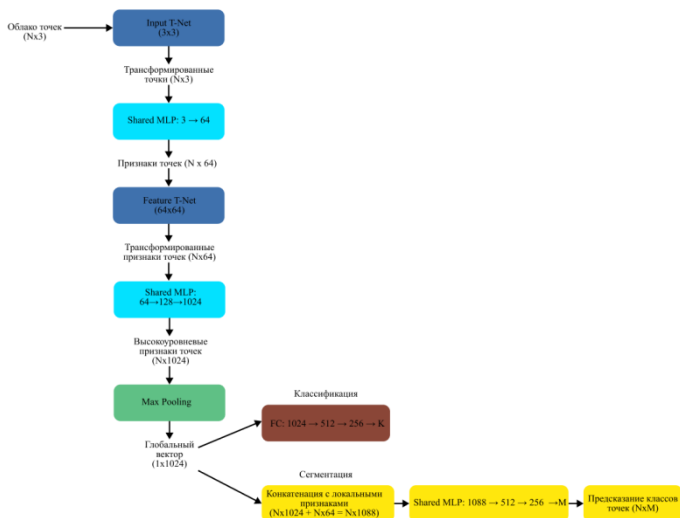


Рис. 2. Схема архитектуры PointNet

Несмотря на это, предложенный гибридный подход демонстрирует конкурентоспособную точность.

3. Гибридный подход обработки облака точек

Гибридный подход обработки облака точек сочетает алгоритмические методы (RANSAC, DBSCAN) и нейросетевую классификацию (PointNet) для обработки облака точек внутренних помещений. Такой метод особенно полезен для трёхмерных сцен, содержащих как регулярные структуры (стены, пол, потолок), так и объекты сложной формы (мебель, оборудование). На рис. 3 представлена схема гибридного подхода обработки облака точек, включающая этапы предобработки, сегментации, кластеризации и классификации.

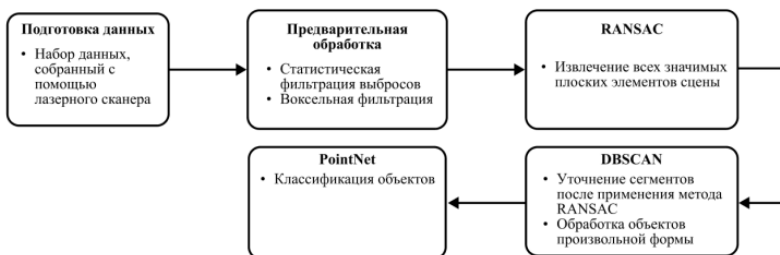


Рис. 3. Схема процесса обработки облака точек гибридным методом

Для классификации объектов применен датасет ModelNet10 [Wu et al., 2015], содержащий 10 классов мебели и сантехники в формате полигональных сеток. Данные предварительно нормализовались, а облака точек генерировались из полигонов, пример одного из файлов датасета показан на рис. 4.

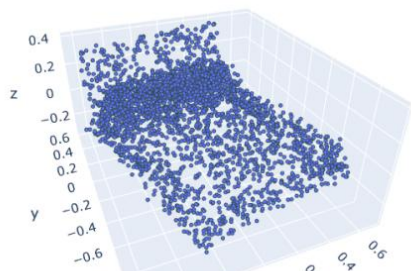


Рис. 4. Предобработанный файл датасета ModelNet10

В роли функции потерь была выбрана NLLoss (Negative Log Likelihood Loss), которая измеряет расхождение между предсказанным распределением вероятностей и истинным распределением – формула (3.1) [Гура и др., 2023]

$$NLLoss(x, y) = -\log(x_y) \, , \tag{3.1}$$

На основании построенной матрицы ошибок (рис. 5) выявлено, что модель демонстрирует наибольшую ошибку классификации при распознавании объектов класса «комод», относя их к классу «письменный стол».

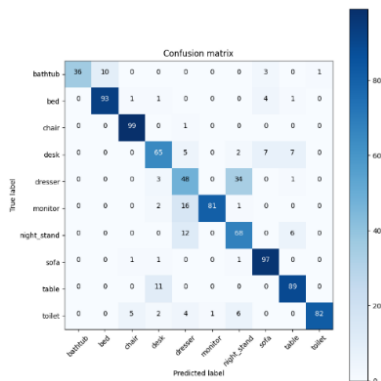


Рис. 5. Матрица ошибок классификации

В представленном гибридном подходе, рассчитана взвешенная F1-мера по формуле (3.2) [Горячкин и др., 2021], учитывающая дисбаланс классов. Данная метрика демонстрирует средневзвешенное значение F1-меры по всем классам, где веса определяются количеством экземпляров каждого класса.

$$Weighted - F1 = \sum_{i=1}^C w_i \times F1_i, \quad (3.2)$$

Применение гибридного метода обработки облаков точек позволило получить сегментированную, кластеризованную трёхмерную сцену с идентифицированными объектами, представленную на рис. 6. Объект «раковина» ошибочно отнесена к другому классу, так как такой класс отсутствует в обучающем наборе ModelNet10.



Рис. 6. Сегментированная и кластеризованная трёхмерная сцена с распознанными объектами

Для сравнения эффективности предложенного метода была реализована семантическая сегментация с помощью PointNet. Этот метод требует полной точечной разметки облаков, что делает его более ресурсоёмким и чувствительным к шуму и неоднородности плотности точек [Христовуло и др., 2023].

Для валидации семантической сегментации использован Stanford 3D Indoor Spaces Dataset (S3DIS) [Armeni et al., 2016], включающий 272 помещения с 13 классами объектов (стены, окна, мебель и др.). Выбор обусловлен репрезентативностью для задач распознавания интерьеров.

В качестве функции потерь PointNet для семантической сегментации выбрана IoU (Intersection over Union) – это метрика, используемая для оценки качества сегментации или обнаружения объектов. Она измеряет степень перекрытия между предсказанным и истинным объектом.

Также составлена матрица ошибок, представленная на рис. 7, где видно, что модель лучше всего распознаёт классы: пол, стена, потолок.

	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	stairs	clutter
ceiling	1130353	1006	20254	2147	0	33	1192	165	0	0	0	0	5647	2240
floor	0	950743	2813	0	0	0	448	0	0	0	7	0	2292	43
wall	43879	16236	1242804	22237	2504	42480	86924	5609	1009	0	7700	3462	6503	29456
beam	33090	0	67994	107685	0	262	24587	0	0	0	942	0	0	14740
column	1418	1679	70139	1061	0	4833	65939	365	24	0	157	0	0	1352
window	117	0	102418	628	0	42451	27	237	20	0	59	0	854	2873
door	0	5253	126206	9380	84	1432	159118	234	189	0	3088	0	2	3892
table	0	7350	30593	0	0	777	4303	180107	30250	174	7841	0	39	61469
chair	0	17734	9752	0	0	0	7952	51586	94371	402	2083	31	372	31147
sofa	0	1516	2499	0	0	5	57	2339	9900	8172	0	0	1143	3043
bookcase	0	5019	112571	2761	1549	3860	34532	9985	4544	135	33182	110	0	25812
board	0	145	90778	0	0	644	1240	150	223	0	1176	7014	0	4602
stairs	0	227	7383	0	0	0	0	0	0	0	0	0	3901	233
clutter	146395	47630	101356	16544	604	5433	35070	61633	61933	893	51693	4393	6033	61633

Рис. 7. Матрица ошибок семантической сегментации

Для оценки эффективности гибридного подхода был применен комплекс метрик, адаптированных к специфике задач, включая Overall Accuracy (OA) для доли верно классифицированных точек, mean Intersection over Union (mIoU) для оценки границ объектов и Weighted F₁-score для взвешенной F₁-меры с коррекцией дисбаланса классов. Сопоставлены результаты гибридного метода (RANSAC+DBSCAN+PointNet) для классификации кластеров и PointNet для поточечной семантической сегментации – табл. 1.

Таблица 1

Метод	OA	mIoU	F ₁
Гибридный подход	0.89	0.83	0.8528
PointNet	0.78	0.33	0.6524

Анализ результатов показал, что гибридный метод демонстрирует значительное преимущество с метрикой mIoU, равной 0.83, благодаря предварительной сегментации с использованием RANSAC и DBSCAN, которая формирует семантически однородные кластеры и снижает влияние шумов и фрагментарности данных. В то же время базовая PointNet имеет ограниченные возможности с метрикой mIoU, равной 0.33, из-за накопления ошибок при поточечной обработке и чувствительности к неоднородной плотности точек.

График демонстрирует превосходство гибридного метода по метрике mIoU рис. 8.

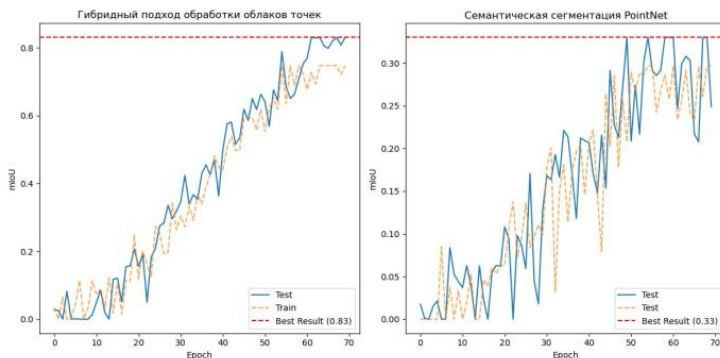


Рис. 8. Сравнение методов по метрике mIoU

Семантическая сегментация трёхмерной сцены с использованием модели PointNet показана на рис. 9. Каждая точка окрашена в соответствии с предсказанным классом. Как видно из изображения, модель корректно сегментирует крупные структурные элементы, такие как стены и пол, однако объекты меньших размеров не распознаются, что свидетельствует о недостаточной чувствительности модели к маловыраженным элементам сцены.

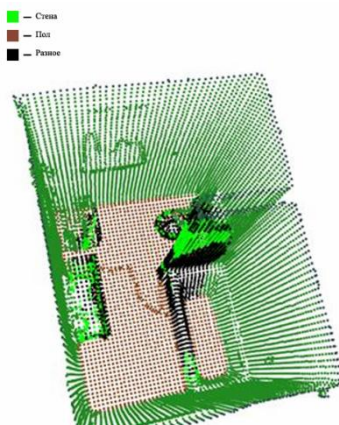


Рис. 9. Семантически сегментированная трёхмерная сцена

С помощью сравнительного анализа выделены преимущества гибридного подхода обработки облака точек:

- Снижение требований к размеченным данным.
- Интерпретируемость результатов.
- Гибкость.

Также выявлены недостатки подхода:

- Зависимость от ручного подбора параметров.
- Ручной отбор кластеров, получившихся в результате геометрической сегментации и кластеризации.
- PointNet анализирует кластеры изолированно, что может приводить к ошибкам в случаях неправильной кластеризации объектов.

Заключение

В работе предложен гибридный подход к обработке облаков точек внутренних помещений, сочетающий методы RANSAC, DBSCAN и нейросетевую архитектуру PointNet. Это повысило эффективность обработки и распознавания объектов при минимальных требованиях к разметке. Однако PointNet имеет ограничения в анализе локальных геометрических особенностей, снижая точность для сложных объектов [Христодуло и др., 2023a]. Рассматривается интеграция с иерархическими архитектурами, такими как PointNet++, для улучшения анализа пространственных контекстов. Экспериментально подтверждена точность классификации 82.6% (F1-мера) при сокращении трудозатрат. Метод применим для автоматизированного создания цифровых планов. Дальнейшие исследования направлены на адаптацию для онлайн-обработки данных и интеграции с мультимодальными источниками.

Список литературы

- [Феттер и др., 2022] Феттер М.Г., Евтушенко С.И. Проблемы применения автоматической расстановки элементов при построении информационной модели трубопроводных систем здания по облакам точек // Строительство и архитектура. – 2022. – №. 2. – С. 71-75.
- [Терещенко и др., 2021] Терещенко С.Н., Перов А.А., Осипов А.Л. Особенности применения предобученных свёрточных нейронных сетей к задачам стегоанализа графических изображений // Автометрия. – 2021. – № 4. – С. 98-105. – doi: 10.15372/AUT20210412.
- [Рахимова, 2013] Рахимова А.Р., Абдуллин А.Х. Аспекты внедрения геоинформационных технологий в Вузе // *Перспективы развития информационных технологий*. – 2013. – № 12.
- [Христодуло и др., 2023] Христодуло О.И., Вохминцев А.В., Романов М.А. Семантическая классификация и сегментация археологических объектов на основе слияния детектора объектов и 3DEF // Труды Международной российской конференции по автоматизации, RusAutoCon 2023. – 2023. – С. 122-127.
- [Гура и др., 2023] Гура Д.А., Болтовнина О.С. Применение существующих алгоритмов по классификации и кластеризации точек лазерного отражения (K-MEANS, DBSCAN, SVM) для решения задач территориального планирования // Вестник ДГТУ. Технические науки. – 2023. – № 1.

- [Новикова и др., 2024] Новикова А.Ю., Кан Е.Н. Автоматизация процессов обновления данных BIM-моделей объектов капитального строительства // ЭПИ. – 2024. – № 2.
- [Соколова и др., 2014] Соколова М.Ю., Костюхина М.А. Применение информационных технологий в научных исследованиях // Экономика и социум. – 2014. – № 2-5(11).
- [Шафроненко и др., 2011] Шафроненко А.Ю., Волкова В.В., Бодянский Е.В. Адаптивная кластеризация данных с пропущенными значениями // Радиоэлектроника, информатика, управління. – 2011. – № 2(25).
- [Волкова, 2013] Волкова Л.Л. Метод подавления шума в изображениях на основании кратномасштабного анализа // Инженерный журнал: наука и инновации. – 2013. – № 6(18). – URL: <https://cyberleninka.ru/article/n/metod-podavleniya-shuma-v-izobrazheniyah-na-osnovanii-kratnomasshtabnogo-analiza> (дата обращения: 30.05.2025).
- [Василенко, 2024] Василенко Д.В. Разработка алгоритма классификации плотных облаков точек на примере городской застройки // Вестник СГУГиТ (Сибирского государственного университета геосистем и технологий). – 2024. – № 6.
- [Горячкин и др., 2021] Горячкин Б.С., Чечнев А.А. Анализ чувствительности метрик бинарной классификации к дисбалансу данных // E-Scio. – 2021. – № 4(55).
- [Христовуло и др., 2023a] Христовуло О.И., Вохминцев А.В., Полякова Л.Ю. Система картографирования для интерпретации зон разломов в сейсмических данных на основе автокодировщика и генеративно-состязательных сетей // Материалы Международной конференции по промышленной инженерии, приложениям и производству, ICIEAM 2023. – 2023. – С. 1003-1007.
- [Wu et al., 2015] Wu Z., Song S., Khosla A., Yu F., Zhang L., Tang X., Xiao J. 3D ShapeNets: A deep representation for volumetric shapes // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. – 2015. – P. 1912-1920. – doi: 10.1109/CVPR.2015.7298801.
- [Armeni et al., 2016] Armeni I., Sener O., Zamir A. R., Jiang H., Brilakis I., Fischer M., Savarese S. 3D semantic parsing of large-scale indoor spaces // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. – 2016. – P. 1534-1543. – doi: 10.1109/CVPR.2016.170.
- [Rusu et al., 2009] Rusu R. B., Blodow N., Beetz M. Fast point feature histograms for object recognition (RANSAC/DBSCAN) // *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. – 2009. – Vol. 31, No. 8. – P. 1451-1465. – doi: 10.1109/TPAMI.2008.58.
- [Qi et al., 2017] Qi C.R., Su H., Mo K., Guibas L.J. PointNet: deep learning on point sets for 3D classification and segmentation // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. – 2017. – P. 77-85. – doi: 10.1109/CVPR.2017.16.
- [Dai et al., 2018] Dai A., Chang A. X., Savva M., Halber M., Funkhouser T., Nießner M. ScanNet: richly-annotated 3D reconstructions of indoor scenes // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. – 2018. – P. 5828-5839. – doi: 10.1109/CVPR.2017.261.

УДК 004.832.2

doi: 10.15622/rcai.2025.054

**ИССЛЕДОВАНИЕ СТРАТЕГИЙ ВЗАИМОДЕЙСТВИЯ
С БОЛЬШИМИ ЯЗЫКОВЫМИ МОДЕЛЯМИ В ЗАДАЧЕ
ОПТИМАЛЬНОГО ВЫБОРА ВРЕМЕНИ ВСТРЕЧИ¹**

А.А. Агафонов (*agafonov.a@spcras.ru*)

Санкт-Петербургский Федеральный исследовательский центр РАН,
Санкт-Петербург

В работе исследуются различные стратегии взаимодействия с большими языковыми моделями (БЯМ) при решении задачи выбора оптимального времени встречи на основе пользовательских ограничений, заданных в свободной форме на естественном языке. Подобная задача является частным случаем поддержки принятия решений в условиях неявных и изменчивых предпочтений. Анализируются две принципиально различные стратегии: (1) агентный подход, основанный на генерации формальных ограничений и последующем решении задачи оптимизации внешним алгоритмом, и (2) прямая генерация решений в виде множества временных слотов, интервалов или бинарного вектора. Результаты показывают, что агентный подход обеспечивает наилучшую точность и стабильность, особенно при увеличении числа предпочтений. В то же время, подходы прямой генерации решений оказываются эффективными при не-

¹ Работа выполнена при финансовой поддержке РНФ (проект № 25-11-00127).

большом числе ограничений и простоте формулировок, что подчеркивает важность выбора формата взаимодействия с БЯМ в прикладных интеллектуальных системах.

Ключевые слова: большие языковые модели, программирование в ограничениях, агенты ИИ, поддержка принятия решений, выявление пользовательских предпочтений, планирование встреч.

Введение

Поддержка принятия решений – одна из ключевых задач в системах искусственного интеллекта, особенно в случаях, когда необходимо учитывать предпочтения пользователя. Русскоязычный термин «системы поддержки принятия решений» (СППР), а также определение и архитектура СППР были предложены академиком О.И. Ларичевым в середине 1980-х годов [Ларичев, 1984]. Так, СППР выполняют две основные функции: 1) помощь в структуризации задачи, в построении ее модели; 2) помощь в выявлении предпочтений в рамках заданной структуры.

Особое внимание уделялось исследованию психологических аспектов принятия решений [Larichev, 1984]. При этом возможности ЭВМ по поддержке ЛПР (лица, принимающего решение) напрямую зависят от качества интерфейса и учета особенностей человеческого восприятия и переработки информации. В работах [Johnson et al., 2005], [Lichtenstein et al., 2006] отмечается, что предпочтения пользователей часто формируются в процессе выбора, а не являются фиксированными. Они могут быть непоследовательными, неполными и зависимыми от текущего контекста взаимодействия с системой.

Еще до появления больших языковых моделей (БЯМ) существовали подходы, позволяющие строить гибкие и человеко-ориентированные интерфейсы взаимодействия с системами поддержки принятия решений. В частности, в отечественной школе были разработаны методы вербального анализа решений [Ларичев, 2006], ориентированные на работу с предпочтениями, выраженными на естественном языке, и не требующие от ЛПР задания параметров в числовой форме. Появление БЯМ существенно расширило эти возможности: благодаря способности таких моделей интерпретировать и обрабатывать произвольные формулировки на естественном языке, они могут использоваться не только для извлечения и уточнения предпочтений пользователя, но и для их автоматической формализации с целью решения задач удовлетворения ограничений [Abdin et al., 2024], [Yuksekgonul et al., 2024].

В данной работе рассматривается задача оптимального выбора времени встречи как частный, но показательный пример задачи поддержки принятия решений, где предпочтения выражаются в виде ограничений на временные интервалы. Исследуются различные стратегии взаимодействия

с БЯМ для учета этих предпочтений, такие как прямая генерация решений, а также генерация формальных ограничений для использования в рамках агентного подхода. Основное внимание уделяется тому, как способ постановки задачи перед языковой моделью влияет на точность решений и их соответствие пользовательским ожиданиям.

1. Обзор современных исследований

Современные исследования подчеркивают растущую роль больших языковых моделей в задачах поддержки принятия решений, связанных с выявлением предпочтений человека. В [Lawless et al., 2024] отмечается значимость итеративного выявления предпочтений в коллаборативных системах, где предпочтения формируются по мере взаимодействия пользователя с предлагаемыми вариантами решений, в частности, в задаче планирования встреч. Их гибридная система, сочетающая БЯМ и методы программирования в ограничениях, позволяет динамически учитывать предпочтения пользователя, преодолевая разрыв между выражениями на естественном языке и формальной логикой ограничений.

В [Ahmed et al., 2024] демонстрируется, что модели, такие как GPT-4 и Llama-2, могут напрямую преобразовывать описания на естественном языке в формальные модели оптимизации. Результаты показывают, что, хотя более крупные модели дают лучшие результаты, даже БЯМ среднего размера могут быть эффективно дообучены для выполнения этой задачи. Сходные направления исследуются в рамках NL4Opt [Ramamonjison et al., 2022], а также в работах по семантическому разбору и распознаванию именованных сущностей в структурированных задачах оптимизации [Dakle et al., 2023], [Prasath et al., 2023].

Чтобы преодолеть ограничения традиционной байесовской оптимизации, которая обычно опирается на явную обратную связь (например, оценки или попарные сравнения), в [Austin et al., 2024] предлагается подход к интеграции БЯМ и байесовской оптимизации. Их метод позволяет итеративно собирать обратную связь в формате диалога на естественном языке, используя вероятностные методы для уточнения представлений модели о предпочтениях пользователя с течением времени.

Наконец, в [AhmadiTeshnizi et al., 2024] предлагается система OptiMUS, которая объединяет БЯМ и решатели задач смешанного целочисленного линейного программирования, автоматизируя весь процесс преобразования текстовых формулировок задач в формальные модели оптимизации.

В совокупности, рассмотренные работы показывают, что сочетание БЯМ с агентами предоставляет возможность эффективно выявлять, интерпретировать и формализовывать человеческие предпочтения, что является важным для создания интерактивных, человеко-ориентированных систем поддержки принятия решений, способных адаптироваться к меняющимся и контекстно-зависимым целям.

2. Постановка задачи

2.1. Оптимальный выбор времени встречи

Рассматривается задача автоматизированного выбора оптимального времени встречи на основе предпочтений пользователя, выраженных на естественном языке. Пользовательские ограничения задаются в виде свободно сформулированных высказываний, каждое из которых сопровождается числовым приоритетом, отражающим его значимость. Цель системы – выбрать временные слоты из дискретного диапазона (например, с 08:00 до 21:00 с шагом 30 минут), максимизируя суммарный приоритет удовлетворённых ограничений.

Пусть $T = \{t_1, t_2, \dots, t_k\}$ – множество временных слотов, где t_i – дискретные моменты времени; $C = \{(c_1, w_1), (c_2, w_2), \dots, (c_n, w_n)\}$ – множество ограничений, где c_i – ограничение на естественном языке, $w_i \in \mathbb{N}$ – соответствующий ему вес.

В рассматриваемой постановке ограничения упорядочены по убыванию их приоритетов (весов), а вес i -го ограничения вычисляется по формуле

$$w^i = 2^{n-i},$$

где n – общее число ограничений. Подобное экспоненциальное распределение весов гарантирует, что ограничение с наивысшим приоритетом всегда имеет наибольшее влияние на выбор решения, что моделирует ситуацию, когда существует множество ограничений, явно ранжированных по важности. Данный способ задания весов не претендует на реализм с точки зрения их получения из экспертных данных и не направлен на решение задачи их оценки, обсуждаемой, например, в работе [Анохин и др., 1997]. Вместо этого он используется для контролируемого моделирования ситуации с заранее известной иерархией приоритетов, чтобы исследовать, в какой степени большие языковые модели способны корректно интерпретировать и учитывать веса (приоритеты) при выборе оптимального решения. Однако стоит подчеркнуть, что на практике подобные веса могут определяться на основе строгости формулировок ограничений о времени встречи, что возможно реализовать как с помощью методов автоматической обработки текста, включая регулярные выражения, так и с использованием БЯМ для анализа и классификации приоритетов. Таким образом, цель задачи – найти подмножество временных слотов $T' \subseteq T$, максимизирующее сумму весов удовлетворенных ограничений.

Введем семейство булевых функций (одна для каждого ограничения): $\varphi_i: T \rightarrow \{0; 1\}$, где $\varphi_i(t) = 1$, если слот t удовлетворяет ограничению c_i . Тогда полезность одного временного слота определяется следующим образом:

$$U(t) = \sum_{i=1}^n w_i \cdot \varphi_i(t).$$

Тогда оптимальное множество временных слотов определяется по следующей формуле:

$$T^* = \arg \max_{T' \subseteq T} \sum_{t \in T'} U(t).$$

2.2. Стратегии взаимодействия с БЯМ

2.2.1. Агентный подход, основанный на генерации формальных ограничений. БЯМ используется для преобразования ограничений на естественном языке c_i в булевы функции $\varphi_i: T \rightarrow \{0, 1\}$, пригодные для обработки внешними оптимизаторами. Формально:

$$\text{БЯМ: } c_i \mapsto \varphi_i \in F, \quad F \subseteq \{f: T \rightarrow \{0, 1\}\}.$$

Далее задача решается как в п. 2.1.

2.2.2. Прямая генерация решений с помощью БЯМ. В этом подходе БЯМ принимает все множество ограничений C и возвращает решение напрямую, без явной формализации ограничений. Рассматриваются три подхода к реализации данной стратегии, отличающиеся по форме выходных данных:

Генерация множества временных слотов:

$$f_{\text{time}}: C \rightarrow \mathcal{P}_{\leq 3}(T), \quad f_{\text{time}}(C) = T', |T'| \leq 3;$$

Генерация допустимых интервалов:

$$f_{\text{interval}}: C \rightarrow \{I_1, I_2, \dots, I_l\}, I_j \subseteq T;$$

Генерация бинарного вектора:

$$f_{\text{binary}}: C \rightarrow \{0, 1\}^k, x_i = 1 \Leftrightarrow t_i \in T'.$$

3. Методология исследования

3.1. Наборы данных

Для оценки способности БЯМ формировать решения и ограничения при выявлении предпочтений относительно времени встречи был разработан набор данных, содержащий 100 пользовательских запросов на естественном языке и соответствующие им функции-ограничения с эталонным поведением.

Запросы были сгенерированы при помощи модели GPT-4o (температура – 0.7). Промпт содержал следующие требования: сгенерировать 100 запросов о времени начала встречи; указать конкретное время в каждом

запросе; указать время как часть диапазона, предпочтения, ограничений или условий (например, “Обязательно до 14:00”, “Желательно не раньше 10 утра”); использовать различные стили формулировок; выводить каждый запрос с новой строки. Затем для каждого полученного запроса была вручную сформирована функция-ограничение с эталонным поведением.

Из полученного множества запросов случайным образом было сформировано 50 наборов из 1, 3 и 5 ограничений. Данное множество наборов использовалось для оценки различных стратегий взаимодействия с БЯМ. Следует отметить, что задача исследования заключается не в получении статистически точной оценки производительности в общем случае, а в выявлении качественных различий между стратегиями при разном числе ограничений. Поскольку структура задачи и распределение наборов по числу ограничений фиксированы, даже относительно небольшое количество примеров обеспечивает устойчивые оценки средних показателей и позволяет выявить основные тенденции. Например, если качество решений заметно снижается при увеличении числа ограничений с одного до пяти, то дальнейшее увеличение их количества представляется нецелесообразным. С помощью эталонных функций было сформировано множество оптимальных решений для каждого из наборов ограничений. Оптимальное решение представляется бинарным вектором, размерность которого соответствует числу временных слотов, а значения отражают то, удовлетворяет ли слот ограничению (0 или 1). Также была рассчитана полезность каждого решения.

3.2. Оценка стратегий взаимодействия

В рамках агентного подхода оценка генерации функций-ограничений осуществлялась с использованием следующих показателей:

- Компиляция (в т.ч. ошибки выполнения) – отношение числа успешно скомпилированных функций к общему числу сгенерированных функций.
- Точность – доля временных слотов, которые функция правильно идентифицировала как удовлетворяющие предпочтению.
- Полнота – доля временных слотов, соответствующих предпочтению, которые функция правильно идентифицировала как удовлетворяющие этому предпочтению.

Временные слоты были сформированы с шагом 30 минут в интервале с 08:00 до 21:00. Для генерации функций-ограничений рассматривались четыре модели: Saiga/Gemma3 12B, GPT-4o mini, Qwen3 32B, Grok 3 Mini. Последние две являются «разсуждающими». Значение температуры во всех случаях принималось равным нулю, чтобы повысить детерминированность генерации.

Внешний решатель реализован в виде Python-класса, который выполняет полный перебор доступных временных слотов с оценкой каждого по сумме весов удовлетворенных ограничений. Ограничения задаются как функции, принимающие время и возвращающие булево значение; каждому ограничению присваивается вес. При поиске оптимального времени решатель проверяет каждый слот на выполнение всех ограничений, фиксирует полезность и список удовлетворенных ограничений. Результатом работы является набор временных слотов с максимальным суммарным весом.

В рамках стратегий прямого решения задачи, набор ограничений с соответствующими приоритетами передавался напрямую в БЯМ в составе промпта. После этого ответ модели обрабатывался и приводился к бинарному вектору из 27 элементов (число слотов), где каждый элемент указывает, выбран ли соответствующий временной слот (1) или нет (0).

Оценка решений производилась по полученным векторам с помощью следующих показателей:

- Точность – доля временных слотов, выбранных моделью, которые действительно удовлетворяют эталонным ограничениям.
- Средняя абсолютная ошибка (MAE) – средняя абсолютная разница между полезностью решения, сгенерированного моделью, и полезностью эталонного решения.
- Доля валидных ответов – доля наборов ограничений, для которых модель сформировала ответ в ожидаемом формате.

Стоит отметить, что агентный подход предполагает дополнительный этап вызова внешнего решателя, что может увеличить общее время выполнения по сравнению с прямым подходом. Однако оценка времени выполнения не являлась целью исследования, а дополнительная вычислительная нагрузка рассматривается как оправданная в условиях сложных сценариев с большим числом ограничений и изменчивыми предпочтениями. При этом доступ к моделям осуществлялся по API внешних сервисов, вносящему неконтролируемые задержки и затрудняющему прямое сравнение времени выполнения между подходами.

4. Результаты и их обсуждение

4.1. Генерация формальных ограничений

В рамках данной стратегии БЯМ используется для преобразования ограничений на естественном языке в булевы функции, пригодные для обработки внешними оптимизаторами. Оцениваются два подхода к генерации формальных ограничений. В первом случае модель генерирует функции Python, возвращающие булево значение, во втором – формальное выражение, которое затем может быть интегрировано в функцию Python.

4.1.1. Генерация ограничений. В табл. 1 приводятся результаты оценки генерации формальных ограничений. Можно видеть, что, в целом, мо-

дели успешно справляются с интерпретацией естественно-языковых запросов пользователя. Большинству моделей легче дается понимание логических выражений, нежели функций-ограничений, за исключением Saiga/Gemma3 12B.

Таблица 1

Подход к генерации	Показатель	Saiga/Gemma3 12B	GPT-4o mini	Qwen3 32B	Grok 3 Mini
Функции-ограничения	Компиляция	1.000	1.000	0.930	1.000
	Точность	0.980	0.995	0.996	0.999
	Полнота	0.995	1.000	1.000	1.000
Логические выражения	Компиляция	1.000	1.000	0.960	1.000
	Точность	0.971	0.996	0.998	0.999
	Полнота	0.975	1.000	1.000	1.000

4.1.2. Формирование решений. Качество решений при использовании данной стратегии генерации ограничений определяется значениями показателей, рассмотренными ранее. На рис. 1 представлены результаты оценки решений, сформированных внешним решателем с помощью сгенерированных выражений. Снижение доли успешных ответов Qwen3 32B связано с тем, что ряд ограничений не были скомпилированы, а, соответственно, те наборы ограничений, в которых они входили, не могли быть корректно учтены.

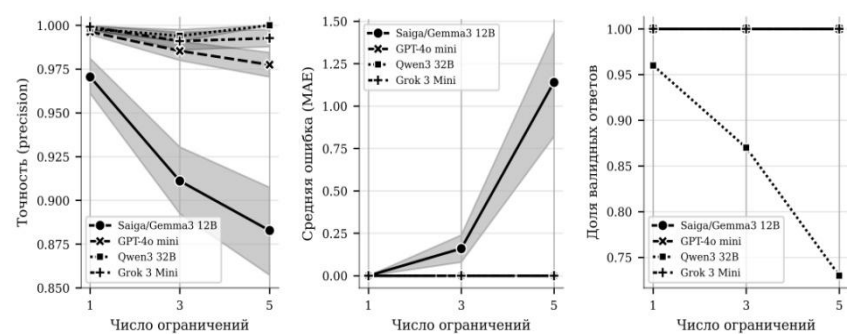


Рис. 1. Эффективность агентного подхода, основанного на генерации логических выражений

4.2. Прямое использование БЯМ

В данном подразделе оценивается способность БЯМ формировать решения напрямую, получая на вход набор ограничений с соответствующими им приоритетами. Рассматриваются три различных подхода к генера-

ции решений, отличающихся по структуре и выразительности: 1) генерация множества временных слотов; 2) генерация множества интервалов; 3) генерация бинарного вектора.

В табл. 2 приведены примеры форматов выходных данных для каждого из подходов.

Таблица 2

Входные ограничения	1. Пожалуйста, после 9:30. (приоритет 8) 2. Можем ли мы запланировать встречу после 15:30? (приоритет 4) 3. Предпочтительно время до 12:30. (приоритет 2)
Множество слотов	16:00; 17:30; 16:30
Множество интервалов	16:00–21:00
Бинарный вектор	0000000000000000111111111111

4.2.1. Генерация множества временных слотов. В данном подходе модели было задано сформировать три временных слота, которые наилучшим образом удовлетворяют заданным ограничениям. Полученные результаты приводятся на рис. 2. Можно заметить, что рассуждающие модели формируют более надежные решения по сравнению с «обычными», однако характеризуются низкой долей валидных ответов при увеличении числа ограничений.

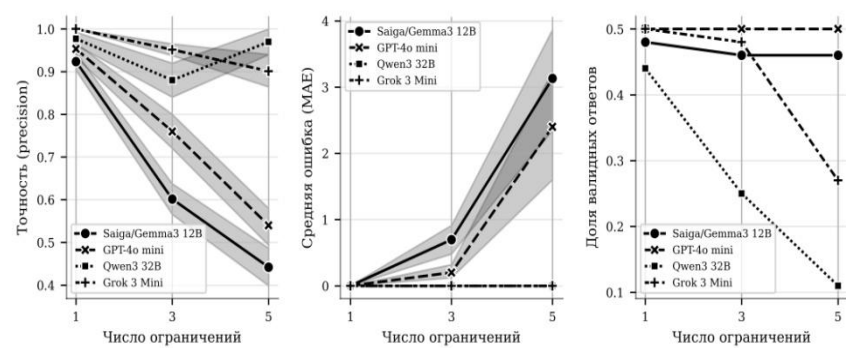


Рис. 2. Эффективность генерации множества из трех временных слотов

4.2.2. Генерация множества интервалов. В данном подходе модель формировала множество интервалов, удовлетворяющих предпочтениям пользователя. Полученные результаты приводятся на рис. 3.

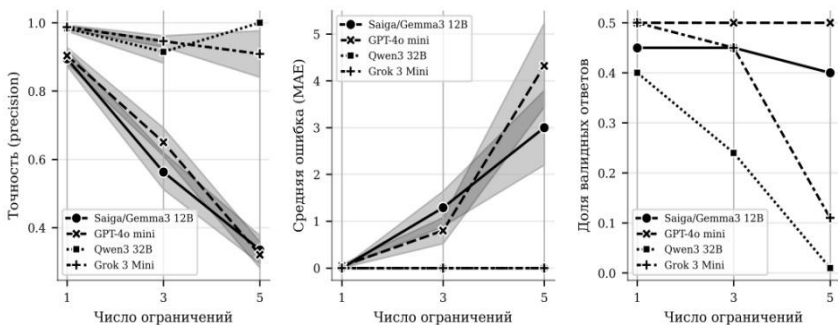


Рис. 3. Эффективность генерации множества интервалов

4.2.3. Генерация бинарного вектора. На рис. 4 представлены результаты оценки подхода генерации бинарного вектора, описывающего временной диапазон с точки зрения удовлетворения ограничений. Наибольшую точность и наименьшую ошибку полезности обеспечивают «рассуждающие» БЯМ (Qwen3 32B и Grok 3 Mini), однако они сильно подвержены отказам, которые проявляются в возврате пустого ответа на большую часть запросов. Так, для наборов, содержащих 5 ограничений, решения были вынесены не более, чем для 2 наборов из 50.

Наибольшая доля валидных ответов характерна для модели GPT-4o mini, но, также как и для Saiga/Gemma3 12B, наблюдается стремительное падение точности и рост MAE.

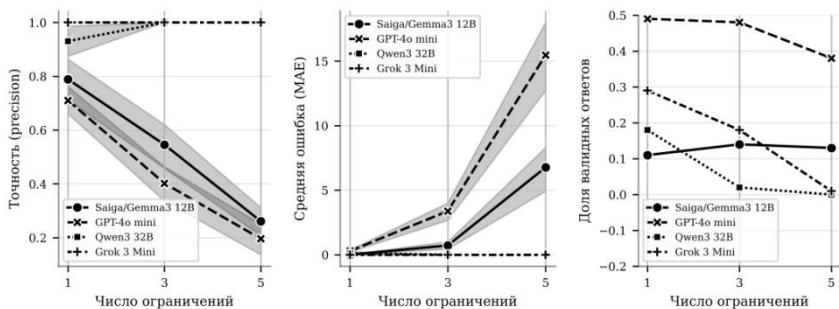


Рис. 4. Эффективность генерации бинарного вектора

4.3. Сравнение стратегий

На рис. 5 представлены обобщенные результаты оценки качества стратегий, детально рассмотренных в предыдущих пунктах. Можно заключить, что наибольшая точность и доля валидных ответов, а также наи-

меньшая ошибка (MAE) характерна для стратегии генерации ограничений, что свидетельствует о перспективности использования данной стратегии в агентных системах поддержки принятия решений.

Стратегии, основанные на прямом решении задачи с помощью больших языковых моделей, демонстрируют деградацию значений рассматриваемых показателей по мере увеличения числа ограничений. Однако следует заметить, что подход генерации множества временных слотов обеспечивает наилучшее качество решений среди стратегий, основанных на прямом решении задачи. Подход генерации множества интервалов сопоставим по качеству с предыдущим, если ограничений не более 3. Таким образом, БЯМ «проще» генерировать решения в форме множества временных слотов или интервалов, выраженных в привычной для человека форме, нежели бинарный вектор, описывающий временной диапазон с точки зрения удовлетворения ограничений.

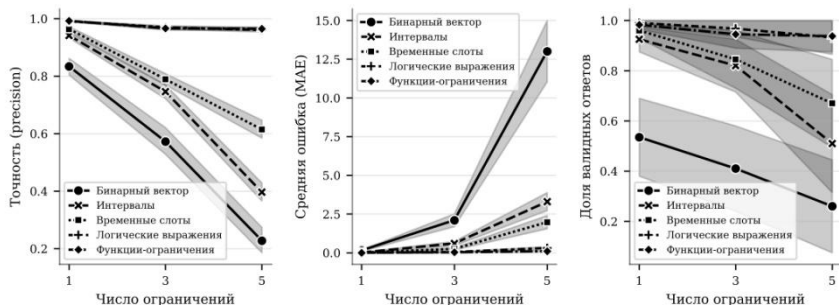


Рис. 5. Сравнение стратегий взаимодействия с БЯМ

Заключение

Проведенное исследование демонстрирует, что выбор стратегии взаимодействия с большой языковой моделью оказывает существенное влияние на качество решений в задачах поддержки принятия решений на основе естественного языка. Наиболее надежной и масштабируемой оказалась стратегия, при которой БЯМ используется для преобразования ограничений в формальные логические функции, пригодные для последующей оптимизации. Подобный агентный подход обеспечивает высокую точность даже при большом числе ограничений, что делает его перспективным для применения в реальных интеллектуальных системах.

В то же время, стратегии прямой генерации решений могут быть успешно использованы в условиях ограниченной сложности задачи – при небольшом числе предпочтений и ясной их формулировке. Особенно эффективной оказалась генерация множества временных слотов.

Список литературы

- [Анохин и др., 1997] Анохин А.М., Глотов В.А., Павельев В.В., Черкашин А.М. Методы определения коэффициентов важности критериев // Автоматика и телемеханика. – 1997. – № 8. – С. 3-35.
- [Ларичев, 2006] Ларичев О.И. Вербальный анализ решений. – М.: Наука, 2006. – 181 с.
- [Ларичев, 1984] Ларичев О.И. Проблемы взаимодействия человек-ЭВМ в системах поддержки принятия решений // Процедуры оценивания многокритериальных объектов. – 1984. – № 9. – С. 20-28.
- [Abdin et al., 2024] Abdin M.I., Gunasekar S., Chandrasekaran V., Li J., Yuksekgonul M., Peshawaria R.G., Naik R., Nushi B. KITAB: Evaluating LLMs on Constraint Satisfaction for Information Retrieval // In: Proc. 12th International Conference on Learning Representations (ICLR 2024), Vienna, Austria, 2024.
- [AhmadiTeshnizi et al., 2024] AhmadiTeshnizi A., Gao W., Udell M. OptiMUS: Scalable Optimization Modeling with (MI)LP Solvers and Large Language Models // In: Proc. 41st International Conference on Machine Learning (ICML 2024), Vienna, Austria, 2024. – P. 577-596.
- [Ahmed et al., 2024] Ahmed T., Choudhury S. LM4OPT: Unveiling the potential of Large Language Models in formulating mathematical optimization problems // INFOR: Information Systems and Operational Research. – 2024. – Vol. 62(4). – P. 559-572. – doi: 10.1080/03155986.2024.2388452.
- [Austin et al., 2024] Austin D., Korikov A., Toroghi A., Sanner S. Bayesian Optimization with LLM-Based Acquisition Functions for Natural Language Preference Elicitation // In: Proc. 18th ACM Conference on Recommender Systems (RecSys 2024), Bari, Italy, 2024. – P. 74-83. – doi: 10.1145/3640457.3688142.
- [Dakle et al., 2023] Dakle P.P., Kadioğlu S., Uppuluri K., Politi R., Raghavan P., Rallabandi S., Srinivasamurthy R. NER4OPT: Named Entity Recognition for Optimization Modelling from Natural Language // In: Proc. 20th International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research (CPAIOR 2023), Nice, France, 2023. – P. 299-319. – doi: 10.1007/978-3-031-33271-5_20.
- [Johnson et al., 2005] Johnson E.J., Steffel M., Goldstein D.G. Making better decisions: From measuring to constructing preferences // Health Psychology. – 2005. – Vol. 24(4). – P. S17-S22. – doi: 10.1037/0278-6133.24.4.S17.
- [Lawless et al., 2024] Lawless C., Schoeffer J., Le L., Rowan K., Sen S., St. Hill C., Suh J., Sarrafzadeh B. “I Want It That Way”: Enabling Interactive Decision Support Using Large Language Models and Constraint Programming // ACM Transactions on Interactive Intelligent Systems. – 2024. – Vol. 14(3). – P. 1-33. – doi: 10.1145/3685053.
- [Lichtenstein et al., 2006] Lichtenstein S., Slovic P. The Construction of Preference. – Cambridge University Press, 2006.
- [Larichev, 1984] Larichev O.I. Psychological Validation of Decision Methods // Journal of Applied Systems Analysis. – 1984. – Vol. 11. – P. 37-46.

- [Prasath et al., 2023]** Prasath G., Karande S. Synthesis of Mathematical programs from Natural Language Specifications // ArXiv. 2023. – doi: 10.48550/arXiv.2304.03287.
- [Ramamonjison et al., 2022]** Ramamonjison R., Li H., Yu T., He S., Rengan V., Banitalebi-dehkordi A., Zhou Z., Zhang Y. Augmenting Operations Research with Auto-Formulation of Optimization Models From Problem Descriptions // In: Proc. 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track (EMNLP 2022), Abu Dhabi, UAE, 2022. – P. 29-62. – doi: 10.18653/v1/2022.emnlp-industry.4.
- [Yuksekgonul et al., 2024]** Yuksekgonul M., Chandrasekaran V., Jones E., Gunasekar S., Naik R., Palangi H., Kamar E., Nushi B. Attention Satisfies: A Constraint-Satisfaction Lens on Factual Errors of Language Models // In: Proc. 12th International Conference on Learning Representations (ICLR 2024), Vienna, Austria, 2024.

УДК 004.89

doi: 10.15622/rcai.2025.055

СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В ДЕТСКОМ СПОРТЕ НА ОСНОВЕ КОМПЬЮТЕРНОГО ЗРЕНИЯ И НЕЙРО-НЕЧЕТКОГО АНАЛИЗА

Д.Г. Арсеньев (*arsenjevasp@yandex.ru*)^A

А.А. Велков (*6339743@mail.ru*)^B

А.Е. Мисник (*anton@misnik.by*)^C

М.А. Шалухова (*shaluhova.m@yandex.ru*)^{A,C}

^A Санкт-Петербургский политехнический университет
Петра Великого, Санкт-Петербург

^B Чайковская государственная академия физической культуры
и спорта, Чайковский

^C Белорусско-Российский университет, Республика Беларусь,
Могилев

В работе рассматривается концепция интеллектуальной системы поддержки принятия решений для детского спорта. Предлагается архитектура, включающая модуль компьютерного зрения, цифровой профиль спортсмена (цифровой двойник), онтологическую базу знаний и нейро-нечеткий модуль для вынесения рекомендаций. Предлагаемая система способна обеспечить дистанционный мониторинг техники выполнения упражнений, персонализировать план нагрузок с учетом возраста и подготовки, выявлять различные отклонения при выполнении тренировки в режиме реального времени. Проведено сравнение традиционного подхода в работе тренера с данными методами цифрового контроля, рассмотрены ключевые компоненты системы, и алгоритмы обработки получаемых системой данных. Интеграция подобных систем в детский спорт позволяет повышать объективность оценки прогресса, позволяет эффективно строить индивидуальные траектории развития, снижает риск травм и открывает новые прикладные возможности в области построения систем поддержки принятия решений.

Ключевые слова: ранняя спортивная специализация; интеллектуальная система поддержки решений; компьютерное зрение; цифровой двойник; нейро-нечеткие системы; сверточные нейронные сети; персонализация тренировок.

Введение

Управление тренировочным процессом и объективная оценка функционального состояния опорно-двигательного аппарата спортсмена является сложной многопараметрической задачей, требующей слаженной работы как тренера и медицинского персонала, так и непосредственно спортсмена. Рассматривая детский спорт, задача приобретает дополнительную сложность, связанную в первую очередь с особенностями формирования костно-мышечной структуры, а также обусловленная влиянием профессионального спорта на рост и развитие опорно-двигательного аппарата спортсмена.

Наиболее распространённым традиционным подходом к отбору будущих спортсменов и построению их тренировочных планов является ранняя спортивная специализация, которая подразумевает интенсивную узконаправленную нагрузку на опорно-двигательный аппарат спортсмена в период его активного формирования, особенно в технически сложных видах спорта [Baker et al., 2022]. Однако, несмотря на высокие достижения, ряд исследований демонстрирует негативные последствия ранней специализации спортсменов в долгосрочной перспективе: риск травм перегрузочного характера выше на 50-80% по сравнению с ровесниками, тренирующимися разнообразно. Нередко спортсмены уже в подростковом возрасте имеют ряд профессиональных заболеваний, связанных с особенностями практикуемого спорта, а также хронический болевой синдром [Kolt, 1999], [Popkin et al., 2019].

Учитывая колоссальные затраты времени, средств и здоровья, на подготовку профессиональных спортсменов, наиболее актуальные тенденции подготовки строятся на поиске альтернативных моделей. Модели, включающие раннюю диверсификацию в концепцию долгосрочного развития, способствуют достижению спортсменами наилучших результатов и продлению пиковой формы, за счет сочетания гармонизации тренировочной нагрузки и снижения рисков профессионального травматизма [Bergeron et al., 2015].

Для построения адекватной траектории развития юного спортсмена при обозначенном подходе необходимо учитывать множество факторов: от физического развития и техники исполнения до психоэмоционального состояния юного спортсмена. Составление таких программ является трудоёмкой: для принятия решений о режиме тренировок, степени нагрузок, необходимости отдыха или смены специализации требуется обработать колоссальный объем разноплановых данных с учетом многофакторной модели управления тренировочным процессом [Balyi, 2004].

Человеческие возможности по обработке такого объема разноплановой информации ограничены: специалист может субъективно оценивать лишь часть параметров одновременно, не застрахован от когнитивных искажений и ошибок, в результате чего значимые сигналы – признаки переутом-

ления, ошибки техники, постепенно снижающаяся мотивация – могут остаться незамеченными. Интеллектуальные системы поддержки принятия решений (СППР), широко применяемые сегодня в различных отраслях, предлагают решение указанных проблем за счет интеграции методов искусственного интеллекта, способных оперативно обрабатывать большие объемы данных, выявлять скрытые закономерности и выдавать обоснованные рекомендации [Кривецкий и др., 2013].

Применение в детском спорте интеллектуальной СППР позволит производить непрерывный мониторинг тренировочного процесса, анализировать технику выполнения упражнений, следить за динамикой физических показателей, а также учитывать субъективные данные (самочувствие, уровень усталости). На основе комплексного анализа, система способна в реальном времени сигнализировать о проблемах (например, о чрезмерной нагрузке или ухудшении техники), рекомендовать индивидуальные корректировки плана тренировок, а также оптимизировать построение тренировочного процесса, качественно улучшая его, благодаря сочетанию объективного анализа данных с экспертными знаниями тренеров [Терёхин и др., 2022].

1. Архитектура интеллектуальной системы

Предлагаемая система представляет собой СППР, объединяющую модуль предобработки и сбора данных, центральную базу данных, модель «цифрового двойника» спортсмена, интеллектуальный модуль анализа и принятия решений (включая нейронные сети и нечеткую логику), и интерфейсные приложения для пользователей – тренеров, спортсменов, врачей. В связи со спецификой работы по разработке и поддержке системных и информационно-аналитических процессов в сложных кибер-физических системах для работы был выбран онтологический подход.

Для реализация данного подхода нами была разработана программно-инструментальная среда, позволяющая специалистам в предметной области проектировать онтологию и бизнес-процессы, на основе мета-ассоциативного графа, который включает в себя процессную составляющую в виде методов и событий [Borisov et al., 2023a].

Онтологическое представление позволяет представить конфигурацию интеллектуальной системы поддержки принятия решений, которая впоследствии может быть дополнена и расширена (рис. 1).

Архитектура предусматривает возможность развертывания серверной части (включая базы данных и модуль анализа данных) на облачной платформе, либо локально в спортивной школе. Модульная реализация позволяет поэтапное введение функций: от простого модуля оценки техники выполнения по видео до сложной СППР, путем постепенного подключения необходимых модулей и компонентов. Наличие необходимых модулей исходит из задач, поставленных перед системой. Рассмотрим основные задачи, поставленные перед модулями СППР в детском спорте.

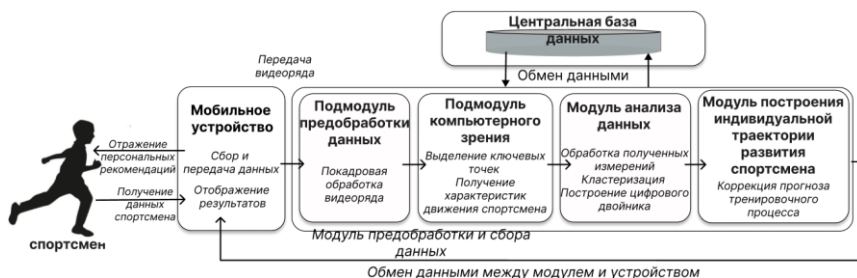


Рис. 1. Общая схема взаимодействия модулей системы

Так как система получает данные для анализа с помощью видеосъемки, необходимо использовать предобработку данных, ведь качество входных данных существенно влияет на достоверность итогового прогноза. Шум, недостаточная освещённость при съёмке, а также неоднородность качества изображений, получаемых с различных устройств – всё это существенно снижает достоверность информации, поступающей в систему. Таким образом, совершенствование модуля предобработки данных является актуальной задачей при разработке интеллектуальной СППР [Borisov et al., 2023b]. В разработанной системе предобработка данных выполняется в несколько этапов:

Применение фильтра Калмана не требует больших вычислительных ресурсов, эффективно сглаживает шум, корректирует траекторию движения объекта на основе наблюдений и нивелирует дрожание видео. В результате координаты опорных точек становятся более стабильными, что позволяет при дальнейшем анализе получать достаточную точность, для определения аномалий функционального развития опорно-двигательного аппарата, либо выявления отклонений от заданных паттернов движений (рис. 2).

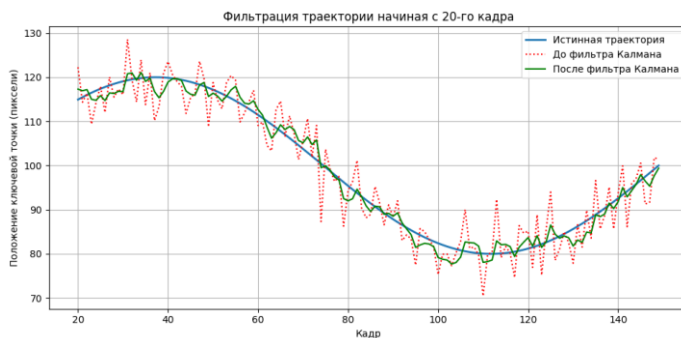






















Рис. 2. Влияние применения фильтра Калмана для предобработки положения опорных точек

Для контроля получаемых результатов, используются данные зафиксированных физиологических и антропометрических параметров спортсменов, получаемые при регулярном обследовании, и хранящиеся в системном профиле спортсмена. На следующем этапе проводится нормализация данных: координаты опорных точек приводятся к единой системе координат относительно антропометрических параметров спортсмена, что позволяет корректно оценивать тренировочный прогресс с различными индивидуальными особенностями строения. В результате на вход модели поступает нормализованная поза спортсмена, полученная путём масштабирования и центрирования. Для повышения устойчивости системы обучающая база данных была качественно улучшена и расширена до 5000 видеофайлов (рис. 3).

Result of diagnostic ↑↓	Exercise ↑↓	Input video ↑↓	Output video ↑↓
спазм круглой мышцы слев...	Вращение плеча лежа	  input.mp4	  input.mp4
спазм длинного разгибатель...	Вращение плеча лежа	  input.mp4	  input.mp4
спазм круглой мышцы слев...	Поворот бедер лежа на ...	  input.mp4	  input.mp4
спазм длинного разгибатель...	Поворот бедер лежа на ...	  input.mp4	  input.mp4
спазм круглой мышцы слев...	Повороты стопы сидя	  input.mp4	  input.mp4

Показано 1 - 6 из 5048 записей.

Рис. 3. Загруженные в систему видеоматериалы

На первом этапе видеоматериалы были обработаны модулем компьютерного зрения, после чего результаты автоматической разметки были проверены и откорректированы специалистами вручную для обеспечения высокой точности.

Одной из главных задач системы является мониторинг техники и физического состояния в реальном времени. Используя для получения данных модуль компьютерного зрения, а как источник видеоряда камеру мобильного устройства, система отслеживает положение тела ребенка при выполнении упражнений: фиксирует ключевые точки (суставы, конечности) и сравнивает технику с эталонными моделями. Для более полной характеристики движения считаются показатели нагрузки (скорость выполнения, число повторений, динамическая устойчивость), что позволяет своевременно обнаружить критические ошибки в технике выполнения и предупредить возможную травму [Hoang et al., 2021].

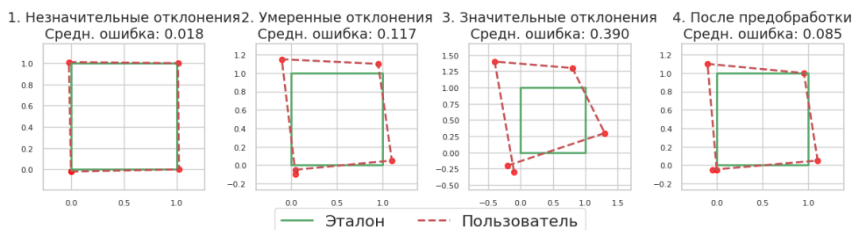


Рис. 4. Сравнение движения спортсмена и эталона

Второй частью успешного управления тренировочным процессом является адаптация нагрузок к текущему функциональному состоянию опорно-двигательного аппарата и индивидуализация тренировочных планов. Для отслеживания психоэмоционального состояния спортсменов, а также последующего анализа его влияния на тренировочный процесс и достигаемые результаты, применяется метод анкетирования. Для допуска к тренировке спортсмен проходит короткий тест в приложении, что позволяет специалисту учесть самооценку состояния, а также внести свои пометки в блок анкетирования. Регулярный учет психоэмоционального состояния спортсмена, при достаточном времени наблюдения, позволяет адаптировать тренировочную нагрузку под индивидуальные особенности. На основе данных о прогрессе каждого спортсмена, с помощью методов нейро-нечеткого анализа, позволяющих имитировать рассуждения опытного тренера (в виде набора правил если-то) и одновременно обучаться на накопленных данных, система предлагает тренеру возможные корректировки в текущий тренировочный план: отслеживая снижение либо устойчивый рост качественных показателей во время тренировочного процесса, согласно заложенным правилам система предложит либо снизить нагрузку на 10–15%, либо рекомендует усложнение задач [Jang, 1993].

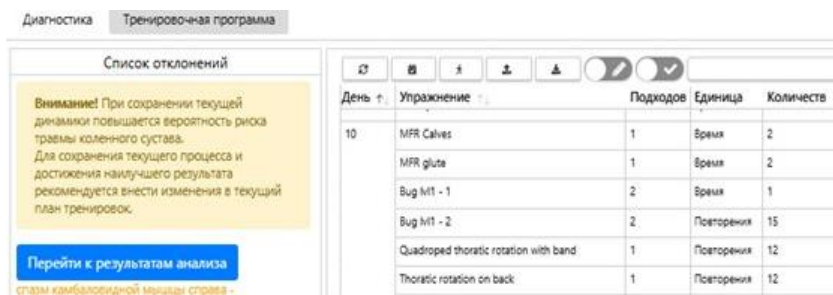


Рис. 5. Рекомендательная система. Оценка риска травмы

Для долгосрочного планирования и построения адекватных траекторий развития спортсменов на продолжительном отрезке времени необходимо прогнозировать потенциальные риски и ожидаемые результаты. Аккумулируя и анализируя данные многих спортсменов различных направлений подготовки, нейросетевые алгоритмы способны прогнозировать вероятность определенных исходов. Наблюдая рост вероятности риска травмы, отслеживаемый по совокупности факторов – увеличение объема тренировок, эмоциональное состояние, ростовой скачок у подростка, возникновение симптомов перетренированности – система не только оповестит тренера о необходимости профилактических мер, но и предложит конкретные пути решения. Соотнося прогноз вероятности риска травм с прогнозом роста спортивных результатов, система позволяет обоснованно строить и корректировать долгосрочный план подготовки [Wang et al., 2022].

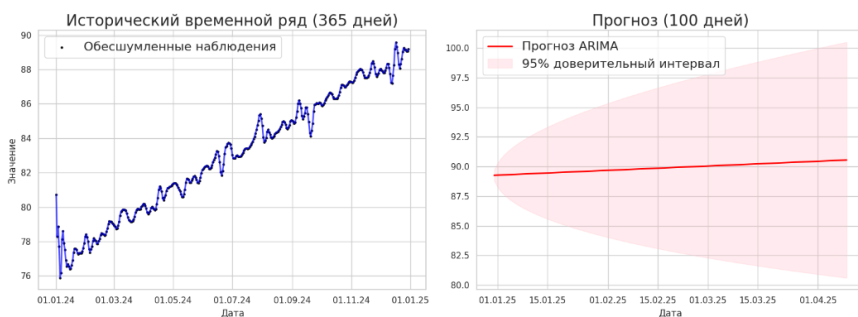


Рис. 6. Долгосрочная траектория развития спортсмена

Еще одним условием управления тренировочным процессом является получение обратной связи и поддержание мотивации. Автоматизация составления отчетов о ходе тренировочного прогресса и результатах позволяют визуализировать прогресс в показателях, что помогает в целеполагании спортсмена. Обратная связь системы объективна, что повышает мотивацию у детей, есть возможность вносить элементы игровой методики (достижения) и делает процесс тренировки более интересным. Родители же получают конкретные результаты в виде отчетов, а также уверенность, что за состоянием ребенка ведется тщательное наблюдение.

Отметим аспект накопления знаний и опыта: накапливая данные о тысячах тренировочных сессий, система фактически создает базу знаний, из которой можно извлекать новые неочевидные закономерности, позволяя влиять на траектории развития навыков, выявлять маркеры перетренированности на ранних стадиях, снижая риск спортивного травматизма, а также объективно оценивать возможный прогресс [Borisov et al., 2023b].

2. Сравнительный анализ СППР и классического подхода

Интеллектуальная СППР для детского спорта может быть внедрена в разных форматах – от персонального тренера-ассистента отдельному спортсмену до комплексной системы мониторинга, однако цифровой подход при любом из сценариев имеет существенные отличия от классического.

Проанализировав предметную область, ключевые отличия подходов представлены в виде табл. 1.

Таблица 1

Аспект тренировочного процесса	Классический подход	Цифровой подход
1. Объективность оценки техники и нагрузок		
Форма оценки	Субъективная, по впечатлениям тренера	Автоматическая, по метрикам
Инструменты	Визуальное наблюдение, разовые тесты	Видео аналитика, алгоритмы СППР
Точность	Ошибки легко упустить, мелкие отклонения игнорируются	Высокая детализация: углы, частота движений, и др.
Регулярность контроля	Периодический	Непрерывный в реальном времени,
2. Персонализация тренировок		
Подход	Унифицированная программа для группы	Индивидуальный план на базе данных спортсмена
Основание для плана	Возраст, уровень, личный опыт тренера	Профиль спортсмена, цифровой двойник, динамика прогресса, кластеризация
Частота корректировок	Эпизодически, по жалобам или падению результатов	Регулярно (неделя, день) в ответ на изменения
Учет индивидуальных параметров	Ограниченный: реакция по факту	Автоматически учитываются рост, уровень техники, состояние и мотивация
3. Раннее выявление проблем		
Обнаружение проблем	По жалобам или ухудшению техники/результата	По снижению биомеханических показателей или отклонениям от динамики
Типичные риски	Усталость, травмы, выгорание замечаются поздно	СППР сигнализирует заранее при снижении параметров или психоэмоциональном состоянии

Аспект тренировочного процесса	Классический подход	Цифровой подход
Реакция тренера/системы	После факта обнаружения проблемы	Прогноз вероятности риска и предложение корректировок программы
4. Масштабируемость и охват		
Максимальная нагрузка на тренера	До 10–15 человек эффективно	30+ спортсменов под контролем через цифровые панели
Индивидуализация при большой группе	Сложно, качество падает	Сохраняется благодаря автоматизированной аналитике
Удалённое сопровождение	Трудно или невозможно	Возможно через облако, тренер получает цифровые отчёты
5. Принятие стратегических решений		
Основа решений	Опыт, ощущения, “интуиция” тренера	Данные, аналитика, прогнозные сценарии
Планирование	По сезону, без симуляции будущих сценариев	Сценарное моделирование “что если”, построение индивидуальных траекторий
Прогноз результата	Интуитивный, опирающийся на личный опыт	Моделируется нейросетями по текущим данным спортсмена

Дистанционный и персонифицированный характер работы системы переводит профилактику травм и управление нагрузками из реактивного подхода к проактивному. Эффективность системы во многом зависит от качества получаемых данных, поэтому на этапе внедрения в процесс тренировок необходим контроль со стороны специалистов за системой, постепенно повышая доверие и вписывая её работу в существующий процесс.

Заключение

Внедрение систем поддержки принятия решений, основанных на компьютерном зрении и нейро-нечетком анализе, представляет собой перспективное направление, в том числе для развития детского спорта и спортивной науки в целом. Проведенное исследование показывает, что такая система способна решить ряд острых проблем, сопутствующих ранней спортивной специализации и интенсивным тренировкам детей. Научная значимость внедрения таких систем состоит в междисциплинарной работе: объединение знаний и методов педагогики, физиологии, информационных технологий, искусственного интеллекта для решения комплексной задачи.

Работа демонстрирует эффективность объединения современных методов искусственного интеллекта (CNN, LSTM, нейро-нечеткие сети) с классическими теориями спорта, что позволяет построить эффективный инструмент управления тренировочным процессом. Особенностью системы является динамическое обновление данных цифрового двойника, что позволяет повысить адаптивность разработанной системы к функциональным изменениям опорно-двигательного аппарата спортсмена, в том числе характерным возрастным и индивидуальным, учитывать текущее состояние и строить более точные траектории развития. Такой подход способствует развитию всех задействованных дисциплин, с помощью создания новых алгоритмов анализа биомеханических данных, улучшенные системы распознавания действий человека в прикладной среде.

Список литературы

- [Арсеньев и др., 2023] Арсеньев Д.Г., Мисник А.Е., Шалухова М.А. Интеллектуальная система поддержки принятия решений для управления процессом реабилитации пациентов после эндопротезирования суставов // Вестник СПбПУ. Информатика. Телекоммуникации. Управление. – 2023. – Т. 16, № 2. – С. 45-57.
- [Кривецкий и др., 2013] Кривецкий И.Ю., Попов Г.И. Возможности применения технологии нейро-нечетких сетей в некоторых видах спорта // Информатика и системы управления. – 2013. – № 4(38). – С. 80-87.
- [Терёхин и др., 2022] Терёхин А.Д., Ильялов О.Р., Степанов А.В. Система оценивания спортивных упражнений по нейросетевому анализу видеоряда // Прикладная математика и вопросы управления. – 2022. – № 1(29). – С. 75-86.
- [Baker et al., 2022] Baker J., Mosher A. [и др.] Revisiting early sport specialization: What's the problem? // Sports Health. – 2022. – 14(1). – P. 20-26.
- [Balyi, 2004] Balyi I., Hamilton A. Long-Term Athlete Development: Trainability in Childhood and Adolescence // Olympic Coach. – 2004. – 16(1). – P. 4-9.
- [Bergeron et al., 2015] Bergeron M.F. [и др.] The Youth Athlete: Do Training Patterns Trigger Growth Plate Injuries? // International Journal of Sports Physiology and Performance. – 2015. – 10(4). – P. 552-559.
- [Borisov et al., 2023a] Borisov V.V., Misnik A.E. Ontological Engineering of Interrelated Processes in Complex Cyber-Physical Systems // Proc. of the 6th Int. Scientific Conf. "Intelligent Information Technologies for Industry" (ITI'22) (Cham, 2022). Lecture Notes in Networks and Systems. Vol. 566. – Cham: Springer, 2023. – P. 432-443. – doi: 10.1007/978-3-031-19620-1_39.
- [Borisov et al., 2023b] Borisov V.V., Misnik A.E., Velkov A.A., Shalukhova M.A. Application of Computer Vision Technologies to Reduce Injuries in the Athletes' Training // Proc. of the 7th Int. Scientific Conf. "Intelligent Information Technologies for Industry" (ITI'23) (Cham, 2023). Lecture Notes in Networks and Systems. Vol. 777. – Cham: Springer, 2023. – P. 143-153. – doi: 10.1007/978-3-031-43792-2_14.
- [Hoang et al., 2021] Hoang V.T., Jo K.H. Practical analysis on architecture of EfficientNet // Proc. of IEEE HSI. – 2021. – P. 1-4.

- [Jang, 1993]** Jang J.-S.R. ANFIS: Adaptive-Network-Based Fuzzy Inference System // IEEE Trans. on Systems, Man, and Cybernetics. – 1993. – 23(3). – P. 665-685.
- [Kolt, 1999]** Kolt G.S., Kirkby R.J. Epidemiology of injury in elite and subelite female gymnasts: a comparison of retrospective and prospective findings // Br. J. Sports Med. – 1999. – 33(5). – P. 312-318.
- [Myer et al., 2016]** Myer G.D., Jayanthi N., DiFiori J.P. Sport specialization, Part I–II // Sports Health. – 2015–2016.
- [Osokin, 2018]** Osokin D. Real-time 2D Multi-Person Pose Estimation on CPU: Lightweight OpenPose // arXiv preprint. – 2018.
- [Popkin et al., 2019]** Popkin C.A., Bayomy A.F., Ahmad C.S. Early sport specialization // J. of the Am. Acad. of Orthop. Surgeons. – 2019. – 27(22): e995–e1000.
- [Wang et al., 2022]** Wang J. [et al.] NGCU: A new RNN model for time-series data prediction // Big Data Research. – 2022. – Vol. 27. – Art. 100296.

УДК 004.85

doi: 10.15622/rcai.2025.056

ПРИНЯТИЕ РЕШЕНИЙ О ПРОВЕДЕНИИ МАНЕВРА С ПРЕДСКАЗАНИЕМ ТРАЕКТОРНЫХ ПОТОКОВ ДЛЯ БЕСПИЛОТНОГО ТРАНСПОРТНОГО СРЕДСТВА

М. Жамал (*mayssjamal@phystech.edu*)^A

А.И. Панов (*panov@airi.net*)^{A,B,C}

^A Московский физико-технический институт, Долгопрудный

^B Федеральный исследовательский центр

«Информатика и управление» РАН, Москва

^C AIRI, Москва

Принятие решений, планирование движения и прогнозирование траекторий - ключевые компоненты систем автономного вождения. В данной статье представлен алгоритм FFStreams++, реализующий гибридную систему принятия решений и планирования движения, которая объединяет генерацию траектории на основе выборки с эвристическим поиском. Такие маневры, как незащищенные левые повороты, обгоны и удержание полосы движения, моделируются с помощью доменного языка Planning Domain Definition Language (PDDL) и генерируются с помощью планировщика Fast-Forward. Предложенная система итеративно уточняет начальное состояние с помощью потоков траекторий, специфичных для конкретного маневра, пока не будет найден оптимальный план. FFStreams++ включает в себя нейросетевой аппроксиматор для предсказания траекторий окружающих препятствий с соответствующими вероятностями. Оценка работоспособности предложенного алгоритма проведена на бенчмарке CommonRoad и демонстрирует эффективность планировщика в выполнении сложных маневров при удовлетворении требований по безопасности маневров.

Ключевые слова: Автономное вождение, планирование поведения, интегрированное планирование задач и движения, планирование маневра, прогнозирование траектории.

Введение

Автономные системы вождения используются в транспортных системах, позволяя автомобилям двигаться без участия человека. Эти системы опираются на ряд ключевых модулей – локализацию, восприятие, плани-

рование и управление, – которые работают совместно, чтобы обеспечить безопасную навигацию в динамичной среде [Teng et al., 2023], [Yurtsever et al., 2020]. Планирование, в частности, включает в себя генерацию безопасных и эффективных траекторий путем анализа данных датчиков, HD карт и прогнозируемых движений окружающих агентов [Jamal et al., 2024], [Gonzalez, 2019].

Точное предсказание траектории имеет решающее значение для предотвращения принятия небезопасных решений, особенно во время сложных маневров, таких как незащищенные левые повороты (рис. 1,а) и обгоны (рис. 1,б) на высоких скоростях. Традиционные модели планирования и предсказания траекторий других агентов эффективны, но с трудом справляются с неопределенностью и взаимодействиями, в то время как глубокое обучение и обучение с подкреплением (RL) обеспечивают более высокую точность и адаптивность, но сталкиваются с такими проблемами, как специфические требования к данным, низкая интерпретируемость и обобщаемость в реальном мире [Schwartz et al., 2018], [Dulac-Arnold et al., 2019].

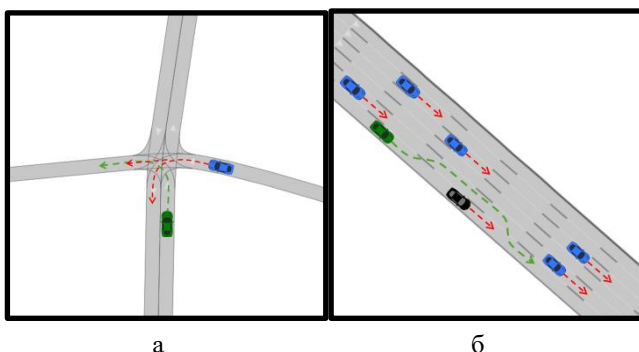


Рис. 1. а – сложный маневр незащищенного левого поворота на перекрестке без светофора, б – сложный маневр обгона на многополосном шоссе

Методы принятия решений для данной задачи можно разделить на классические (основанные на правилах, оптимизации, вероятностные) и основанные на обучении, каждый из которых по-разному реализует соотношение между интерпретируемостью, адаптивностью и производительностью в неопределенной среде [Liu et al., 2021]. Для устранения недостатков предыдущих работ в данной работе представлена система FFStreams++, объединяющая прогнозирование траектории, принятие решений и планирование движения для таких маневров, как незащищенные левые повороты, обгоны и удержание полосы движения.

FFStreams++ расширяет подход FFStreams [Jamal et al., 2024], интегрируя высокоточную модель прогнозирования -Query-Centric Network (QCNet) [Zhou et al., 2023] и представляя задачи планирования с помощью доменного языка PDDL, дополненного так называемым потоками (streams), специфичными для конкретных маневров [Garrett et al., 2020]. Предложенный подход использует эвристический планировщик Fast-Forward [Hoffmann, 2001] для поиска безопасных и эффективных траекторий путем оптимизации ограничений на рывок, ускорение и кривизну траекторию.

FFStreams++ был проверен на реальных сценариях вождения из бенчмарка CommonRoad [Althoff et al., 2017]. В данной работе представлена новая интеграция принятия решений и планирования движения с помощью потоков, оптимизированных для безопасного и комфортного маневрирования, включающая модель QCNet для точного предсказания траектории, и демонстрирующая улучшенную производительность планирования по сравнению с базовыми методами, основанными на эвристическом поиске.

1. Постановка задачи

Planning Domain Definition Language (PDDL) – это формальный язык для моделирования задач планирования автономных агентов, позволяющий определять домен задачи (действия, типы, предикаты и функции) и собственно саму задачу (начальное и целевое состояния). PDDL2.1 [Fox et al., 2003] расширяет этот язык, поддерживая числовые переменные потока (fluents) и метрики оптимизации.

Домен планирования $Dom = \langle T, P, F, A \rangle$ задается четверкой, где T – набор типов, используемых для классификации объектов в домене, P – набор предикатов, $P = \{p_1, p_2, \dots, p_p\}$, где каждый предикат p может быть применен к определенной последовательности объектов $\{o_1, o_2, \dots, o_o\}$ для формирования литералов, F – набор функций (переменных), $F = \{f_1, f_2, \dots, f_f\}$, где каждая функция f может быть применена к определенной последовательности объектов с присваиванием им значений, A – набор действий, $A = \{a_1, a_2, \dots, a_j\}$, где каждое действие a определяется кортежем аргументов объектов $\bar{o} = \langle o_1, o_2, \dots, o_r \rangle$ и набором предусловий $pre(a)$ для \bar{o} , которые задаются положительными литералами $pre^+(a)$ и отрицательными литералами $pre^-(a)$, которые должны выполняться для применения действия, и набором эффектов $eff(a)$ для \bar{o} , которые являются положительными литералами $eff^+(a)$ и отрицательными литералами $eff^-(a)$, которые являются результатом применения действия.

Действие a применимо в состоянии I , если

$$(pre^+(a(\bar{o})) \subseteq I) \wedge (pre^-(a(\bar{o})) \cap I = \emptyset). \quad (1.1)$$

Результирующее состояние I' после применения действия a в состоянии I задается как

$$I' = (I \setminus \text{eff}^-(a(\bar{o}))) \cup \text{eff}^+(a(\bar{o})). \quad (1.2)$$

Задача планирования $Prob = \langle O, I_0, G \rangle$ определяется тройкой, где O – множество объектов, являющихся экземплярами определенных типов, I_0 – начальное состояние, множество положительных литералов, выражающих начальное состояние, G – целевое состояние, множество как положительных, так и отрицательных литералов, выражающих целевое состояние. План $\pi = [a_1(\bar{o}_1), \dots, a_k(\bar{o}_k)]$ – это конечная последовательность из k экземпляров действий, таких, что каждое $a_i(\bar{o}_i)$ применимо в состоянии I_{i-1} , ведущему к состоянию I_i . Состояние цели G удовлетворяется после применения всей последовательности π .

Поток(stream) $s(\bar{o})$ – это условная функция входного набора аргументов объектов $\bar{o} = \langle o_1, o_2, \dots, o_m \rangle$. Эта функция может модифицировать задачу планирования $Prob$ генерируя на выходе кортеж новых объектов $\bar{r} = \langle r_1, r_2, \dots, r_l \rangle$, и набор сертифицированных фактов, связанных с ними, где $s.cert = \{p \mid \forall \bar{o} \in \bar{O}, \forall \bar{r} \in s(\bar{o}). p(\bar{o} + \bar{r})\}$. Поток может давать значение None, если генерация новых объектов невозможна. Поток может быть применен к входным параметрам \bar{o} только в том случае, если в домене существует множество связанных с ними положительных литералов p , где $s.dom = \{p \mid \forall \bar{o} \in \bar{O}. p(\bar{o})\}$. Поток может изменять начальное состояние I_0 в задаче $Prob$.

Когда потоки интегрируются в планировщик, генерирующий план на языке PDDL, процесс итеративно запускается на различных уровнях планирования. Вначале применимые потоки используются для изменения начального состояния. Затем планировщик PDDL выполняет эвристический поиск оптимального плана. Если план не найден, уровень планирования повышается, и процесс повторяется. Эта итерационная процедура продолжается до тех пор, пока либо не будет найден оптимальный план, либо не будет достигнут максимальный уровень планирования.

Планировщик FastForward (FF) [Hoffmann, 2001] решает задачи PDDL с помощью эвристического поиска, руководствуясь эвристикой h^{FF} , которая оценивает стоимость плана, игнорируя негативные эффекты. Такой подход позволяет эффективно исследовать пространство поиска для достижения состояния цели.

2. Метод

В данной работе предлагается фреймворк FFStreams++ для решения комплексной задачи принятия решений о совершении маневра и планирования движения (рис. 2). FFStreams++ включает в себя сеть Query-Centric Network (QCNNet) для предсказания траектории.

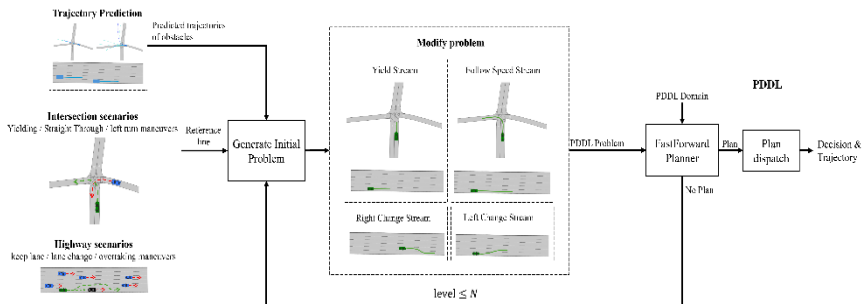


Рис. 2. Схема предлагаемого фреймворка FFStreams++ для интегрированного принятия решений и планирования движения с прогнозированием траектории при автономном вождении в динамических средах

2.1. Модель прогнозирования траектории QCNet

В данной работе используется предварительно обученная на бенчмарке Argoverse 2 сеть аппроксимации запроса (Query-Centric Network, QCNet), для предсказания траекторий движения окружающих автомобилей в сценариях CommonRoad. Модель принимает на вход данные карты и прошлые состояния препятствий (позиции и ориентации) и выдает шесть возможных будущих траекторий с соответствующими вероятностями. Для обеспечения совместимости картографические данные были преобразованы из формата Lanelet2 в формат Argoverse.

Для каждого препятствия две наиболее вероятные предсказанные траектории передаются планировщику FFStreams++. Горизонты предсказания и планирования были установлены на 5 секунд на основе оценок с использованием RMSE в сценариях для шоссе CommonRoad.

2.2. Система принятия решений и планирования движения FFStreams++

Фреймворк FFStreams++ расширяет оригинальную версию FFStreams [Jamal et al., 2024] для поддержки незащищенных левых поворотов, а также маневров удержания в полосе, смены полосы движения и обгона. Он включает в себя более точные прогнозируемые траектории окружающих препятствий, учитывающие их ускорения для повышения безопасности и генерации адаптивного поведения.

FFStreams++ представляет задачу планирования в PDDL2.1 и использует потоки для итеративной генерации оптимизированных траекторий в системе координат Френе. Эти потоки создают соответствующие предикаты, которые вставляются в задачу PDDL. Затем эвристический планировщик FastForward (FF) ищет оптимальную последовательность маневров.

2.2.1. Домен PDDL и FastForward nouck. Принятие решений и планирование движения формализуется как домен планирования PDDL2.1 (Dom) и постановка задачи планирования (Prob). Домен определяет предикаты, типы и действия, а задача – объекты, начальные и целевые предикаты, а также метрику качества решения. PDDL2.1 поддерживает численные операции, позволяя выполнять арифметические операции, такие как вычисление расстояния для проверки столкновений.

В домене определяется два типа объектов: конфигурации автомобилей (conf) и препятствия (obstacle). Функции в домене включают (*total_cost*), (*curr_time*) и числовые значения, связанные с конфигурациями и препятствиями.

Маневр определяется как действие:

$$A = \{keep_speed(\bar{o}), keep_yield_speed(\bar{o}), \\ left_change(\bar{o}), right_change(\bar{o}), overtake(\bar{o})\} \quad (2.1)$$

Каждое действие манипулирует сегментами траектории и обновляет предикаты для поддержки планирования и проверки столкновений.

После определения (*Dom*) и (*Prob*) планировщик FastForward (FF) выполняет эвристический поиск, используя эвристику h^{FF} . Эта эвристика строит свободный граф планирования (Relaxed Planning Graph, RPG), в котором игнорируются эффекты удаления – предполагается, что факты остаются истинными после их достижения, – что упрощает процесс планирования. Когда предикаты цели появляются в слое фактов, планировщик отслеживает необходимые действия, и совокупная стоимость действий дает эвристическую оценку. FF использует взвешенный поиск A^* , руководствуясь этой оценкой. Как построение RPG, так и эвристический расчет имеют полиномиальную временную сложность, что обеспечивает эффективное планирование.

2.2.2. Маневрирование с потоками и оптимизацией рывка. В данной работе используется отдельный поток для каждого маневра – скорость движения, действие «уступи дорогу», смена полосы движения и обгон. На каждой итерации FFStreams++ вызываются только соответствующие потоки, в зависимости от условий (например, поток обгона используется только при наличии препятствия впереди).

Для каждого маневра определяется желаемая конечная скорость $v_{desired}$ и генерируются кандидаты траекторий с помощью оптимизации координат Френе с минимизацией рывков для генерации комфортной траектории. Опорной траекторией является осевая линия полосы движения, а начальное состояние $[x_0, y_0, \theta_0, v_0, a_0]$ преобразуется в координаты Френе:

$$x(t) = [s(t), s'(t), s''(t), s'''(t), l(t), l'(t), l''(t), l'''(t)], \quad (2.2)$$

где s и l представляют собой продольное и поперечное движение и их производные. Используя квинтовые полиномы для бокового и квартовые для продольного движения, генерируются выполнимые траектории. Невыполнимые (нарушающие ограничения a_{max} , v_{max} или кривизну) отбрасываются.

Каждая траектория τ оценивается функцией стоимости:
 $J(\tau) = J_{comfort}(\tau) + J_{efficiency}(\tau) + J_{lateral_error}(\tau) + J_{speed_error}(\tau), \quad (2.3)$

где

$$J_{comfort} = \omega_j \int_0^T (s'''^2 + l'''^2) dt,$$

$$J_{efficiency} = \omega_t \cdot T,$$

$$J_{lateral_error} = \omega_{err} \cdot (l(T) - l_{desired})^2,$$

$$J_{speed_error} = \omega_{err} \cdot (s'(T) - s'_{desired})^2,$$

$l_{desired} = 0$, цель - следовать осевой линии. Оптимальная траектория τ^* минимизирует общую стоимость $J(\tau)$ (табл. 1).

Таблица 1

Параметр	Символ	Значение
Jerk weight	ω_j	0.1
Travel time weight	ω_t	0.1
Error weight	ω_{err}	1.0
Planning horizon	T	5.0 [s]
Time step	Δt	0.2 [s]
Maximum acceleration	a_{max}	2.0 [m/s^2]
Maximum speed	v_{max}	57.6 [m/s]
Maximum Curvature	κ	1 [$1/m$]

3. Эксперименты и результаты

Возможности алгоритма FFStreams++ по принятию решений и планированию движения в сложных сценариях, включая незащищенные левые повороты на перекрестках, были оценены экспериментально. На перекрестках автомобиль должен решить, проехать или уступить.

Был использован бенчмарк CommonRoad с картами *DEU_Nuremberg-39* (перекресток без светофора) и *USA_US101-22* (автомагистраль). Эффективность оценивалась по показателям безопасности (коэффициент успешного предотвращения столкновений) и комфорта с помощью метрики предпочтений водителя (Occupant's Preference Metric, OPM), основанной на оценке ускорения и рывка. Сравнение проводилось с планировщиком на основе поиска, использующим 2697 примитивов движения из SBPL.

В сценариях перекрестка препятствие имеет начальное случайное положение на полосе движения и движется с высокой скоростью, следуя профилю скорости и ускорения, взятому из реалистичного сценария CommonRoad "DEU_Nuremberg-39_5_T-1". Беспилотный автомобиль име-

ет начальное положение [360.51,-30.79] и движется со случайной начальной скоростью [3.5 m/s, 11.5 m/s]. После проведения 100 экспериментов (табл. 2) по двум типам сценариев (поворот налево и движение прямо) в 89.00% экспериментов по сценарию «движение прямо» беспилотный автомобиль успешно проехал прямо и миновал перекресток. В 84.00% экспериментов по сценарию «левый поворот» беспилотный автомобиль успешно выполнил незащищенный левый поворот. С другой стороны, планировщик, основанный на поиске, имел 75.00% успеха в экспериментах по сценарию «движение прямо» и только 53.00% в экспериментах по сценарию «левый поворот». Полученные результаты демонстрируют превосходство FFStreams++ над планировщиками, основанными на поиске. После оценки эффективности двух методов по метрике OPM траектории FFStreams++ были классифицированы как «нормальный водитель», а траектории на основе поиска - как «агрессивный водитель», что доказывает лучшую эффективность FFStreams++ и его близость к поведению реального водителя.

Таблица 2

Method	Go straight Success rate	Left turn Success rate	OPM
FFStreams++	89%	84%	Normal Driver
Search-based planner	75%	53%	Aggressive Driver

На рис. 3 показан успешный эксперимент с незащищённым левым поворотом. При приближении к перекрёстку FFStreams++ уступает, а после уточнения траекторий соседнего автомобиля – ускоряется и выполняет поворот. Профили скорости, ускорения и рывка соответствуют ограничениям и обеспечивают комфортную траекторию движения, тогда как базовый планировщик не смог построить безопасную траекторию.

На рис. 4 показан успешный проезд перекрёстка. FFStreams++ снижает скорость, а после уточнения траектории соседнего автомобиля ускоряется и проезжает перекрёсток. Профили движения соответствуют ограничениям и обеспечивают безопасный маневр. В отличие от него, базовый планировщик планирует действие остановки, что отклоняется от необходимого поведения водителя, тогда как траектория FFStreams++ не имеет рывков и близка к человеческому стилю вождения.

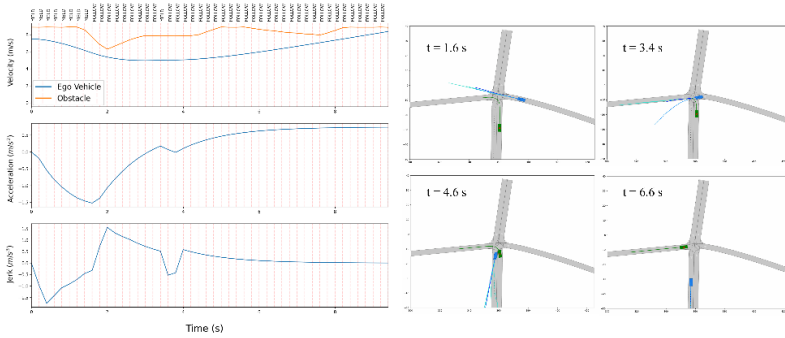


Рис. 3. Успешный эксперимент по планированию незащищенного левого поворота с помощью планировщика FFStreams++ и профили скорости, ускорения и рывка планируемых траекторий, а также решение на каждом временном шаге

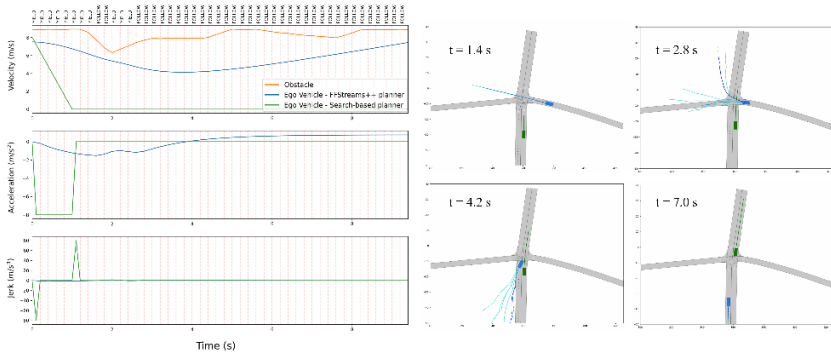


Рис. 4. Успешный эксперимент по прохождению перекрестка, запланированный планировщиком FFStreams++, и профиль скоростей, ускорений и рывков запланированных траекторий, а также решение на каждом временном шаге

Заключение

В работе представлен **FFStreams++**, фреймворк, объединяющий принятие решений и планирование движения для беспилотных автомобилей с помощью выборки траекторий для конкретного маневра и эвристического планировщика FastForward в PDDL. Благодаря включению модели предсказания QCNet она учитывает динамику препятствий и повышает точность планирования.

Эксперименты на бенчмарке CommonRoad показали, что FFStreams++ успешно справляется со сложными маневрами, такими как незащищенные левые повороты и обгоны, демонстрируя безопасность, адаптивность и человекоподобное поведение водителя в городских сценариях.

Список литературы

- [Teng et al., 2023] Teng S., Hu X., Deng P., Li B., Li Y., Ai Y., Yang D., Li L., Xuanyuan Z., Zhu F., et al. Motion planning for autonomous driving: The state of the art and future perspectives // IEEE Transactions on Intelligent Vehicles. – 2023.
- [Yurtsever et al., 2020] Yurtsever E., Lambert J., Carballo A., and Takeda K.. A survey of autonomous driving: Common practices and emerging technologies // IEEE access. – 2020. – 8. – P. 58443-58469.
- [Jamal et al., 2024] Jamal M., Panov A.. FFstreams: Fast search with streams for autonomous maneuver planning // IEEE Robotics and Automation Letters. – 2024. – 9(7). – P. 6752-6759. – doi:10.1109/LRA.2024.3412633.
- [Gonzalez, 2019] Gonzalez D.S. Towards human-like prediction and decision-making for automated vehicles in highway scenarios. – Universite Grenoble Alpes, 2019.
- [Schwartz et al., 2018] Schwartz W., Alonso-Mora J., and Rus D. Planning and decision-making for autonomous vehicles. Annual Review of Control // Robotics, and Autonomous Systems. – 2018. – 1(1). – P. 187-210.
- [Dulac-Arnold et al., 2019] Dulac-Arnold G., Mankowitz D., and Hester T. Challenges of real-world reinforcement learning // arXiv preprint arXiv:1904. – 2019. – 12901.
- [Liu et al., 2021] Liu Q., Li X., Yuan S., Li Z. Decision-Making Technology for Autonomous Vehicles: Learning-Based Methods, Applications and Future Outlook // IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 2021. – P. 30-37. – doi: 10.1109/ITSC48978.2021.9564580, 2021.
- [Zhou et al., 2023] Zhou Z., Wang J., Li Y. and Huang Y. Query-Centric Trajectory Prediction . Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 17863-17873, 2023.
- [Garrett et al., 2020] Garrett C.R., Lozano-Pérez T., and Kaelbling L.P. Pddlstream: Integrating symbolic planners and blackbox samplers via optimistic adaptive planning // In Proceedings of the international conference on automated planning and scheduling. – 2020. – Vol. 30. – P. 440-448.
- [Hoffmann, 2001] Hoffmann J. FF: The fast-forward planning system // AI magazine. – 2001. – 22(3). – P. 57-57.
- [Althoff et al., 2017] Althoff M., Koschi M., and Manzing S. Commonroad: Composable benchmarks for motion planning on roads // In 2017 IEEE Intelligent Vehicles Symposium (IV). – 2017. – P. 719-726. – doi: 10.1109/IVS.2017.7995802.
- [Fox et al., 2003] Fox M., Long D. PDDL2. 1: An extension to PDDL for expressing temporal planning domains // Journal of artificial intelligence research. – 2003. – Vol. 20. – P. 61-124.

ПРИМЕНЕНИЕ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ В СИСТЕМАХ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

А.А. Кулинич (*alexkul@rambler.ru*)

Институт проблем управления РАН, Москва

Система поддержки принятия решений представлена как сложная когнитивная система, включающая интеллект лица, принимающего решения в виде модели его ментального пространства и предобученную большую языковую модель. Языковая модель используется для объяснения решений, формально полученных в модели ментального пространства. Предложен метод оценки качества и полезности взаимодействия человека и большой языковой модели. Предложена модель группы респондентов для оценки качества объяснений языковой модели на задаваемые вопросы. Приводится пример оценки языковой модели группой гипотетических респондентов. На примере показан рост понимания сути проблемы лицом принимающим решение, благодаря объяснениям языковой модели. Предложенная модель системы поддержки принятия решений и метод оценивания качества взаимодействия человека и большой языковой модели могут быть полезны для предварительного оценивания таких систем.

Ключевые слова: поддержка принятия решений, большая языковая модель, объяснение альтернатив решений, оценка качества объяснения решения.

Введение

Для принятия решений в социальных, политических и экономических системах применяют «мягкий» системный анализ [Checkland, 1981]. «Мягкий» системный анализ основывается на принципе ограниченной рациональности [Simon, 1988], который утверждает, что в силу ограниченных когнитивных ресурсов лиц, создающих модель объекта, можно создать упрощенную системную модель с гипотетической структурой, а параметры модели могут иметь лингвистические значения. При построении модели в рамках «мягкого» системного анализа исходными данными являются неструктурированные данные – свободный текст. Модель опи-

сывает объект на ограниченном естественном языке, который предлагает эксперт. Результаты моделирования – альтернативы решений, в такой модели также представлены на ограниченном естественном языке и интерпретируются самим экспертом в терминах его знаний, которых может быть недостаточно для получения нового решения. При этом интерпретация должна включать знания о предметной области релевантные решению, но находящиеся за пределами ограниченного экспертного языка и упрощенной модели ситуации.

В психологии считается, что знания о предметной области представлены в интеллекте человека в психической среде, которая называется ментальным пространством [Fauconnier, 1994] и в котором происходит процесс принятия решения. Это рассуждения, обобщения, интерпретация возможных альтернатив решений. Поддержать процесс принятия решений в ментальном пространстве можно с помощью больших языковых моделей предобученных большим корпусом текста.

В этой работе исследована возможность применения больших языковых моделей в системах поддержки принятия решений в условиях неопределенности для интерпретации и объяснения альтернатив решений.

1. Большие языковые модели в поддержке принятия решений

Применение технологий искусственного интеллекта и машинного обучения в обработке естественного языка привело к появлению больших языковых моделей, популярность которых и области применения расширяются. Большие языковые модели способны генерировать текст, отвечать на сложные вопросы, решать логические задачи, и выполнять множество других задач с качеством сравнимым с качеством выполнения этих задач человеком. Они существенно влияют на многие аспекты нашей жизни: образование, творчество, принятие решений, управление и т.д.

Большие языковые модели (Large Language Model, LLM) представляют собой нейронные сети глубокого обучения, которые обучаются на гигантских объемах текстовых данных. В их основе лежит архитектура трансформера, которая включает набор нейронных сетей, состоящих из энкодера и декодера [Vaswani, 2017]. Энкодер преобразует входной текст в числовые векторные представления, сохраняя информацию о структуре и взаимосвязях слов. Декодер использует закодированную информацию для генерации ответа или выполнения предсказаний в виде нового оригинального текста, учитывая предыдущие слова и контекст.

Нейронная сеть большой языковой модели представляется в виде многослойной структуры, в которой каждый слой состоит из множества искусственных нейронов, связанных с нейронами соседних слоев. В результате обучения нейронной сети большим корпусом текста формируется словарь токенов – отдельных букв, слов, фраз, предложений для которых

определена вероятность их совместного употребления в определенном контексте. При кодировании входной информации и передачи ее от слоя к слою используется механизм внимания, позволяющий выявить важные элементы структуры языка. В результате обучения образуется векторное пространство токенов, в котором определена (косинусная) мера близости токенов. При декодировании, для ответа на вопрос из векторного пространства выбираются токены близкие к теме вопроса, из которых выстраиваются предложения – ответ на вопрос. Выбор токенов из векторного пространства близких к токенам вопроса позволяет сконструировать ответ на поставленный вопрос из слов, часто встречающийся в контексте вопроса.

Таким образом, большая языковая модель формирует статистический ответ на поставленный вопрос, не работая на уровне семантики, не анализируя смысл вопроса и ответа. Смысл ответа зависит от качества корпуса текста, которым обучалась модель. Если в корпус включались тексты из проверенных источников, то ответ будет верен и понятен.

Если языковая модель не понимает вопрос или она обучалась неполными или ошибочными данными, то модель пытается угадать ответ, опираясь на имеющиеся шаблоны, что может привести к ложным ответам – галлюцинациям. Галлюцинация - это феномен больших языковых моделей, когда она генерирует неверные или вымышленные данные, которые не основаны на реальной информации. Такое поведение свойственно и людям, когда в условиях дефицита информации или времени для изучения проблемы, они формируют случайные или вымышленные ответы и объяснения, далекие от реальности.

В настоящее время разработаны различные метрики оценки склонности языковых моделей к галлюцинациям и методы их уменьшения. Необходимо отметить, что галлюцинации языковой модели в случаях решения исследовательских или творческих задач могут стимулировать интуицию человека и привести к оригинальному решению. Напомним, что эвристические методы решения творческих задач, такие как мозговой штурм, синектика не отвергают абсурдные и контрпродуктивные альтернативы решений, считая их стимуляторами интуиции способной привести к новому и оригинальному решению.

Несмотря на этот недостаток языковых моделей, они находят применение благодаря их способностям решать задачи, представленные на естественном языке, которые ранее решал только человек.

Способности больших языковых моделей оценивают с помощью специальных программ - бенчмарков, которые измеряют их основные качества: объем знаний, точность ответов, надёжность и др. Выделяют бенчмарки: на проверку знаний; логического мышления; понимания прочитанного текста; здравого смысла и др. [Barskaya, 2024].

Бенчмарки на проверку знаний тестируют модели в различных областях. Он оценивает, насколько эффективно модель может вспоминать информацию из разных сфер, таких как физика, география и т.д.

Известный бенчмарк MMLU (Multimodal Language Understanding) создан для проверки уровня знаний модели по различным темам, таким как гуманитарные и социальные науки, история, компьютерные науки и даже право. Он включает 57 вопросов и 15 тысяч задач, направленных на то, чтобы убедиться в уровне знаний модели. На этом бенчмарке языковая модель GPT-4-omni правильно ответила на 88,7% заданных ей вопросов.

Бенчмарки на проверку логического мышления тестируют способности модели «думать» пошагово и делать логические выводы.

Используются бенчмарки оценки математических способностей языковых моделей. Например, тест GSM8K состоит из 8,5 тысяч задач по математике для средней школы. Решение этих задач требует выполнения моделью нескольких последовательных элементарных вычислений. Специально обученные языковые модели показывают хорошие результаты на этом бенчмарке, например, модели GPT-4 достигают 96,5%.

Бенчмарк вопросов и ответов (GPQA) уровня аспирантуры оценивает логическое мышление модели, используя набор данных из 448 вопросов. Это сложный тест, разработанный экспертами из областей биологии, физики и химии. Языковая модель GPT-4-omni проходит этот, набирая лишь 53,6%, в то время как аспиранты достигают точности всего 65%.

В 2024 году компания OpenAI выпустила языковую модель OpenAI o1 [Metz, 2024], которая демонстрирует превосходные результаты в сложных рассуждениях, превосходя людей в тестах по математике, кодированию и естественным наукам. На отборочных экзаменах Международной математической олимпиады (IMO) новая модель правильно решила 83% задач, тогда как ее предшественница GPT-4o дала лишь 13% верных ответов. Утверждается, что при решении сложных контрольных задач по физике, химии и биологии модель демонстрирует результаты, сопоставимые с результатами аспирантов.

Бенчмарки на понимание прочитанного текста тестируют способности модели интерпретировать естественный язык и генерировать ответы. Тестирование заключается в ответах на вопросы по текстам, что позволяет оценить понимание, способность делать выводы и удерживать детали.

Одним из бенчмарков для тестирования понимания прочитанного является тест DROP (Discrete Reasoning Over Paragraphs), который ставит перед моделями задачу выполнения рассуждений на основе анализа содержимого абзацев. Этот бенчмарк включает 96,000 вопросов для проверки способностей языковой модели к рассуждению. Вопросы DROP содержат информацию, которая требует от моделей выполнения математи-

ческих операций, на основе информации, разбросанной по всему тексту. Модели GPT-4 достигает точности 80%, в то время как люди достигают 96% правильных ответа на наборе тестов DROP.

Бенчмарки на проверку общих знаний (здорового смысла) оценивают способность модели к обобщённым знаниям о мире. Такие наборы тестов обычно включают вопросы, требующие для правильного ответа обширных энциклопедических знаний. Тестирование здравого смысла в языковых моделях оценивает способность модели делать суждения и выводы, соответствующие человеческому мышлению. Люди формируют целостное представление о мире через практический опыт, а языковые модели обучаются на огромных наборах данных, не понимая контекста.

Бенчмарк HellaSwag (Harder Endings, Longer Contexts, and Low-shot Activities for Situations With Adversarial Generations) предназначен для проверки способности модели предсказывать правдоподобное продолжение заданного сценария. Тестирование методами HellaSwag показывают, что современные модели, такие как GPT-4, достигли уровня производительности, близкого к человеческой точности.

Тестирование способностей языковых моделей с помощью бенчмарков позволяет объективно оценить и сравнить качество моделей разных производителей, понять текущий уровень и динамику развития языковых моделей. Как показал анализ тестирования способностей языковых моделей на тестах бенчмарков, эти способности приближаются к способностям человека. Это означает, что большая языковая модель может быть ассистентом в системах поддержки принятия решений.

Однако решения практических задач, например, по принятию решений в сложных экономических, политических или социальных ситуациях требует от ЛПР сложного сочетания разных способностей человеческого интеллекта. Поэтому, автоматизированные метрики сами по себе не могут охватить весь спектр оценки языковой модели, особенно когда дело касается субъективных аспектов понимания языка и генерации текста. Здесь человеческая оценка является гораздо более точной.

В этом случае целесообразно для оценки привлекать экспертов, способных дать точную и надежную оценку способностей больших языковых моделей как ассистентов в поддержке принятия решений.

2. Семиотическая модель ментального пространства и генерация альтернатив решений

В работе [Kulinich, 2023a] были рассмотрены вопросы генерации и интерпретации решений в условиях ограниченной рациональности с помощью семиотической модели ментального пространства. Семиотическая модель ментального пространства – это модель знаний эксперта о ситуации принятия решений, которая включает три взаимодействующие модели: синтаксическая, семантическая и прагматическая модели. В синтаксиче-

ской модели на ограниченном естественном языке эксперт описывает ситуацию принятия решений, используя имена всех объектов и их параметров, множество их возможных лингвистических значений, причинно-следственные отношения на множествах значений. Причинно-следственные отношения задают модели рассуждений эксперта в виде причинной сети. Рассуждения эксперта заключается в изменении им значений параметров, которые по причинным связям передаются другим параметрам ситуации принятия решений. В результате изменяются все связанные параметры ситуации. Новый вектор значений параметров, полученный в результате рассуждения эксперта, передается в семантическую модель. Семантическая модель представляется в виде качественного концептуального каркаса. Это качественная идеализированная онтология предметной области, в которой задана начальная ситуация, ее имя, параметры, их значения и все возможные обобщения начальной ситуации по разным параметрам. Начальная ситуация и обобщенные ситуации связаны родовидовыми отношениями (класс-подкласс) образуя решетку понятий предметной области. Рассуждая в синтаксической системе, эксперт может перейти к обобщенным ситуациям, что приводит к появлению нового класса решений и новым альтернативам.

В семантической модели мы получаем решения в виде составных имен классов решений, которые выражены на ограниченном естественном языке автора этой модели. Для представления решения в альтернативных языковых формах решения интерпретируются и объясняются с помощью больших языковых моделей языка.

3. Интерпретации и оценка объяснения языковыми моделями альтернатив решений

В работе [Kulinich, 2023b] было предложено интерпретировать и объяснять составные имена классов решений с помощью статических и динамических векторных моделей языка, представленных искусственной нейронной сетью. В работе [Hoffman, 2018] приводятся системные характеристики системы объяснения.

Во-первых, система объяснения – это сложная когнитивная система, включающая интеллект человека и обученную языковую модель. Со стороны человека в формировании объяснений задействовано его ментальное пространство. Оно отвечает за понимание ответов сгенерированных языковой моделью. Если объяснение языковой модели понятно человеку, то оно увеличивает его ментальное пространство.

Во-вторых, система объяснения – это диалоговая система. Если языковая модель отвечает на вопрос, то у человека могут возникнуть дополнительные вопросы. В этом случае хорошим вариантом дополнительных вопросов является метод «5 почему?» [Sarrat, 2017]. Он позволяет исследовать причинно-следственные цепочки проблемы.

В третьих, система объяснения – это исследовательская система. Она должна поддерживать любопытство человека, заставляя его разобраться в проблеме и найти нужное решение.

В работе [Hoffman, 2018] предложен ряд критериев для оценки качества систем объяснения. Это критерии: Полезность объяснения; Удовлетворенность объяснением; Понятность объяснения; Способность объяснения стимулировать любопытство; Доверие к объяснению.

Перечисленные критерии можно измерить. В работе [Hoffman, 2018] предложено для определения значений критериев в этой когнитивной системе, включающей интеллект человека (его ментальное пространство) и обученную нейронную сеть, использовать метод тестирования группой респондентов объяснений языковой модели по каждому из перечисленных критериев. Задача тестирования заключается в измерении латентной переменной – это удовлетворенность респондентов качеством объяснения языковой моделью альтернатив решений и контекста их применения.

Серьезной проблемой такого подхода является формирование группы респондентов. Формирование группы респондентов для измерения латентной переменной отношения человека к объяснению языковой модели считается искусством. В этой работе для предварительной оценки латентной переменной предложена модель группы респондентов.

4. Модель группы респондентов

Для тестирования каждого критерия используется семизначная шкала Лайкерта [Likert, 1932]. Пример шкалы Лайкерта приведен в табл. 1.

Модель респондента это пара - (E, σ) , где E – оценка респондента, $E \in \{1, 2, 3, 4, 5, 6, 7\}$, σ - среднее отклонение оценки эксперта.

Будем различать три типа респондентов: (Эксперт (E, σ_1) , Эрудит (E, σ_2) , Новичок (E, σ_3)). Эксперт всегда дает оценку с малым средним отклонением. Экспертная оценка знатока, характеризуется большим средним отклонением, и наконец, новичок имеет самое большое среднее отклонение, т.е. $\sigma_1 < \sigma_2 < \sigma_3$.

Группа респондентов формируется из респондентов разных типов. Число респондентов разных типов является параметром модели и может варьироваться в широком диапазоне. Далее считаем, что известна оценка одного эксперта объяснения языковой модели на вопрос «почему?» по всем пяти критериям, $E = (e_1, e_2, e_3, e_4, e_5)$, где $e_i \in \{1, 2, 3, 4, 5, 6, 7\}$. Считаем, что в группе N_1 экспертов, N_2 знатоков и N_3 новичков.

Для каждого эксперта сформируем случайную оценку $E_{ex} = (e_1 \pm \sigma_1, e_2 \pm \sigma_1, e_3 \pm \sigma_1, e_4 \pm \sigma_1, e_5 \pm \sigma_1)$, $ex = 1, \dots, N_1$. Аналогично, сформируем оценки для знатоков, $E_{co} = (e_1 \pm \sigma_2, e_2 \pm \sigma_2, e_3 \pm \sigma_2, e_4 \pm \sigma_2, e_5 \pm \sigma_2)$, $co = 1, \dots, N_2$ и новичков, $E_{be} = (e_1 \pm \sigma_3, e_2 \pm \sigma_3, e_3 \pm \sigma_3, e_4 \pm \sigma_3, e_5 \pm \sigma_3)$, $be = 1, \dots, N_3$. Число из диапа-

зонов отклонений $\pm\sigma_1, \pm\sigma_2, \pm\sigma_3$ – формирует генератор случайных чисел для каждого респондента. Таким образом, формируется случайная анкета, содержащая оценки $N_1 + N_2 + N_3$ респондентов.

Таблица 1

Шкала Лайкерта

Полез- ность	Удовлетво- ренность	Понятность	Любопыт- ство	Доверие	Оцен- ка
Очень полезно	Очень доволен	Абсолютно понятно	Обязатель- но возни- кает	Абсолютно доверяю	7
Полезно	Доволен	Понятно	Возникает	Доверяю	6
Мало полезно	Мало дово- лен	Мало понятно	Мало возникает	Мало доверяю	5
Затрудня- юсь отве- тить	Затрудняюсь ответить	Затрудняюсь ответить	Затрудня- юсь отве- тить	Затрудня- юсь отве- тить	4
Скорее бесполезно	Скорее недо- волен	Скорее не понятно	Скорее не возникает	Скорее не доверяю	3
Бесполезно	Недоволен	Не понятно	Не возни- кает	Не доверяю	2
Совершен- но беспо- лезно	Совершенно недоволен	Совершенно непонятно	Совершен- но не воз- никает	Совершен- но не дове- ряю	1

Однако, такая случайная анкета не может быть использована для оценки латентной переменной, характеризующей отношение лица, принимающего решение к объяснениям языковой модели.

Для проверки согласованности случайной анкета воспользуемся показателем α -Кронбаха [Cronbach, 1951]:

$$\alpha = \frac{k}{k-1} \left(1 - \frac{\sum_{i=1}^k \sigma_i^2}{\sigma_x^2} \right) \quad (1)$$

где k – число вопросов респондентам, σ_i^2 – среднее квадратичное отклонение оценок респондента по каждому вопросу, σ_x^2 – среднее квадратичное отклонение суммы оценок респондентов.

Если показатель $\alpha > 0.7$, то считается, что случайная анкета согласована. Если анкета не согласована, то осуществляется новая генерация случайной анкеты до получения согласованной анкеты.

Для согласованных анкет рассчитываем коэффициент ранговой корреляции Спирмена по формуле 2:

$$r = 1 - 6 \frac{\sum_{i=1}^n d_i^2}{n^3 - n} \quad (2)$$

где n – число респондентов, d_i^2 – квадрат разности рангов оценок респондентов по каждому вопросу и рангов суммы этих оценок.

Ранговая корреляция рассчитывается для каждого критерия, т.е. получаем вектор корреляций $R=(r_1, r_2, r_3, r_4, r_5)$. Для каждого коэффициента корреляции определим его статистическую значимость. По следующей формуле рассчитываем наблюдаемое значение t -критерия:

$$t_{obs} = \frac{r\sqrt{n-2}}{\sqrt{1-r^2}}, \quad (3)$$

где n – число респондентов, r – значение корреляции, $r \in R$.

По таблице критических значений Стьюдента для значения n определяем критическое значение t_{cr} . Если наблюдаемый t -критерий больше критического, $t_{obs} > t_{cr}$, то корреляция значима, т.е. гипотезу о существовании корреляции можно принять. Статистическая значимость коэффициента корреляции по некоторому вопросу позволяет говорить о нормальности распределения ответов респондентов по этому вопросу. Это повышает доверие к случайно сформированной анкете, основанной на одной экспертной оценке.

Такой подход может быть полезен для качественной оценки различных больших языковых моделей в системах принятия решений.

5. Пример

Большой языковой модели «`gpt_medium_turbo_instruct`» [Huggingface, 2025] задавался ряд вопросов по методу «5 почему?»

Первый вопрос к языковой модели такой: *Почему у бизнесменов возникают финансовые проблемы?*

Ответ языковой модели: *Финансовые проблемы могут возникнуть в различных ситуациях, включая отсутствие прибыли и роста бизнеса. Некоторые из наиболее распространенных причин - это потеря денег на инвестициях или кредитах, ухудшение финансового положения компании, снижение эффективности работы сотрудников, изменение законодательства о налогах и сборах, а также другие факторы.*

Второй вопрос: *Почему у бизнесменов нет прибыли и роста бизнеса?*

Ответ языковой модели: *Бизнесмены не могут получать прибыль в связи с отсутствием доходов. Причина - отсутствие прибылей, которые позволили бы им развиваться дальше.*

Третий вопрос: *В каких условиях должны работать бизнесмены, чтобы были доходы?*

Ответ языковой модели: *Бизнесмен должен иметь разные источники дохода – от продажи товаров и услуг до управления бизнесом в целом. Однако для того, чтобы стать успешным бизнесменом, необходимо уметь управлять своими финансами, инвестировать деньги в акции или облигации, а также работать эффективно.*

Для эксперта была разработана анкета, в которой предлагалось последовательно оценить все три вопроса и ответы языковой модели на них. Вопросы были следующие:

- Оцените полезность этого объяснения?
- Вы удовлетворены следующим объяснением?
- Вам понятно это объяснение?
- Возникают при таком объяснении дополнительные вопросы?
- Вы доверяете такому объяснению?

Вопросы предназначены для экспертной оценки каждого ответа языковой модели по критериям полезности, удовлетворённости, понятности, способности стимулировать любопытство и доверию к объяснению. Оценки по всем вопросам и объяснениям языковой модели с использованием шкалы Лайкерта приведены в табл. 2.

Таблица 2

Оценки Эксперта

	Полезность	Удовлетворенность	Понятность	Любопытство	Доверие
Вопрос 1	Полезно (6)	Доволен (6)	Понятно (6)	Скорее не возникает (3)	Доверяю (6)
Вопрос 2	Полезно (6)	Мало доволен (5)	Понятно (6)	Обязательно возникает (7)	Доверяю (6)
Вопрос 3	Малополезно (5)	Мало доволен (5)	Малопонятно (5)	Обязательно возникает (7)	Мало доверяю (5)

Векторы оценок эксперта, следующие по: вопросу 1 –(6, 6, 6, 3, 6); вопросу 2 –(6, 5, 6, 7, 6); вопросу 3 –(5, 5, 5, 7, 5).

Формировалась случайная анкета из ответов респондентов, для которой рассчитывался коэффициент α -Кронбаха (1). Многочисленные генерации случайных анкет позволили выделить по 12 анкет для каждого вопроса с удовлетворительным значением коэффициента согласованности.

Коэффициенты ранговой корреляции рассчитывались по формуле (2). Значения усредненных коэффициентов ранговой корреляции приведены в табл. 3.

Таблица 3

Ранговая корреляция для критериев оценивания

	Вопрос 1	Вопрос 2	Вопрос 3
Полезность	0,68	0,69	0,75
Удовлетворенность	0,66	0,65	0,78
Понятность	0,48	0,57	0,63
Любопытство	0,68	0,57	0,59
Доверие	0,67	0,72	0,73

Наблюдаемые значения t-критерия рассчитывались по формуле (3). Критические значения t-критерия выбирались из таблицы критических значений Стьюдента для значений $\beta=0.05$.

В результате проверки для $\beta=0.05$ коэффициенты корреляции по всем критериям оказались статистически значимы. Это означает, что вероятность того, что эти коэффициенты корреляции случайные составляет всего 5 процентов.

Анализ динамики изменения коэффициентов корреляции по разным критериям по таблице 3 показывает, что при последовательной оценке вопросов по методике «5 почему?» происходит рост значений коэффициентов по всем критериям, кроме критерия «Любопытство». Это может свидетельствовать о росте ментального пространства респондента. То есть по мере того как он получает ответы языковой модели, он лучше понимает проблему и меньше задает дополнительных вопросов.

Заключение

Рассмотрена модель сложной когнитивной системы, включающей интеллект лица, принимающего решения в виде модели его ментального пространства и большой языковой моделью. Большая языковая модель используется для интерпретации и объяснения решений, формально полученных в модели ментального пространства.

Предложен метод оценки качества и полезности взаимодействия человека и большой языковой модели. Рассмотрена модель гипотетической группы респондентов для оценки качества объяснений языковой модели на задаваемые вопросы. Были выполнены эксперименты с группой гипотетических респондентов, которые показали рост понимания сути проблемы лицом, принимающим решение благодаря объяснениям языковой модели.

Предложенная модель системы поддержки принятия решений, включающая большие языковые модели и метод оценивания качества взаимодействия человека и большой языковой модели может быть полезен для предварительного оценивания таких систем.

Список литературы

- [Barskaya, 2024] Barskaya I. Benchmarks For LLMs. Unite.AI, Aug. 28, 2024. – <https://www.unite.ai/benchmarks-for-llms/>.
- [Checkland, 1981] Checkland P.B. Systems Thinking, Systems Practice. – New York: Wiley, 1981. – 330 p.
- [Cronbach, 1951] Cronbach L.J. Coefficient alpha and the internal structure of tests // Psychometrika. – Springer Science and Business Media LLC, 1951. – 16(3). – P. 297-334. – doi:10.1007/bf02310555. – hdl:10983/2196. S2CID 13820448.
- [Fauconnier, 1994] Fauconnier G. Mental Spaces: Aspects of meaning construction in natural language. – Cambridge: Cambridge University Press, 1994.
- [Hoffman, 2018] Hoffman R., Mueller S., Klein G., Litman J. Metrics for explainable AI: Challenges and prospects // arXiv:1812.04608. – 2018.
- [Huggingface, 2025] <https://huggingface.co/models/>.
- [Kulinich, 2023a] Kulinich A. A. Semiotic Model of Mental Space // Pattern Recognit. Image Anal. – 2023. – Vol. 33. – P. 373-382. – <https://doi.org/10.1134/S1054661823030239>.
- [Kulinich, 2023b] Kulinich A.A. A Model to Support Intuitive Decision-Making // Proceedings of the 16th International Conference Management of Large-Scale System Development (MLSD). Moscow, Russian Federation: IEEE Xplore, 2023. – P. 1-5. – <https://ieeexplore.ieee.org/document/10304012>.
- [Likert, 1932] Likert R. A technique for the measurement of attitudes // Arch. Psychol. – 1932. – Vol. 7, No. 140.
- [Metz, 2024] Metz C. OpenAI Unveils New ChatGPT That Can Reason Through Math and Science. The New York Times (12 sept. 2024) (дата обращения: 28-05-2025).
- [Sarrat, 2017] Sarrat, O. The Five Whys Technique. In. Knowledge Solutions. – Springer, Singapore, 2017. – https://doi.org/10.1007/978-981-100983-9_32.
- [Simon, 1988] Simon HA. Rationality as Process and as Product of Thought / In: Bell DE, Raiffa H, Tversky A, eds. Decision Making: Descriptive, Normative, and Prescriptive Interactions. – Cambridge University Press, 1988. – P. 58-77.
- [Vaswani, 2017] Vaswani A., Shazeer N., et. all. Attention is all you need // arXiv pre-print arXiv:1706.03762, 2017. – <https://doi.org/10.48550/arXiv.1706.03762>.

СИНТЕЗ АДАПТИВНОГО УПРАВЛЕНИЯ В МНОГОКАНАЛЬНЫХ НАВИГАЦИОННЫХ ПРИЕМНИКАХ НА ОСНОВЕ ИНТЕЛЛЕКТУАЛЬНОЙ ОБРАБОТКИ ДОКОРРЕЛЯЦИОННЫХ СИГНАЛОВ

М.А. Лихачев (*likhachevma@cps.tver.ru*)

АО НИИ «Центрпрограммсистем», Тверь

В статье представлен двухэтапный синтез адаптивных управляющих решений для весовых коэффициентов диаграммы направленности цифровой антенной решетки (ЦАР) навигационного приемника в условиях преднамеренных шумовых и имитационных помех, заключающийся в интеллектуальном пространственно-временном анализе принятых сигналов спутниковой навигации. Предложенный подход направлен на повышение помехоустойчивости и информационной безопасности навигации за счет шумоподавления в суммарном докорреляционном сигнале. На первом этапе синтеза формируются веса, обеспечивающие глубокие провалы диаграммы направленности в направлениях источников активных помех, мощность которых превышает уровень шума. На втором этапе формируется новый набор весовых коэффициентов на основе идентификации события присутствия сигналов спуфинга и оценки угловых направлений их излучения. Прогнозирование координат источника спуфинговых сигналов (азимут и угол места) осуществляется с помощью LSTM-нейронной сети, обученной на данных циклической корреляционной матрицы, характеризующих циклостационарные свойства аутентичных навигационных сигналов. Синтезируемые интеллектуальные решения по управлению набором адаптивных весовых коэффициентов приемных каналов, позволяют минимизировать влияние комбинированных помех обеспечивая целостность и достоверность принимаемых навигационных сигналов для дальнейшей их корреляционной обработки и трекинга.

Ключевые слова: адаптивное управление, интеллектуальная обработка сигналов, глобальные системы спутниковой навигации (ГССН), цифровая антенная решётка (ЦАР), комбинированные помехи радиоглушения и спуфинга, спуфинг, LSTM-нейронная сеть.

Большое расстояние от навигационных спутников до аппаратуры потребителей (навигационных приемников) приводит к тому, что спектральная плотность мощности полезных сигналов на входе антенны становится меньше, чем мощность шума в антенне. Уровень информационных сигналов на 20...30 дБ меньше, чем уровень белого шума в антенных элементах, а мощные помехи радиоглушения превышают уровень шума на 20...40 дБ [Вейцель и др., 2014]. В случае приема навигационных сигналов на фоне мощной электромагнитной помехи для его выделения требуется обеспечить необходимое для детектирования отношение «сигнал/помеха плюс шум» путем пространственной фильтрации комплексного сигнала с использованием антенных решеток (АР) [Монзинго и др., 1986], [Перов, 2010], [Петров, 2015].

На первом этапе адаптивного двухступенчатого подхода интеллектуальной обработки выполняется подавление мощных помех радиоглушения с неизвестных направлений. Основным уравнением формирования выхода ступени подавления y с параметрами весового вектора $w(\theta, \varphi)$ по углу места θ и углу азимута φ , в направлении которых установлен максимум исходной ДН АР, обратной ковариационной матрицы R^{-1} и вектора принятого сигнала x является известная формула (1.1):

$$y = w^H R^{-1} x. \quad (1.1)$$

Процесс пространственной фильтрации целесообразно выполнить последовательно. Сигнальный вектор вначале умножается на обратную ковариационную матрицу для минимизации мощности помех радиоглушения, далее многоканальная весовая обработка в соответствии с выражением (1.2):

$$y_1 = R^{-1} x \rightarrow y_2 = w^H y_1. \quad (1.2)$$

Таким образом, для каждого компонента сигнального вектора x_n оценивается помеховая составляющая \hat{x}_n на основе других x_m ($m \neq n$). Оценка $y_n = x_n - \hat{x}_n$ ($n = 1 \dots N$) является оптимальным условием подавления помех. Ковариационная матрица рассчитывается на основе оценки канальной выборки отсчетов ограниченного объема \hat{R} .

Ограниченный объем данных для формирования ковариационной матрицы требует внесения некоторых технологических приемов для повышения точности оценки и исключения флуктуаций. Наиболее простым и эффективным методом является диагональное подмешивание некоторой шумовой матрицы αI , которая позволяет уменьшить относительные влияния малых собственных значений ковариационной матрицы, приведенным в выражении (1.3):

$$\hat{R} = \frac{1}{N} XX^H + \alpha I. \quad (1.3)$$

Применение методов пространственной адаптации позволяет обеспечить подавление помех радиоглушения в каждой компоненте сигнального вектора и оптимизировать ошибки измерений собственных координат в помеховой обстановке. Алгоритм первого этапа адаптивного двухступенчатого подхода изображен на рис. 1.

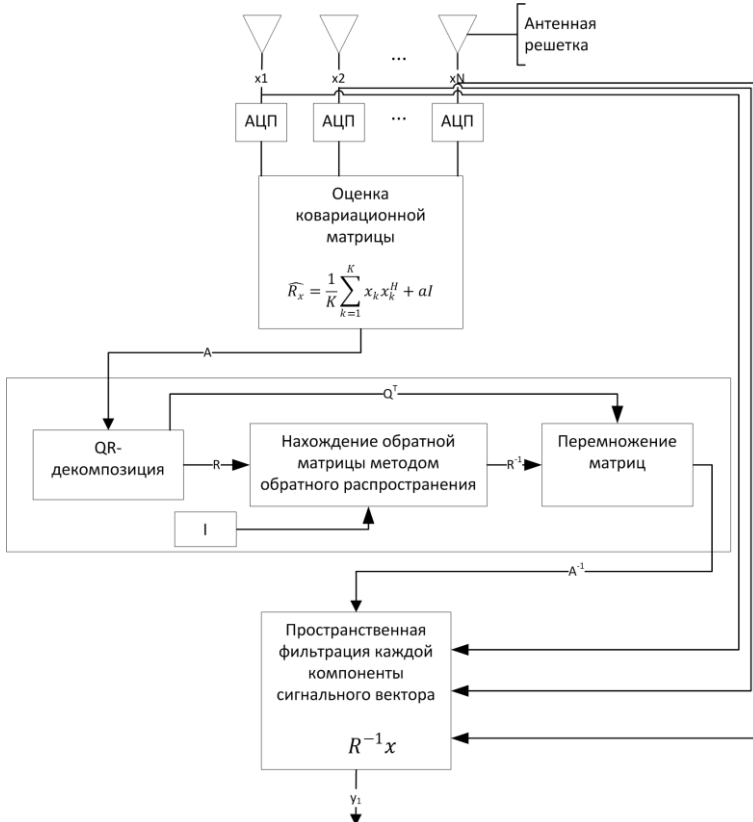


Рис. 1. Алгоритм первого этапа пространственной фильтрации сигналов радиоглушения

В отличие от методов радиоглушения, метод спуфинга преследует цель подмены истинного сигнала ложным с сообщением ложного местоположения или времени путем симуляции генератором сигналов, либо ретрансляцией реального сигнала ГССН с внесенными изменениями.

Сигнал спуфинга имеет схожие временные и спектральные характеристики с подлинными сигналами ГССН, поэтому приемник может захватывать поддельный сигнал после потери истинного или при энергетическом преимуществе ложного сигнала в корреляционном канале. Наиболее эффективным способом защиты от спуфинга является отслеживание зашифрованного сигнала (например, сигнала Y-кода на GPS L1 и L2), который транслируется несколькими созвездиями ГССН, а также методы посткорреляционной обработки, позволяющие отслеживать динамическое поведение корреляционных пиков в следящих каналах коррелятора и искажений навигационных сообщений. Однако, реализация указанных методов затруднена, поскольку требует с одной стороны доступа к кодирующей информации, а с другой сложного анализа деструкции формы и положения пиков в каждом канале коррелятора приемника совместно с историей содержания навигационных сообщений.

Поэтому наиболее рациональным решением в условиях воздействия спуфинг сигналов является недопущение их проникновения в каналы коррелятора.

Второй этап синтеза адаптивного управления в навигационном приемнике следует данному положению. Формирование управляющих воздействий на сигналы (канальные весовые коэффициенты) опирается на понятие циклостационарности, присущее и навигационным сигналам (в том числе сигналам спуфинга).

Сигнал считается циклостационарным, если его циклическая автокорреляционная функция не нулевая с шагом циклостационарности $\tau = lT_{C/A}$ ($l = 1, 2, 3, \dots$) и определяется следующим образом:

$$R_{ss}^{cc}(\tau) = E\{s(t)s^H(t - \tau)\}.$$

Например, сигнал L1 C/A излучается спутниками системы GPS с использованием периодического воспроизведение определенной модуляционной последовательности (кодов Голда) длиной $N = 1023$ чипов, уникальной для каждого спутника, с периодом $T = 1$ мс, который образуется суммированием по модулю 2 двоичных символов, поступающих с тактовой частотой $f = 1,023$ МГц от двух сдвиговых регистров. Такой сигнал удовлетворяет условиям циклостационарности на периоде $T_{C/A} = NT$.

Поскольку сигналы спуфинга обычно ниже уровня шума то они присутствуют на выходе первого этапа адаптивной системы совместно с сигналами ГССН. Чтобы осуществить пространственную фильтрацию спуфинга необходимо вместо пространственной ковариационной матрицы в первую очередь рассчитать циклическую ковариационную матрицу, которая определяется как кросс-ковариация принятого вектора сигнала и его задержанной на период цикла версии следующим образом:

$$\widehat{R}_{y_1}^c = \frac{1}{K} \sum_{k=1}^K y_{1k} (y_{1k}^D)^H + \alpha I.$$

Это прямой способ оценки циклической ковариационной матрицы наиболее популярный в литературе [Dong et al., 2017]. Однако этот способ дает плохую оценку при применении к реальной системе. Это связано с тем, что выборки данных, используемые для оценки циклической ковариационной матрицы, выбираются случайным образом, а длина блока данных ограничена. Возьмем в качестве примера, один из спутниковых (спуфинговых) сигналов GPS L1 C/A структура которого показана на рис. 2. На рисунке отмечено несколько пар блоков данных, и отмечено, что блок данных G разделен (Bit Transition) между двумя соседними навигационными символами с противоположными знаками.

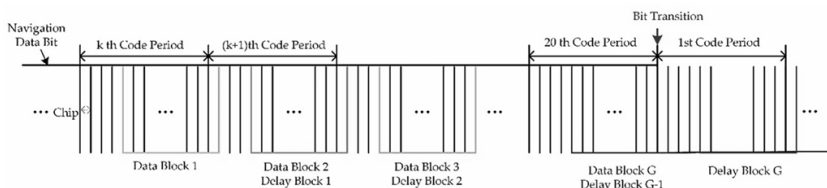


Рис. 2. Структура сигналов GPS L1 C/A

Проблема наилучшей оценки циклической матрицы решается путем использования нескольких блоков данных для получения гораздо большего количества ковариационных матриц. Как показано на рис. 2, $G(1 \leq G < 20)$ блоки данных выбираются, и усредненная циклическая ковариационная матрица может быть выражена следующим образом:

$$\widehat{R}_{y_1}^c = \frac{1}{G} \sum_{g=1}^G \frac{1}{K} \sum_{k=1}^K y_{1_k}^g (y_{1_k}^{gD})^H + aI. \quad (1.4)$$

Алгоритм второго этапа синтеза адаптивного управления по формированию весов диаграммы направленности и место интеллектуального блока анализа присутствия в принятом сигнале спуфинга и прогнозирования направления его излучения изображены на рис. 3.

Из рисунка следует, что предлагается комплексный подход, основанный на адаптивных системах с идентификацией (АСИ) [АСИ, 1980], сочетающий использование LSTM-нейронной сети [Thanh Han Trong et al., 2022] для точной оценки направлений прихода сигнала (DOA) спуфинга и метод sidelobe canceller для формирования адаптивной диаграммы направленности. Этот подход обеспечивает эффективное подавление помех и сохранение чувствительности к полезному сигналу, что повышает надёжность работы навигационного приёмника в условиях сложной помеховой обстановки.

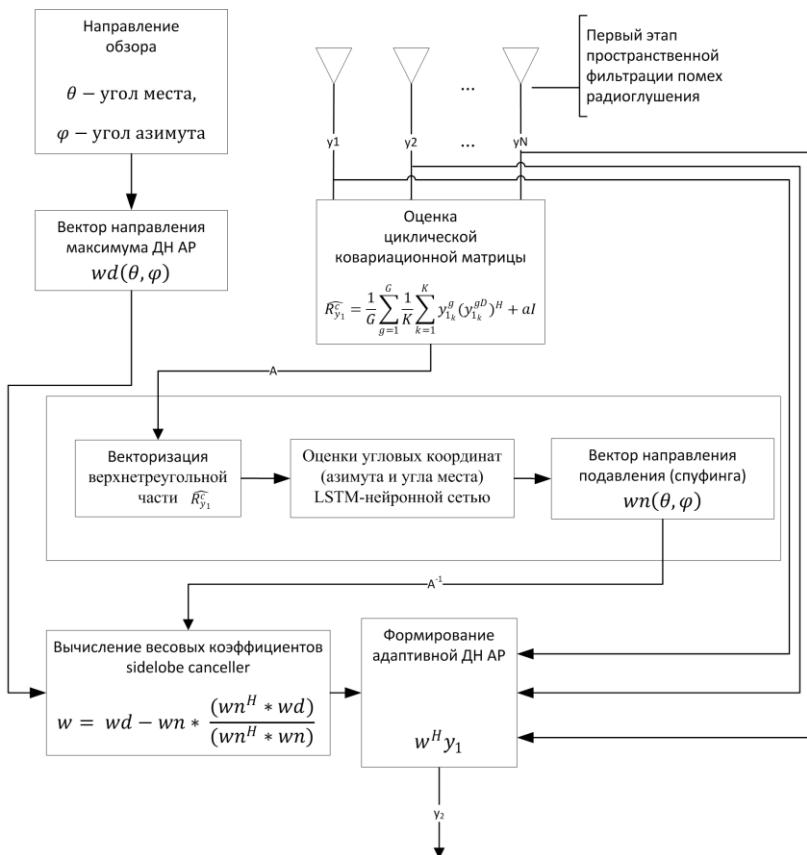


Рис. 3. Алгоритм второго этапа пространственной фильтрации сигналов спуфинга

LSTM-нейронная сеть используется для принятия решения о наличии спуфинга и прогнозирования угловых координат (азимута и угла места) при подаче на входной слой значений оценки циклической корреляционной матрицы.

Известно, что принятый сигнал m -го элемента прямоугольной плоской антенной решетки, состоящей из $M = N_x \times N_y$ элементов, расположенных на расстоянии d_x и d_y друг от друга по горизонтали и вертикали соответственно определяется следующим образом:

$$x_m(t) = \sum_{k=1}^K s_k(t) e^{-j2\pi \left(n_x \frac{d_x}{\lambda} \sin(\theta_k) \cos(\varphi_k) + n_y \frac{d_y}{\lambda} \sin(\varphi_k) \right)} + n_m(t),$$

где

$s_k(t)$ – комплексная амплитуда k -го источника сигнала,

θ_k – азимутальный угол направления прихода,

φ_k — угол места направления прихода,

$n_m(t)$ – шум в m -м элементе.

На основе принятого сигнала формируется циклическая ковариационная матрица $\widehat{R}_{y_1}^c$ в соответствии с выражением (1.4).

Так как матрица является эрмитовой (симметричная относительно главной диагонали), используется только верхнетреугольная часть, которая преобразуется в вещественный вектор следующим образом:

$$r = [\text{Re}(r_{1,2}), \text{Im}(r_{1,2}), \dots, \text{Re}(r_{M-1,M}), \text{Im}(r_{M-1,M})].$$

Этот вектор подается на вход LSTM-сети, которая имеет следующую структуру. Размер входного слоя определяется как: $M(M-1)$. Например: при $M = 9$, размер входного вектора равен 72. Два скрытых LSTM-слоя по 256 состояний. Используется функция активации \tanh и sigmoid для гейтов. Слои соединены последовательно. Число выходов: 121×121 , соответствует дискретизации пространства по азимуту и углу места от -60° до $+60^\circ$ с шагом 1° . Формат выхода: двумерный one-hot кодированный вектор, указывающий на позиции источников.

Полученные оценки направлений используются для формирования весовых коэффициентов антенной решетки, обеспечивающих подавление спуфинговых сигналов и усиление полезных сигналов.

Для реализации пространственной фильтрации применяется метод *sidelobe canceller*, который используя направляющие векторы для направлений подавления $wn(\theta, \varphi)$ (спуфинг) и максимума ДН АР $wd(\theta, \varphi)$. Вычисляет коэффициент проекции целевого направления на помеховое следующим образом:

$$rn = \frac{(wn^H * wd)}{(wn^H * wn)}$$

и формирует итоговый весовой вектор антенной решетки:

$$w = wd - wn * rn.$$

Интеграция LSTM-сети в этап антиспуфинга позволяет повысить эффективность обнаружения и подавления ложных сигналов за счет интеллектуальной обработки циклической матрицы докорреляционного сигнала.

Таким образом, синтез адаптивного управления в условиях меняющейся помеховой обстановки соответствует концепции координатно-параметрического управления нестационарными объектами, где адаптация осуществляется по текущим оценкам пространственных и энергетических параметров сигнально-помеховой обстановки [Петров и др., 1980], осуществляется по двухступенчатой схеме интеллектуальной обработки докорреляционных сигналов АР, приведенной на рис. 4.

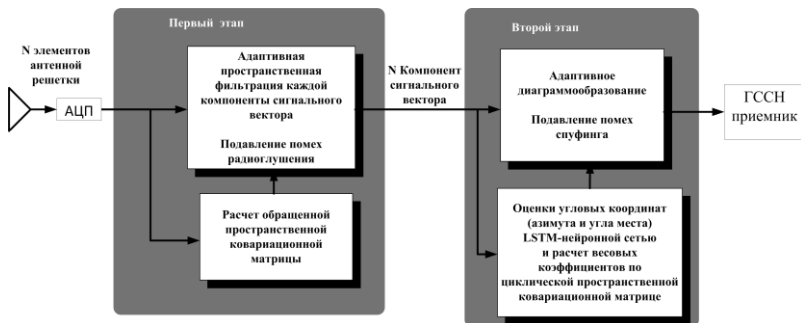


Рис. 4. Этапы синтеза адаптивного управления на основе интеллектуальной обработки докорреляционных сигналов

Результаты моделирования

В данном разделе представлены результаты нейросетевого моделирования оценки направления прихода сигнала (DOA), основанного на использовании LSTM-сети, и проведено сравнение его эффективности с традиционными алгоритмами MUSIC и глубокими нейронными сетями (DNN) из работы [Z.-M. Liu et al., 2018].

Для сравнения LSTM и DNN при различении сигналов с шагом $\Delta\theta = 10^\circ$ проведен эксперимент с двумя сигналами, имеющими угловое расстояние $\Delta\theta = 10^\circ$. DOA первого сигнала изменялся в диапазоне $[-30^\circ, 20^\circ]$ с шагом 1° , а второй сигнал находился под углом $\theta + \Delta\theta$. Полученные результаты (рис. 5) показывают, что как LSTM, так и DNN обеспечивают высокую точность оценки DOA при SNR = 10 дБ.

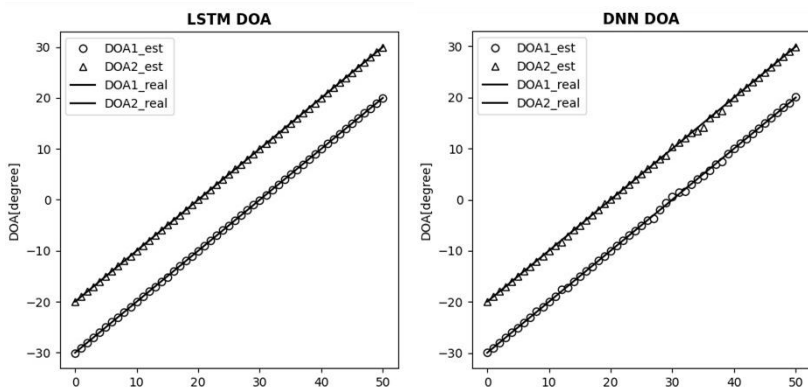


Рис. 5. Сравнение LSTM и DNN при различении сигналов с шагом $\Delta\theta = 10^\circ$

Для анализа RMSE при различных уровнях SNR методы LSTM, MUSIC и DNN сравнивались при уровне шума от -10 дБ до 10 дБ с шагом 2 дБ. Углы прихода сигналов составляли 31° и 41° (в обучающей выборке) и 30° и 39° (вне обучающей выборки). Оценка проводилась на основе среднеквадратичной ошибки (RMSE) по 1000 испытаниям.

Как видно из рисунка 6, все три метода показывают хорошую точность при высоких значениях SNR ($RMSE < 0.5^\circ$). Однако при снижении SNR LSTM сохраняет высокую стабильность и точность, тогда как эффективность MUSIC и DNN заметно ухудшается. Особенно важно отметить, что LSTM сохраняет низкий уровень ошибки даже для углов, не включённых в обучающий набор.

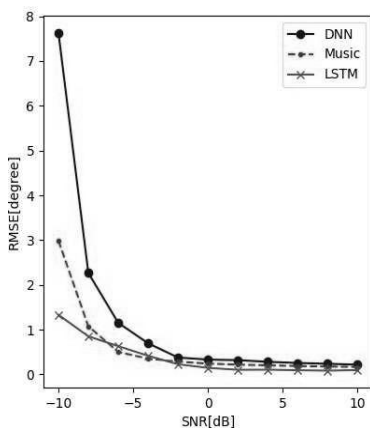


Рис. 6. Анализ RMSE при различных уровнях SNR методы LSTM, MUSIC и DNN

На основе полученных результатов можно обосновать преимущество использования LSTM перед традиционными и другими нейросетевыми подходами.

Во-первых, традиционные алгоритмы вроде MUSIC основаны на предположениях о стационарности сигналов, высоком отношении сигнал/шум (SNR) и точном знании количества источников. В реальных условиях эти предпосылки часто нарушаются: сигналы могут быть короткими, нестационарными, а уровень шума — высоким. Как показано в результатах (рис. 6), эффективность MUSIC существенно падает при SNR ниже 0 дБ, что ограничивает её применимость в сложных радиопомеховых условиях.

Во-вторых, стандартные DNN, хотя и способны обрабатывать нелинейные зависимости, не учитывают временную структуру входных данных. В задачах DOA входные данные — это, как правило, выборки сигнала

лов с антенной решётки или их ковариационные матрицы, которые могут рассматриваться как последовательности. DNN обрабатывает такие данные как статический вектор, теряя временную динамику, что снижает обобщающую способность, особенно на углах, не попавших в обучающую выборку.

Таким образом, нейросетевой подход на основе LSTM выбран как компромисс между высокой точностью, устойчивостью к шуму и способностью к обобщению. Он сочетает в себе гибкость нейросетевого моделирования с архитектурной способностью обрабатывать последовательности, что делает его особенно подходящим для сложных, нестационарных сценариев оценки DOA, где классические методы теряют эффективность.

Список литературы

- [Вейцель и др., 2014] Вейцель А.В., Вейцель В.А., Татарников Д.В. Аппаратура высокоточного позиционирования по сигналам глобальных навигационных спутниковых систем: Высокоточные антенны. Специальные методы повышения точности позиционирования / под ред. М.И. Жодзишского. – М.: Изд-во МАИ-Принт, 2014. – 368 с.
- [Монзинго и др., 1986] Монзинго Р.А., Миллер Т.У. Адаптивные антенные решетки: Введение в теорию. – М.: Радио и связь, 1986. – 448 с.
- [Перов, 2010] Перов А.И. ГЛОНАСС. Принципы построения и функционирования / под ред. А.И. Перова, В.Н. Харисова. – М.: Радиотехника, 2010. – 800 с.
- [Петров, 2015] Петров А.С. Моделирование характеристик адаптивных антенных компенсаторов помех, применяемых в глобальных навигационных спутниковых системах // Антенны. – 2015. – № 8. – С. 44.
- [Dong et al., 2017] Dong K., Zhang Z., Xu X. A hybrid interference suppression scheme for global navigation satellite systems // In Proceedings of the 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, China, 11–13 October 2017.
- [АСИ, 1980] Трапезников В.А., Райбман Н.С., Чадеев В.М. и др. Адаптивная система с идентификацией. – М.: Ин-т проблем управления, 1980. – 67 с.
- [Thanh Han Trong et al., 2022] Thanh Han Trong*, Ngo Duc Nam, Tran Van Hung DOA Estimation Based on LSTM Neural Network with Uniform Linear Antenna Array // UD - JOURNAL OF SCIENCE AND TECHNOLOGY: ISSUE ON INFORMATION AND COMMUNICATIONS TECHNOLOGY. – 2022. – Vol. 20, No. 6.2.
- [Петров и др., 1980] Петров Б.Н., Рутковский В.Ю., Земляков С.Д. Адаптивное координатно-параметрическое управление нестационарными объектами. – М.: Наука, 1980. – 244 с.
- [Liu et al., 2018] Liu Z.-M., Zhang C., and Philip S.Y. Direction-of-arrival estimation based on deep neural networks with robustness to array imperfections // IEEE Transactions on Antennas and Propagation. – 2018. – Vol. 66, No. 12. – P. 7315-7327.

АВТОНОМНАЯ МУЛЬТИАГЕНТНАЯ СИСТЕМА УПРАВЛЕНИЯ ГРУППОЙ РОБОТОВ

Б.К. Лебедев (*lebedev.b.k@gmail.com*)

О.Б. Лебедев (*lebedev.ob@mail.ru*)

М.И. Бесхмельнов (*m_beskhmelnov@mail.ru*)

МИРЭА – Российский технологический университет, Москва

В работе рассмотрены методы управления группой беспилотных летательных аппаратов, обладающих свойствами автономности. Проведен анализ основных стратегий управления и их особенностей. Определены общие принципы и механизмы, необходимые для разработки алгоритма обеспечивающего групповое управление коллектива воздушных роботов, позволяющие каждому роботу автономно ориентироваться в пространстве. При этом каждый робот должен сам, устанавливая свое местоположение, используя данные с камеры или лидара идентифицировать препятствия, а также формировать наиболее рациональный маршрут и принимать решения, направленные на достижения цели и выполнения задачи.

Ключевые слова: беспилотный летательный аппарат, автономность, искусственный интеллект, децентрализованное управление, роевая стратегия.

Введение

Использование беспилотных летательных аппаратов (БПЛА) в составе групп, или, как принято говорить, роев, представляет собой одно из наиболее перспективных направлений развития современной робототехники [Пшихопов и др., 2015], [Нейдорф и др., 2016].

Главная задача при организации роя БПЛА – существенно повысить эффективность управления формированием и поддержанием заданной геометрической структуры, или строя, в котором каждый аппарат занимает строго определенное место относительно других. Это особенно важно при выполнении разнообразных задач, начиная от мониторинга больших территорий и заканчивая поисково-спасательными операциями. Для достижения этой цели необходимо разработать эффективные методы и алго-

ритмы децентрализованного управления, что повышает надежность системы. В случае отказа одного или нескольких аппаратов, весь рой продолжает функционировать, адаптируясь к изменившимся условиям.

Ключевой принцип децентрализованного управления заключается в том, что основная вычислительная нагрузка распределяется между самими аппаратами. Каждый БПЛА в рое принимает решения на основе локальной информации, получаемой от своих непосредственных «соседей». Полностью децентрализованное управление подразумевает, что каждый дрон в группе самостоятельно определяет свою позицию в строю, опираясь только на данные, полученные от ближайших дронов.

Существуют разные подходы к децентрализованному управлению роем дронов, включая коллективный, стайный и роевой методы.

В коллективном подходе дроны действуют скоординировано, придерживаясь общих правил, установленных централизованно, но при этом сохраняют некоторую самостоятельность в принятии решений.

Стайный подход отличается большей гибкостью и адаптивностью. В этом случае каждый дрон реагирует на движения соседних аппаратов, приспосабливаясь к меняющимся условиям окружающей среды.

Роевая стратегия подразумевает наибольшую степень децентрализации, где аппараты взаимодействуют исключительно с ближайшими соседями, не имея ясного представления о глобальной структуре строя. Определение строя – это задание требуемого взаимного расположения аппаратов на плоскости или в пространстве.

Ключевой аспект при разработке такой системы – способность не просто формировать группы БПЛА с назначением «ведущего» аппарата, но и автономно планировать выполнение сложных полетных заданий, учитывая множество переменных факторов. Взаимодействие между аппаратами критически важно для координации действий всего роя и обеспечения успешного выполнения миссии [Котов и др., 2024].

Во-первых, система непрерывно измеряет текущее состояние каждого БПЛА и всего роя в целом. Это включает в себя множество параметров: координаты, высоту, скорость, углы ориентации, статус бортовых систем, данные с целевого оборудования и уровень заряда батареи.

Во-вторых, система сравнивает текущее состояние с желаемым состоянием, определенным на основе заданной миссии. Разница между этими состояниями представляет собой отклонение, которое система стремится минимизировать [Лебедев и др., 2024].

В-третьих, на основе анализа этого отклонения система вычисляет необходимые управляющие воздействия, передаваемые каждому БПЛА для корректировки его положения и действий [Лебедев и др., 2024].

Для получения эффективного результата, при разработке системы такого уровня, необходимо решить задачи в сфере искусственного интеллекта, анализа и обработки больших объемов данных и надежного программного и аппаратного обеспечения [Карпенко, 2021].

1. Использование интеллектуальных технологий для решения задач планирования и управления

В контексте возрастающей сложности задач планирования и управления интеллектуальные технологии приобретают первостепенное значение как средство их эффективного решения и оптимизации. Неопределенность, которая является неотъемлемой частью любой сложной системы, компенсируется использованием интеллектуальных алгоритмов, анализирующих огромный объем данных и принимающих решения в условиях неполной информации [Городецкий, 2016].

Одним из наиболее распространенных сформировавшихся подходов являются экспертные системы (ЭС).

Вторым подходом является разработка программных продуктов, которые воспроизводят некоторые когнитивные возможности человека: распознавание естественного языка, распознавание образов в виде изображений, логические рассуждения и т.д.

В качестве альтернативы используют третий подход – создание искусственных нейронных сетей (ИНС), способных эффективно решать задачи, быстро выполнять обучение на больших объемах данных и обладающих свойством адаптивности к реальным условиям.

Наиболее эффективным является комбинированный подход, объединяющий преимущества разных методов, и позволяющий решать сложные задачи управления и планирования с наилучшим результатом.

2. Интеллектуальная система управления группой роботов

В данной работе предлагается система управления, которая представляет собой интеграцию нескольких ключевых компонентов: ЭС, набора специализированных компьютерных программ, ИНС, а также блока адаптации, основанного на принципах конечных автоматов. Основная цель этой системы заключается в том, чтобы, например, обучить рой БПЛА, не только объединяться в группы и выбирать лидера, но и разрабатывать стратегические планы для выполнения поставленных задач.

Эффективность работы группы БПЛА обеспечивается благодаря наличию заранее подготовленного перечня действий, который представлен в виде набора правил, встроенных в ЭС. Эти правила позволяют БПЛА выполнять различные задачи, адаптируясь к меняющимся условиям местности.

Оператор задаёт полётное задание, а группа БПЛА задействует полученные от него данные, и сведения из окружающей среды для выполнения задач. Разработана система правил, при помощи которой БПЛА контролируют расстояние между собой и не дают им сблизиться с препятствиями [Веселов и др., 2021].

Авторами разработаны алгоритмы на основе коллективной альтернативной адаптации (КАА), позволяющие БПЛА откликаться на изменения в окружающей среде, что повышает взаимодействие между БПЛА, адаптирует их поведение к текущим условиям, что особенно важно для успешного выполнения задачи в нетривиальных и изменяющихся условиях. Ключевая идея разработанной системы состоит в оптимальном построении маршрута передвижения каждого робота для достижения общей цели, предполагающее определение наиболее эффективной конфигурации траектории, учитывающей ряд ограничений: особенности самой цели, встречающиеся препятствия, расстояние между роботами.

Наиболее простым способом, позволяющим коллективу роботов эффективно выполнять задачи, является его разделение на кластеры, которое происходит по разным критериям, например, местоположения роботов в пространстве или разновидность решаемых задач. Кластеру присваивается индивидуальный номер, по которому происходит определение местоположения и выполнение действий каждого робота в текущий момент времени. При этом алгоритм работает последовательно и заключается в передаче информации по цепочке, т.е. каждый робот получает команды от предыдущего и передает их следующему. Такой подход обеспечивает согласованность действий, минимизирует конфликты и позволяет оптимизировать весь процесс. Разработана система управления, которая решает задачу автоматического перестроения группы в условиях неопределенности, т.е. роботы должны уметь адаптироваться к динамически изменяющимся условиям, избегая столкновений и сохраняя общую цель [Байгутлина, 2021].

Построение группы БПЛА в воздухе осуществляется на основе заранее разработанных шаблонов (схем), представляющих собой набор правил, заданных в определенном формате и загруженных в облачную систему. Эти шаблоны определяют желаемую конфигурацию группы БПЛА в зависимости от выполняемой задачи.

Для решения задачи управления группой мобильных роботов был использован подход, основанный на принципах коллективного интеллекта. Система должна обеспечивать автоматическое перестроение роботов в динамической среде с препятствиями, координируя их действия для достижения общей цели. Рой должен эффективно распределять задачи между своими роботами, а также при выходе из строя одного или нескольких роботов выполнить переназначение задач среди оставшихся.

В основе управления каждым БПЛА лежит конфигурация, объединяющая нейронную сеть и конечный автомат. Нейронная сеть выполняет роль «сенсорного анализатора» и «классификатора» – она обрабатывает множество вещественных входных данных от различных датчиков (скорость, положение, расстояние до препятствий, данные от других БПЛА), переводя их в дискретное логическое представление, более удобное для работы конечного автомата. Конечный автомат, в свою очередь, используя эти логические сигналы, вырабатывает управляющие воздействия, определяющие параметры движения БПЛА. Такая гибридная система позволяет сочетать адаптивность нейронной сети к непредсказуемым ситуациям с детерминированностью и предсказуемостью конечного автомата [Успанова и др., 2021]. Конфигурация системы строится на мультиагентном подходе, где каждый БПЛА рассматривается как самостоятельный агент, который функционирует как с окружающей средой, так и с другими представителями группы, что дает возможность решать сложные задачи коллективного управления [Лебедев и др., 2024], [Морозова, 2015], [Веселов и др., 2021].

3. Управление движением группы роботов

В данной работе представлено исследование инновационного метода управления группой автономных мобильных роботов, организованных в линейный строй. Ключевая особенность этого подхода – децентрализованный характер управления. Каждый робот, за исключением ведущего, обладает полной автономностью и принимает решения, основываясь исключительно на информации, полученной от своих непосредственных соседей. Роботы выстроены в линию и пронумерованы от R_0 (ведущий робот) до R_n (замыкающий робот). Такая организация требует от каждого робота R_i (где $i=1,2,\dots,n$), кроме ведущего R_0 , поддержания строго определенного поведения: движения параллельно траектории предшествующего робота R_{i-1} с постоянной скоростью V_i , равной скорости ведущего робота V_0 , и поддержания неизменного расстояния D между собой и роботом R_{i-1} . Таким образом, каждый робот R_i непрерывно корректирует свою траекторию, «выравниваясь» по траектории робота R_{i-1} , обеспечивая тем самым поддержание строя.

Каждый робот описывается вектором состояния $W_i=(x_i, y_i, \alpha_i, h_i, V_i)$, где (x_i, y_i) – координаты робота в абсолютной системе координат, V_i – его скорость, α_i – угол между вектором скорости V_i и базовой линией, а h_i – модуль вектора скорости (по сути, скалярная величина скорости). Для начала движения задаются параметры ведущего робота: угол α_0 между вектором его скорости и базовой линией, а также постоянное расстояние D между роботами в строю. Суть децентрализованного управления заключается в достижении равенства условий движения для каждого робота относительно своего предшественника [Морозова, 2015].

Каждый ведомый робот R_i стремится выполнить два основных условия: поддержание постоянного расстояния D до робота R_{i-1} и движение параллельно траектории робота R_{i-1} . Это достигается за счет постоянного обмена информацией между соседними роботами, позволяющего каждому роботу оценивать своё положение относительно соседа и корректировать свою скорость и направление движения. В алгоритме управления учитывается информация о текущем положении и скорости как самого робота, так и робота, идущего впереди него. Таким образом, каждый робот автономно регулирует свою траекторию, стремясь к идеальному выполнению условий поддержания строя, при этом основываясь лишь на данных от своего непосредственного предшественника. Это исключает необходимость глобального планирования траекторий и повышает устойчивость системы к сбоям отдельных роботов. В случае, если один из роботов выходит из строя, остальные продолжают двигаться в соответствии с установленными правилами, адаптируясь к изменившимся условиям. Таким образом, децентрализованный подход обеспечивает робастность и масштабируемость системы управления группой мобильных роботов [Лебедев и др., 2024], [Морозова, 2015].

Основная идея при управлении ведомыми роботами заключается в выполнении (достижении R_i курса робота R_{i-1}) равенств 3.1:

$$\begin{aligned} 1. & V_{i-1}(t) = V_i(t) \\ 2. & h_{i-1}(t) = h_i(t) \\ 3. & x_i(t) - x_{i-1}(t) = D \\ 4. & y_i(t) = y_{i-1}(t) \\ 5. & \alpha_i(t) = \alpha_{i-1}(t) = \alpha_0 \\ & \varphi_i(t) = \alpha_i(t) - \alpha_{i-1}(t) \text{ — угол между } V_{i-1}(t) \text{ и } V_i(t). \end{aligned} \quad (3.1)$$

Система управления группой роботов, движущихся параллельно, основана на непрерывном пересчете параметров движения каждого робота в режиме реального времени. Представьте себе колонну роботов, где каждый робот R_i движется рядом с предыдущим R_{i-1} . Задача системы – поддерживать заданное расстояние и направление движения между роботами, корректируя их траекторию. Ключевым элементом является вычисление ошибки каждого робота, обозначаемой δ_i . Эта ошибка представляет собой сумму отклонений по различным параметрам: горизонтальное расстояние (x), вертикальное расстояние (y), высота (h) и угол ориентации (α). Каждое отклонение взвешивается соответствующим коэффициентом (k_1, k_2, k_3, k_4), отражающим важность данного параметра для поддержания строя. Общая ошибка всей группы роботов Δ является суммой индивидуальных ошибок δ_i каждого робота: $\Delta = \sum_i \delta_i$.

$$\delta_i = k_1(|D - (x_i(t) - x_{i-1}(t))|) + k_2(|y_i(t) - y_{i-1}(t)|) + k_3(|h_i(t) - h_{i-1}(t)|) + k_4(|\alpha_i(t) - \alpha_0|). \quad (3.2)$$

Система управления стремится минимизировать общую ошибку Δ , ограничивая при этом максимальные значения отклонений скорости δV и угла «рыскания» ($\varphi_{i,i-1}(t+1)$). Угол «рыскания» $\varphi_i(t)$ – это разница в углах ориентации соседних роботов $\alpha_i(t)-\alpha_{i-1}(t)$. Система использует четыре основных правила коррекции [Веселов Г.Е. и др., 2021].

Правило 1: Коррекция угла ориентации. Если угол рыскания $\varphi_i(t)$ между роботом R_i и его предшественником R_{i-1} больше нуля (т.е., робот R_i отклоняется от курса R_{i-1}), система корректирует угол ориентации $\alpha_i(t)$ робота R_i , стремясь уменьшить $\varphi_i(t)$ до нуля. Эта коррекция выполняется в соответствии с определенным алгоритмом (равенства 3.1). Цель – выравнивание курса робота R_i с курсом R_{i-1} .

Правило 2: Коррекция высоты. Если существует вертикальное расхождение $\varepsilon_h=h_i(t)-h_{i-1}(t)$ между высотами роботов R_i и R_{i-1} , и это расхождение не превышает заданного порога, система корректирует скорость изменения высоты $h_i(t)$ робота R_i , используя выражение $h_i(t+1)=h_i(t)+\zeta_v$. Параметр ζ_v представляет собой корректировку скорости, вычисляемую исходя из величины расхождения ε_h . Задача – сгладить разницу высот между роботами. Время отработки отклонения критический фактор – его превышение приведет к отказу от коррекции.

Правило 3: Коррекция по вертикальному смещению. Если существует вертикальное смещение $y_i(t)-y_{i-1}(t)=\varepsilon_y$ между роботами R_i и R_{i-1} , и это смещение больше нуля, система выполняет мгновенную коррекцию вертикальной координаты робота R_i : $y_i(t+1)=y_i(t)+\zeta_y$. Параметр ζ_y – это величина корректировки, зависящая от ε_y . Цель – минимизировать $|\varepsilon_y|$, приводя к выравниванию вертикального положения роботов.

Правило 4: Коррекция по горизонтальному смещению. Если горизонтальное расстояние $x_i(t)-x_{i-1}(t)=\varepsilon_x$ между роботами R_i и R_{i-1} больше заданного расстояния D , система корректирует горизонтальную координату $x_i(t)$ робота R_i : $x_i(t+1)=x_i(t)+\zeta_x$. Параметр ζ_x служит для внесения поправок в траекторию робота, основываясь на разнице между абсолютной величиной ошибки расстояния ($|\varepsilon_x|$) и целевым расстоянием D . Цель состоит в минимизации $|\varepsilon_x|-D$. Коррекция реализуется путем кратковременного изменения угла ориентации робота R_i на величину δ_i , рассчитанную на основе ошибки. В течение короткого интервала времени δ_i робот движется под скорректированным углом $\alpha_i^*(t)=\alpha_i(t)+\delta_i$, после чего возвращается к исходному углу $\alpha_i(t)$. Такой подход обеспечивает плавную коррекцию траектории. Система непрерывно контролирует и корректирует ошибки для поддержки заданного строя. Важно отметить, что конкретные значения параметров $k_1, k_2, k_3, k_4, D, \zeta_v, \zeta_x, \zeta_y$ и алгоритм коррекции угла (3.1) определяют эффективность и точность управления системой. Выбор этих параметров зависит от характеристик роботов, условий окружающей среды и требований к точности движения, рис. 1.

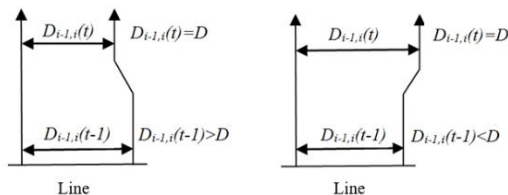


Рис. 1. Схема маневра, выполняемого роботом для коррекции параметра $x_i(t)$

Первый маневр предназначен для решения задачи управления при условии оказания управляющего воздействия на направление перемещения робота, а второй – при условии оказания управляющего воздействия на скорость агента [Веселов и др., 2021].

4. Движение роботов на основе коллективной адаптации

Система управления группой роботов осуществляется алгоритмом КАА. В данном контексте, объектом адаптации для каждого отдельного робота R_i служит вектор W_i , определяющий его состояние в пространстве. Этот вектор включает в себя несколько параметров: координаты робота (x_i, y_i) в абсолютной системе координат, его ориентацию α_i (угол поворота относительно оси Z), высоту h_i (робот имеет вертикальную степень свободы) и, наконец, его линейную скорость V_i .

Совокупность всех векторов W_i для каждого робота в группе составляет коллективный объект оптимизации. Другими словами, цель управления – найти оптимальное сочетание параметров всех роботов, обеспечивающее эффективное выполнение групповой задачи. Для достижения этой цели, каждому параметру p_{ij} вектора W_i (где p_{ij} может представлять собой x_i , y_i , α_i , h_i или V_i) присваивается свой собственный автомат адаптации (АА) [Котов и др., 2024], [Городецкий, 2016].

Каждый АА моделирует поведение объекта адаптации в окружающей среде, используя для этого конечный автомат с двумя группами состояний. Первая группа, $C_1 = \{c_{1l} | l=1, 2, \dots, g\}$ и вторая группа, $C_2 = \{c_{2l} | l=1, 2, \dots, g\}$, представляют две альтернативные стратегии поведения: A_1 – изменение значения параметра и A_2 – сохранение его текущего значения. Параметр « g » определяет глубину памяти автомата, т.е., сколько предыдущих состояний учитывается при принятии решения. Выходной алфавит автомата $A = \{A_1, A_2\}$ отражает эти две стратегии.

В качестве входного сигнала автомат использует отклики среды, представленные алфавитом $Q = \{+, -\}$. Символ « $+$ » обозначает «поощрение» (ситуация, когда изменение параметра приблизило систему к цели), а символ « $-$ » обозначает «наказание» (ситуация, когда изменение параметра ухудшило результат). Переходы между состояниями автомата определяются этим входным сигналом. Схема представлена на рис. 2.

Рассмотрим функционирование одного АА. Начальное состояние выбирается случайным образом. Автомат получает сигнал «поощрение» (+) или «наказание» (-), в зависимости от того, улучшило ли изменение параметра общее состояние системы. На основе полученного сигнала и текущего состояния, автомат переходит в новое состояние, выбирая либо A_1 (изменение параметра), либо A_2 (без изменений). Локальная цель каждого АА – достижение состояния, при котором его «оценка» равна нулю. Оценка определяет степень близости текущего значения параметра к его оптимальной конфигурации [Котов и др., 2024].

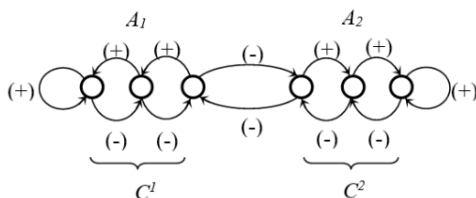


Рис. 2. Схема переходов автомата адаптации

Каждый автономный агент (робот) стремится к локальной оптимизации своего параметра. Глобальная оптимизация поведения всей группы роботов достигается за счет эмерджентного взаимодействия этих локальных оптимизаций. Механизм адаптивен к динамике окружающей среды, обеспечивая непрерывную корректировку параметров как на уровне отдельных агентов, так и на уровне системы в целом, что гарантирует эффективное и гибкое управление групповым поведением. Глубина памяти «g» является критическим параметром, определяющим способность системы к обучению на основе «исторических» данных. Увеличение «g» расширяет горизонт планирования и адаптации. Процесс КАА, который управляет передвижением группы роботов, выстроенных в линию, осуществляется в рамках четко определенной структуры, состоящей из четырех последовательных тактов для каждого отдельного робота на каждом временном шаге t . Основной локальной целью данного процесса является достижение состояния, при котором оценка адаптации объекта равняется нулю. В то же время глобальная цель всего коллектива роботов заключается в достижении определенного состояния S , то есть таких значений параметров для роботов, при которых будет минимизирована «Ошибка перемещения». В рамках процесса КАА, который управляет движением группы роботов, необходимо отметить, что перепланирование параметров движения для каждого робота, обозначенного как R_i , происходит в каждый момент времени t .

В ходе первого такта коллективной адаптации для каждого параметра R_i вычисляются значения вектора W_i , который представляет собой набор данных, необходимых для дальнейшего анализа.

На втором такте происходит генерация откликов среды для каждого автомата адаптации, которые могут быть классифицированы как «поощрение» или «наказание», в зависимости от того, насколько успешно робот выполняет поставленные задачи.

На третьем этапе каждый механизм адаптации переходит в новое состояние, основываясь на полученной информации об окружающей среде. Это позволяет роботам реагировать на изменения и подстраивать свои действия.

На четвертом этапе, в зависимости от состояния механизма адаптации, принимается решение о дальнейших действиях:

- Если механизм адаптации показывает, что текущая стратегия работает хорошо (состояние группы C^1ij), то параметры поведения робота прежние.
- Если же механизм адаптации сигнализирует о необходимости изменений (состояние группы C^2ij), то параметры поведения корректируются пропорционально отклонению $\mu_{ij}(t)$, чтобы адаптироваться к новым условиям [Котов Д.В. и др., 2024].

5. Экспериментальные исследования

Для тестирования алгоритмов управления коллективом автономных БПЛА были заданы следующие условия полетного задания: горизонтальный полет (автономный); высота не меняется (постоянная); коллектив осуществляет полет из точки взлета до указанной цели с огибанием встречных препятствий и последующим возвратом в точку взлета. Коллектив БПЛА содержит до шести аппаратов. Расстояние между БПЛА в коллективе варьируется от 0,5 метра до 40 метров. Определенная цель находится на расстоянии от 1 до 10 километров от точки взлета. Максимальная скорость – до 130 км/ч. Типы препятствий включают: прямоугольные формы (моделирующие строения, дома) или круглые формы (моделирующие деревья, лес), рис. 3. Коллектив воздушных роботов при выполнении задания, осуществляет полет в виде неких геометрических фигур: линия, колонна, треугольник и т.д. рис. 4.

В качестве критерия была использована аддитивная свертка, в которой параметрами, позволяющими оценить качество выполнения задачи при движении коллектива БПЛА являются: длина траектории (P_L), показатель безопасности (S_m), время выполнения задания (t_m), коэффициент успешности миссии (M) [Пшихопов и др., 2015].

При этом степень важности того или иного параметра определяется при помощи коэффициентов K_1 , K_2 , K_3 , и K_4 .

$$F=\kappa_1P_L+\kappa_2S_m+\kappa_3t_m+\kappa_4M. \quad (5.1)$$

При этом необходимо повысить показатель безопасности (S_m) и коэффициент успешности миссии (M), а показатели длина траектории (P_L), время выполнения задания (t_m) – понизить.

Проведя сравнительный анализ полученных после тестирования результатов (в тестах применялся полученный новый метод управляющий коллективом БПЛА в строе), было установлено, что при его использовании результативность реализации поставленной задачи увеличилась на 18-20 процентов.



Рис. 3. Обход препятствий



Рис. 4. Построение группы БПЛА

В табл. 1 указаны усредненные значения используемых в работе параметров и сравнение их с существующими методами. Оценка временной сложности представленного метода составляет $O(n^2) - O(n^3)$.

Таблица 1

Методы	Показатели			
	S_m	P_L	t_m	F
Метод потенциальных полей 1	0,104	0.01975	0.01593	0,6
Метод потенциальных полей 2	0,028	0,00665	0,00532	0,2
Метод диаграмм Вороного	0,44	0,02052	0,01728	0,6
Метод реактивной навигации	0,656	0.02412	0.02473	0,8
Метод управления с применением нечетких систем	0,495	0,02507	0,01253	1,0
DVN-метод в классическом базисе	0,068	0.0313	0.07008	1,0
DVN-метод в базисе формальных нейронов	0,068	0.0313	0,07008	1,0
Нейросетевой гибридный метод	0,076	0.0317	0.03795	1,0
Генетический поиск без картографии	0,074	0,01353	0,01103	0,4
Генетический поиск с картографией	0,565	0.03094	0.023 73	1,0
Метод планирования с использованием неустойчивых режимов	0,317	0,02032	0,01964	0,8
Разработанный метод	0,176	0.0324	0.0162	1,0

Заключение

В данной работе представлено исследование новых методов, предназначенных для моделирования движения группы роботов.

Ключевым элементом разработанной системы управления является алгоритм коллективной альтернативной адаптации. Разработанная структура процесса КАА обеспечивает согласованное движение всей группы ро-

ботов. Суть механизма адаптации заключается в сопоставлении каждому параметру вектора состояния робота своего собственного автомата адаптации. Эти автоматы, работая параллельно и взаимодействуя друг с другом, обеспечивают гибкое и эффективное управление.

Каждый робот принимает решения автономно, основываясь на информации о своем собственном состоянии и состоянии ближайших соседей, что обеспечивает высокую устойчивость. Система способна автоматически корректировать отклонения, обеспечивая поддержание заданного построения. Для компенсации отклонений в работе роботов разработан специальный маневр, представляющий собой последовательность управляющих действий, которые автоматически рассчитываются на основе текущей ситуации. Проведены тестовые испытания и сравнения с существующими методами и алгоритмами, подтвердившие эффективность разработанного метода.

Список литературы

- [Пшихопов и др., 2015] Пшихопов В.Х., Медведев М.Ю., Гуренко Б.В. Алгоритмы адаптивных позиционно-траекторных систем управления подвижными объектами // Проблемы управления, 2015. – С. 66-74. *(статья в журнале, авторов не более трех)*.
- [Нейдорф и др., 2016] 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. – С. 127-143. *(статья в журнале, авторов более трех)*.
- [Котов и др., 2024] Котов Д.В., Лебедев О.Б. Управление передвижением группы БПЛА с соблюдением геометрической структуры строя на основе альтернативной коллективной адаптации // Известия ЮФУ. Технические науки. – Ростов-на-Дону: Изд-во ЮФУ, 2024. – С. 155-167. *(статья в журнале, авторов не более трех)*.
- [Лебедев и др., 2024] Лебедев Б.К., Лебедев О.Б., Бесхмельнов М.И. Децентрализованное управление группой автономных подвижных объектов при формировании траектории движения // Известия ЮФУ. Технические науки. – Ростов-на-Дону: Изд-во ЮФУ, 2024. – С. 177-190. *(статья в журнале, авторов не более трех)*.
- [Карпенко, 2021] Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учебное пособие. – 3-е изд. – М: Изд-во МГТУ им. Н.Э. Баумана, 2021. – 448 с. *(книга, авторов не более трех)*.
- [Городецкий, 2016] Городецкий В.И. Управление коллективным поведением роботов в автономной миссии // Робототехника и техническая кибернетика. – 2016. – С. 40-54. *(статья в журнале, авторов не более трех)*.
- [Байгутлина, 2021] Байгутлина И.А. Реализация новых сервисов с использованием робототехнических комплексов // Славянский форум, 2021. – С. 162-170. *(статья в журнале, авторов не более трех)*.

- [**Морозова, 2015**] Морозова Н.С. Децентрализованное управление движением строя роботов при динамически изменяющихся условиях // Искусственный интеллект и принятие решений, 2015. – С. 65-74. *(статья в журнале, авторов не более трех)*.
- [**Веселов и др., 2021**] 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. *(статья в журнале, авторов не более трех)*.
- [**Успанова и др., 2021**] Успанова Р.А. Актуальные проблемы управления группой БПЛА // Труды тринадцатой общероссийской молодежной научно-технической конференции: в 2х т. Сер. Библиотека журнала «Военмех. Вестник БГТУ», 2021. – С. 128-130. *(статья в журнале, авторов не более трех)*.

УДК 004.85

doi: 10.15622/rcai.2025.060

ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ С ВНЕШНИМ УЧИТЕЛЕМ ДЛЯ УПРАВЛЕНИЯ ЭНЕРГООБЪЕКТАМИ

О.Ю. Марьясин (*maryasin2003@list.ru*)

А.Н. Плохотнюк (*admin@nixson.ru*)

Ярославский государственный технический университет,
Ярославль

Рассмотрена задача управления энергообъектом с применением алгоритмов обучения с подкреплением. Для решения проблемы длительного обучения агента авторы предложили подход, заключающийся в использовании внешнего учителя. В работе описаны различные способы применения внешнего учителя и рассмотрена постановка задачи управления энергообъектом, допускающая использование популярных алгоритмов обучения с подкреплением с внешним учителем. Результаты обучения агентов на виртуальной среде показали, что для всех алгоритмов обучения с подкреплением применение внешнего учителя позволяет достичь гораздо меньшего энергопотребления за фиксированное время чем для подобных алгоритмов без внешнего учителя. Таким образом, использование алгоритмов обучения с подкреплением с внешним учителем может как значительно ускорить обучение агентов, так и повысить эффективность управления на начальном этапе обучения.

Ключевые слова: управление энергообъектами, обучение с подкреплением, внешний учитель, оптимизация энергопотребления.

Введение

Одним из многообещающих подходов к управлению энергетическими объектами является обучение с подкреплением (Reinforcement Learning – RL). RL это область машинного обучения, в которой используется понятие агента [Graesser et al., 2019]. В RL агент взаимодействует со средой, наблюдает текущее состояние среды и использует данную информацию при выборе своего действия. Среда в результате действия агента переходит в следующее состояние и возвращает агенту свое новое состояние и вознаграждение, позволяющее агенту оценить успешность его действия.

В настоящее время алгоритмы RL широко применяются для решения сложных задач в таких областях как игры, робототехника, автономные мобильные транспортные средства и др. Растет интерес, связанный с использованием алгоритмов RL для управления сложными техническими объектами и системами, в том числе энергетическими. Например, согласно [Perera et al., 2021] число публикаций посвященных использованию методов RL в области энергетических систем с 2000 года неуклонно растет, а с 2016 года темпы роста числа публикаций значительно ускорились. В том числе растет число публикаций, описывающих применение методов RL для управления такими энергетическими объектами как здания и инженерные системы зданий [Sierla et al., 2022].

Первый опыт применения RL для управления энергообъектами выявил, как достоинства, так и некоторые проблемы, связанные с использованием данного подхода. Несомненным достоинством алгоритмов RL по сравнению с классическими методами оптимального управления, таким как Model Predictive Control [Afram et al., 2014], является то, что для их работы не требуется знания точных и сложных моделей энергообъектов [Yu et al., 2021]. Агенты RL могут методом проб и ошибок изучить оптимальную стратегию управления энергообъектом. Это также решает проблемы, связанные с изменчивостью и неопределенностью параметров энергообъекта.

Главной проблемой, с которой столкнулись исследователи при применении алгоритмов RL для управления энергообъектами является неприемлемо длительное время обучения агента. По этой причине ряд авторов пришел к выводу, что RL является проблематичным подходом к управлению энергообъектами [Sierla et al., 2022]. Кроме того, поскольку агент RL обучается методом проб и ошибок, то длительное время стратегия управления может быть неэффективной или вообще энергозатратной. Поэтому большинство авторов считают, что основное обучение агента RL не должно производиться на реальном объекте в реальном масштабе времени. Для этого необходимо использовать виртуальные среды и проводить обучение в сжатом модельном времени [Sierla et al., 2022]. Последующее применение и дообучение агента может происходить в реальных условиях.

К другим вопросам, которые необходимо решить при применении алгоритмов RL для управления энергообъектами относятся: выбор переменных среды, образующих множество состояний, выбор функции вознаграждения, адекватной поставленной цели управления, выбор алгоритмов RL и настройка их гиперпараметров [Sierla et al., 2022]. Анализ литературы показывает, что сначала наиболее популярным алгоритмом RL, который использовался для решения задач управления энергообъектами был Q-Learning [Wang et al., 2023]. Это связано с простотой реализации данного алгоритма. Затем, когда появились и стали доступными алгоритмы глубокого RL и алгоритмы RL с непрерывным множеством действий, то они также стали применяться для управления энергообъектами [Yu et al., 2021].

1. RL с внешним учителем

В научной литературе по RL нет изобилия публикаций, в которых описываются подходы, близкие к рассматриваемому в данной работе. В [Argerich et al., 2020] внешний репетитор (Tutor) применялся для улучшения качества обучения агента за счет использования внешних знаний для управления решениями агента. Авторы назвали данный метод обучением с подкреплением с использованием внешних знаний (External knowledge). Внешние знания, такие как экспертные или предметные знания, выражаются в виде программируемых функций, которые агент использует на этапе обучения. В [Argerich et al., 2020] рассматривается два вида программируемых функций – это функции ограничения и направляющие функции. Функция ограничения принимает текущее состояние среды и возвращает вектор, указывающий, можно ли выполнить соответствующее действие агента или нет. Направляющая функция принимает состояние и вознаграждение в качестве входных данных и выдает вектор, представляющий вес каждого действия агента.

Более известен другой подход к RL, получивший название система учитель-ученик (Teacher-student) [Zimmer et al., 2014]. В рамках этой системы агент-ученик учится выполнять задачу посредством RL, в то время как агент-учитель может оказывать помощь ученику учиться быстрее. В этой схеме предполагается что учитель также является агентом RL и уже усвоил оптимальную политику. Кроме того, учитель может дать лишь ограниченное количество советов, называемое бюджетом. В [Zimmer et al., 2014] представлено четыре эвристических метода для определения того, когда учитель может давать совет, включая раннее консультирование, консультирование по важности, исправление ошибок и прогнозное консультирование. При этом учитель может дать совет в форме действия, которое ученик должен выполнить (действие, лучшее, чем объявленное учеником).

В данной работе предлагается подход, использующий некоторые идеи из отмеченных ранее источников. В данном подходе в классическую схему RL, включающую агента и среду вводится внешний блок учитель (Teacher). Учитель принимает информацию о множестве состояний среды, действиях агента и некоторую внешнюю информацию (внешние сигналы), недоступную для агента. Когда наступает определенное событие, связанное со средой или агентом, учитель тем или иным способом оказывает воздействие на агента. При этом учитель может не только подсказывать агенту какое действие будет лучшим из сформированных агентом действий, но и непосредственно участвовать в формировании действия или даже выполнять определенные действия вместо агента. Этим учитель отличается от репетитора из [Argerich et al., 2020], способного лишь советовать агенту предпочтительные действия. Имеется два основных отличия пред-

лагаемой системы от системы учитель-ученик. Первым является то, что учитель не обязан быть агентом RL и может быть реализован любым другим способом. Например, он может быть построен на основе правил. Правила не должны охватывать все возможные ситуации. Они служат для отработки наиболее важных или критических событий. Второе отличие в том, что учителю доступна дополнительная внешняя информация, которая недоступна агенту. Важно отметить, что учитель не может заменить ученика. Учитель не обладает всеми способностями, которые получает агент в процессе обучения. Учитель только помогает агенту учиться быстрее, за счет того, что он позволяет агенту избегать неэффективных стратегий управления.

Далее рассматриваются различные способы использования внешнего учителя. Это стимулирование агента учителем, непосредственное изменение его действия и комбинированный способ. Блок-схема поясняющая способ стимулирования агента учителем показана на рис. 1. На рис. 1 использованы следующие обозначения: s_t – состояние среды в момент t , $s_t \in S$, r_t – вознаграждение (reward), a_t – действие (action) агента, $a_t \in A$, g_t – функция, выполняющая роль стимула для агента.

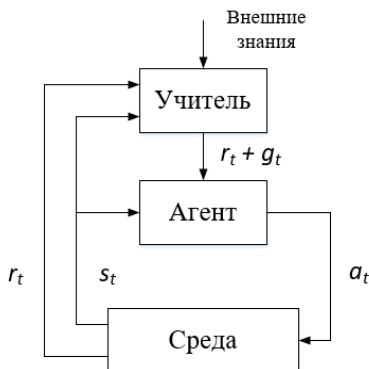


Рис. 1. Блок-схема для стимулирования агента учителем

Способ стимулирования заключается в том, что учитель изменяет вознаграждение r_t , получаемое агентом увеличивая (стимулируя) или уменьшая его. Функция стимула g_t выполняет роль функции штрафа в методах оптимизации. Например, когда необходимо заставить агента соблюдать ограничения накладываемые на состояния среды s_t . Учитель для формирования функции g_t может использовать как информацию о состоянии s_t , передаваемую агенту, так и внешнюю информацию о среде, которая не доступна агенту.

На рис. 2 показана блок-схема, поясняющая способ непосредственного изменения действия агента учителем. На рис. 2 к обозначениям, показанным на рис. 1 добавлены: h_t – сигнал учителя, f_t – преобразованное действие. Блок трансформации F – в общем случае нелинейный блок, в котором на основании информации о действии агента a_t и сигнала учителя формируется новое действие f_t , передаваемое в среду.

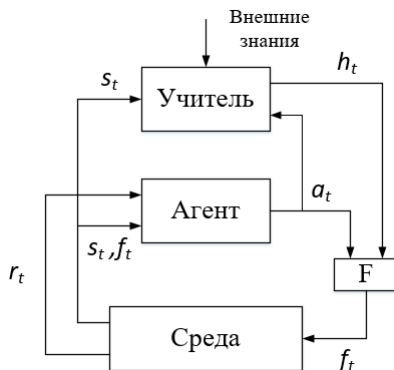


Рис. 2. Блок-схема для непосредственного изменения действия агента учителем

Способ непосредственного изменения действия агента учителем заключается в том, что при наступлении определенного события (внутреннего или внешнего), связанного со средой или агентом, учитель может изменить действие агента, усиливая или, наоборот, ослабляя его. При этом учитель изменяет действия агента только для конечного числа определенных ситуаций и только в том случае если действие, предложенное агентом, не соответствует требуемой реакции на данную ситуацию. Таким образом, учитель может корректировать действия агента, приводящие к неэффективным стратегиям управления, что особенно важно на начальных этапах обучения.

Учитель для формирования h_t может использовать как информацию о состоянии s_t , передаваемую агенту, так и внешнюю информацию, которая не доступна агенту. Информация о преобразованном действии должна быть добавлена к переменным состояния s_t и использоваться при обучении агента. По мере того, как агент накапливает больше опыта, он изучает свои действия и действия, скорректированные учителем, тем самым улучшая свои результаты. Дальнейшее развитие данного способа может привести к имитационному обучению (Imitation Learning) [Shenfeld et al., 2023], когда агент учится имитировать действия учителя.

Блок-схема, поясняющая комбинированный способ использования учителя показана на рис. 3. Комбинированный способ объединяет возможности двух предыдущих способов.

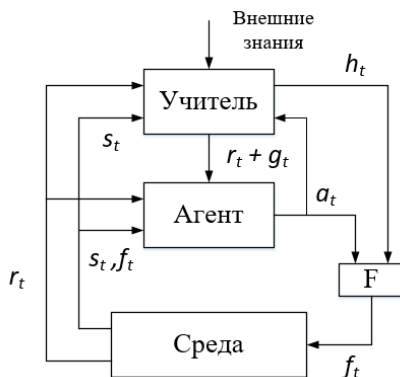


Рис. 3. Блок-схема для комбинированного способа использования учителя

2. Задача управления энергообъектом

Прежде чем решить задачу управления энергообъектом с использованием RL необходимо ответить на ряд вопросов: какая цель управления, что будет использоваться в качестве среды, какие переменные среды войдут в множество состояний s_t , какое вознаграждение будет получать агент и какие управляющие воздействия на среду составят набор действий агента a_t .

В качестве энергообъектов в данной работе рассматриваются здания. На здания приходится около 30–40% всей энергии, потребляемой как в развитых, так и в развивающихся странах [Hossain et al., 2023]. Следовательно, необходимо разрабатывать эффективные методы управления энергопотреблением зданий, которые могут обеспечить оптимальный компромисс между потреблением энергии и обеспечением комфортного микроклимата помещений здания. Одной из наиболее энергоемких инженерных систем здания является система отопления, вентиляции и кондиционирования (ОВК). Параметры работы ОВК оборудования могут сильно влиять на энергопотребление здания [Марьясин и др., 2017]. Поэтому целью управления будет снижение энергозатрат на отопление и охлаждение здания с учетом сохранения комфортного микроклимата внутри помещений здания.

В настоящее время в качестве среды для обучения агентов RL при решении задачи управления энергопотреблением зданий принято использовать виртуальные среды на базе энергомоделей, построенных с применением систем энергомоделирования (Building Energy Modeling – BEM)

[Wang et al., 2023]. При этом, в большинстве случаев, в качестве ВЕМ системы применялась популярная программа EnergyPlus [Wang et al., 2023]. Поэтому в данной работе в качестве виртуальной среды также использовалась ВЕМ система EnergyPlus. Энергомоделирование производилось для того же 5-зонного одноэтажного здания, с той же системой ОВК и при тех же условиях, что и в [Maryasin et al., 2023].

В научной литературе нет единого мнения, какие переменные среды необходимо включать в состав состояния s_t при решении задачи управления энергопотреблением здания с помощью RL. Включение в состав состояния s_t большого числа переменных приводит к резкому росту трудоемкости обучения агентов RL (проклятие размерности) [Jia et al., 2019]. Поэтому в состав состояния s_t стараются включить только наиболее важные, по мнению авторов, переменные. В данной работе состояние s_t включает температуру зон здания, показатель, определяющий степень комфорта людей в помещении (Predicted Mean Vote [Dyvia et al., 2021]) в каждой из зон здания и наличие людей в помещениях зон здания. Кроме того, в зависимости от способа использования учителя, в состояние s_t могут включаться задания локальным регуляторам системы ОВК.

Для достижения поставленной цели функция вознаграждения должна поощрять агента к снижению энергопотребления при сохранении комфортного микроклимата внутри помещений здания. В соответствии с этим функция вознаграждения r_t агента будет иметь вид

$$r_t = -q_t - \lambda_1 p_t + \lambda_2 g_t. \quad (2.1)$$

где q_t – суммарное количество энергии, затраченное на отопление и охлаждение всех зон здания в момент времени t , p_t – функция штрафа за нарушение ограничений по температуре в каждой из зон здания, g_t – функция стимула учителя, λ_1 , λ_2 – заданные коэффициенты. Функция штрафа p_t может быть постоянной, штрафуя за любые нарушения ограничений по температуре или она может зависеть от величины отклонения от заданного температурного диапазона

$$T_{ztl} \leq T_{zt} \leq T_{ztu}, z = 1, \dots, Z, t = 0, \dots, H, \quad (2.2)$$

где T_{ztl} , T_{ztu} – минимальное и максимальное значения температуры в z -й зоне здания в момент времени t , H – горизонт управления, Z – число зон здания. Функция стимула также может быть постоянной, штрафуя за любые неправильные действия агента или она может зависеть от степени различия действий агента и учителя. Для способа непосредственного изменения действия агента учителем $\lambda_2 = 0$.

Набор действий агента $a_t = \{T_{zhsb}, T_{zgst}\}$ включает задания на отопление T_{zhsb} и охлаждение T_{zgst} локальных регуляторов температуры зон здания. Таким образом, агент RL учится решать задачу определения значений заданий

по температуре T_{zhst} и T_{zcst} максимизирующих функцию вознаграждения агента (2.1), тем самым минимизируя количество энергии, затраченное на отопление и охлаждение всех зон здания с учетом ограничения (2.2).

Как было сказано в разделе 1 учитель может обладать некоторой внешней информацией, недоступной для агента. В качестве такой информации в задаче управления энергопотреблением здания с учетом наличия людей в помещениях выступают данные о графике работы персонала здания в рабочие и выходные дни. Тогда с учетом данной информации функция стимула g_t в (2.1) может быть реализована следующим образом. Эта функция равна нулю в рабочее время и равна

$$g_t = |T_{zhst} - T_{zhstb}| + |T_{zcst} - T_{zcstl}| \quad (2.3)$$

в нерабочее время, где T_{zhstb} , T_{zcstl} – минимальные значения задания для отопления и охлаждения в z -ой зоне здания в момент времени t . Аналогично может быть реализован блок F в схеме для непосредственного изменения действия агента учителем. В рабочее время выход блока f_i равен действию агента a_i , в нерабочее время f_i принимает значения

$$f_i = \min(a_i, h_i), \quad (2.4)$$

где $h_i = \{T_{zhstb}, T_{zcstl}\}$ – значения, сообщаемые учителем. При применении комбинированного способа использования учителя функции (2.3) и (2.4) реализуются одновременно.

3. Решение задачи управления энергообъектом

Для управления энергопотреблением здания использовались алгоритмы RL с непрерывным множеством действий Proximal Policy Optimization (PPO), Advantage Actor Critic (A2C), Deterministic Policy Gradient (DPG) и Deep Deterministic Policy Gradient (DDPG).

В табл. 1 приведены значения суточного энергопотребления для отопления и охлаждения помещений здания для двух вариантов, соответствующих одному и тому же зданию. Вариант 1 включает значения, полученные с помощью указанных алгоритмов RL для способа непосредственного изменения действия агента учителем. Вариант 2 – значения, полученные для способа стимулирования. Во всех вариантах используется график посещения людьми различных зон здания и график работы системы ОВК здания, предусматривающие снижение температуры в нерабочее время. Обучение агентов производилось в течение 10000 шагов.

Анализ данных из табл. 1 показывает, что способ непосредственного изменения действий агента учителем, обеспечивает получение лучших результатов, чем способ стимулирования. Это объясняется тем, что в первом способе учитель непосредственно принимает участие в управлении, выдавая правильные действия, в то время как для способа стимулирования агент не успевает обучиться достаточно хорошо за фиксирован-

ное время. Следовательно, применение способа непосредственного изменения действия агента учителем позволяет быстрее добиться лучших результатов.

Таблица 1

Алгоритм	Вариант 1, кВт/час	Вариант 2, кВт/час
PPO	76,310	109,561
A2C	75,408	107,257
DPG	69,010	102,635
DDPG	68,643	86,851

Кроме алгоритмов RL с непрерывным множеством действий для управления энергопотреблением здания были применены алгоритмы глубокого RL с дискретным множеством действий Deep Q-Network (DQN) и Double DQN (DDQN). Для данных алгоритмов была исследована зависимость суточного энергопотребления от степени дискретности действий агентов. При этом весь диапазон изменения действий агентов разбивался на заданное число интервалов, равное коэффициенту дискретности. Значение дискретности были взяты равными 4, 10, 20. Исследование проводилось для способа непосредственного изменения действия агента учителем. Результаты исследования приведены в табл. 2.

Таблица 2

Дискретность	DQN, кВт/час	DDQN, кВт/час
4	64,713	71,876
10	64,007	69,068
20	64,725	62,600

Анализ данных из табл. 2 показывает, что для алгоритма DQN изменение дискретности слабо влияет на суточное энергопотребление. Для алгоритма DDQN при увеличении дискретности суточное энергопотребление немного снижается. Для обоих алгоритмов значение энергопотребления, при дискретности равной 20, немного ниже, чем для алгоритмов с непрерывным множеством действий. Это говорит о небольшом преимуществе использования алгоритмов глубокого RL с дискретным множеством действий для управления энергопотреблением зданий.

Заключение

В работе рассмотрена задача управления энергообъектом (зданием) с применением алгоритмов RL. Для решения проблемы длительного обучения агента RL авторы предложили подход, заключающийся в использовании внешнего учителя. В работе описаны различные способы применения внешнего учителя и рассмотрена постановка задачи управления энергопо-

треблением зданий, допускающая использование популярных алгоритмов RL с внешним учителем. Результаты обучения агентов RL на виртуальной среде показали, что для всех алгоритмов RL применение внешнего учителя позволяет достичь гораздо меньшего энергопотребления за фиксированное время чем для подобных алгоритмов без внешнего учителя. Таким образом, использование алгоритмов RL с внешним учителем может как значительно ускорить обучение агентов RL, так и повысить эффективность управления на начальном этапе обучения.

В работе также произведено сравнение эффективности применения различных алгоритмов RL, в том числе алгоритмов RL с непрерывным и дискретным множеством действий. Результаты сравнения не выявили значительного преимущества использования алгоритмов глубокого RL с дискретным множеством действий (при дискретности равной четырем и более) перед алгоритмами RL с непрерывным множеством действий.

Список литературы

- [Марьясин и др., 2017] Марьясин О.Ю., Колодкина А.С. Управление тепловым режимом зданий с использованием прогнозирующих моделей // Вестник СамГТУ. – 2017. – № 1 (53). – С. 122-132.
- [Afram et al., 2014] Afram A., Janabi-Sharifi F. Theory and applications of HVAC control systems – A review of model predictive control (MPC) // Building and Environment. – 2014. – Vol. 72. – P. 343-355.
- [Argerich et al., 2020] Argerich M.F., Furst J., Cheng B. Tutor4rl: Guiding reinforcement learning with external knowledge // Proc. AAAI Spring Symposium Combining Machine Learning with Knowledge Engineering. – 2020. – P. 1-4.
- [Dyvia et al., 2021] Dyvia H.A., Arif C. Analysis of thermal comfort with predicted mean vote (PMV) index using artificial neural network // IOP Conference Series: Earth and Environmental Science. – 2021. – Vol. 622. – P. 1-12.
- [Graesser et al., 2019] Graesser L., Keng W.L. Foundations of Deep Reinforcement Learning. – Addison-Wesley, 2019.
- [Hossain et al., 2023] Hossain J., Kadir A.F.A., Hanafi A.N., Shareef H., Khatib T., Baharin K.A., Sulaima M.F. A Review on Optimal Energy Management in Commercial Buildings // Energies. – 2023. – Vol. 16. – P. 1-40.
- [Jia et al., 2019] Jia R., Jin M., Sun K., Hong T., Spanos C. Advanced Building Control via Deep Reinforcement Learning // Energy Procedia. – 2019. – Vol. 158. – P. 6158-6163. – doi: <https://doi.org/10.1016/j.egypro.2019.01.494>.
- [Maryasin et al., 2023] Maryasin O.Yu., Plohotnyuk A. Reinforcement Learning-Based Approach to Optimization of Energy Consumption in a Building // Proc. 5th International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency. – 2023. – P. 685-690.
- [Perera et al., 2021] Perera A.T.D., Kamalaruban P. Applications of reinforcement learning in energy systems // Renewable and Sustainable Energy Reviews. – 2021. – Vol. 137. – P. 1-22. – doi: <https://doi.org/10.1016/j.rser.2020.110618>.

- [Shenfeld et al., 2023]** Shenfeld I., Hong Z., Tamar A., Agrawal P. TGRL: An Algorithm for Teacher Guided Reinforcement Learning // Proc. 40th International Conference on Machine Learning. – 2023. – P. 1-18.
- [Sierla et al., 2022]** Sierla S., Ihasalo H., Vyatkin V. A Review of Reinforcement Learning Applications to Control of Heating, Ventilation and Air Conditioning Systems // Energies. – 2022. – Vol. 15. – P. 1-25.
- [Wang et al., 2023]** Wang M., Willes J., Jiralerspong T., Moezzi M. A Comparison of Classical and Deep Reinforcement Learning Methods for HVAC Control // arXiv preprint arXiv:2308.05711. – 2023. – P. 1-7.
- [Yu et al., 2021]** Yu L., Qin S., Zhang M., Shen C., Jiang T., Guan X. A Review of Deep Reinforcement Learning for Smart Building Energy Management // IEEE Internet of Things Journal. – 2021. – Vol. 8(15). – P. 12046-12063.
- [Zimmer et al., 2014]** Zimmer M., Viappiani P., Weng P. Teacher-Student Framework: A Reinforcement Learning Approach // Proc. AAMAS Workshop Autonomous Robots and Multirobot Systems. – 2014. – P. 1-17.

УДК 519.712.7, 519.711.2, 519.651
doi: 10.15622/rcai.2025.061

АППРОКСИМАЦИЯ ГРАНИЦЫ МНОЖЕСТВА ДОСТИЖИМОСТИ МЕТОДАМИ МАШИННОГО ОБУЧЕНИЯ

А.А. Петров (*aleksand.a.petrov@yandex.ru*)^А

В.В. Латонов (*WLatonov@gmail.com*)^В

^А Московский государственный университет им. М.В. Ломоносова,
Москва

^В ПАО «Сбербанк»

В данной работе решается задача оценки множества достижимости динамической системы при помощи методов машинного обучения и нейросетей. Этими средствами возможно решить такую задачу быстрее, чем классическими численными методами, такими, как пиксельный метод. В работе приведены два метода аппроксимации границы множества достижимости: метод опорных векторов (SVM) и полносвязная нейросеть с одним скрытым слоем, которые сравниваются с пиксельным методом. Приведена математическая модель «машина Дубинса» в различных постановках, которая используется в качестве примера поиска множества достижимости. Описаны метод генерации данных с помощью метода Чебышева и схемы формирования обучающих выборок. На основе численных экспериментов проведен анализ точности и быстродействия методов, их поведения в окрестности границы и чувствительности к способу формирования выборки. Итоговый рассмотренный нейросетевой метод обеспечил точность в 93% относительно пиксельного метода, при этом увеличив скорость работы более, чем в 3000 раз, что говорит о применимости нейросетевых моделей для оценки множества достижимости в реальном времени (например, для спутников или автономных систем).

Ключевые слова: множество достижимости; машина Дубинса; аппроксимация границы; пиксельный метод; метод опорных векторов; нейросети; дифференциальные нейросети.

Введение

Задача оценки множества достижимости (reachability) динамической системы заключается в определении всех состояний, которые могут быть достигнуты из заданного начального состояния при всех допустимых управлениях за ограниченный ресурс (например, за заданное время или при ограничении на длину пути). Множество достижимых состояний (часто рассматривается его граница) является важнейшим понятием в теории управления, робототехнике и анализе безопасности, поскольку позволяет ответить на вопросы о том, какие области пространства состоятельны для движения системы, а какие *не* могут быть достигнуты (например, из-за ограничений на управление). Оценка множества достижимости находит применение в синтезе безопасных траекторий движения автономных устройств, в проверке корректности работы систем управления, при анализе столкновений и в реальном времени может выступать инструментом для принятия решений об ограничениях управления.

Современные подходы к задаче достижимости (reachability) все чаще делают акцент на *аппроксимации* множества достижимости, жертвуя строгой точностью ради существенного выигрыша в скорости вычисления. Одним из перспективных направлений является использование методов *машинного обучения* и классификации. Идея заключается в том, чтобы рассматривать задачу принадлежности точки пространству достижимых состояний как задачу бинарной классификации, которую можно решить с помощью модели, обученной на заранее сгенерированных данных. Однако в большинстве случаев всё ещё используются классические методы решения, несмотря на запрос по увеличению скорости обработки, так как методы, подобные описанному выше, не дают достаточных гарантий, что критично в большинстве задач управления. Таким образом, остаётся потребность в разработке подходов, которые могли бы совместить преимущества машинного обучения и гарантировать достаточную надежность и стабильность решений, что и определяет актуальность данной работы. В представленной работе решена задача быстрого поиска множества достижимости, которая была освещена в упрощённом виде в работе [Латов, 2018], где быстрый подсчёт множества достижимости был достигнут за счёт геометрических свойств системы, что бывает крайне редко.

Обзор литературы

Формальное введение множества достижимости и основные подходы к его определению рассмотрены, например, в [Maler, 2008]. Классические методы вычисления таких множеств включают уравнения Гамильтона–Якоби и перебор траекторий, но они требуют значительных ресурсов. Поэтому разрабатываются приближённые алгоритмы. В [Лотов, 1998] пред-

ложен дискретный подход (метод достижимых целей), в котором граница множества аппроксимируется на сетке с последовательным уточнением. В [Горнов, 2015] предложен пиксельный метод с кусочно-линейной аппроксимацией границы, подходящий для систем средней размерности, но теряющий эффективность при росте размерности из-за увеличения числа граней.

Современные исследования направлены на ускорение вычислений в задачах реального времени. В [Allen, 2017] применяются методы машинного обучения для быстрого приближения достижимых множеств, что полезно для онлайн-планирования. В [Bui, 2021] разработана FPGA-реализация алгоритма решения уравнений Гамильтона–Якоби, позволяющая вычислять 4D-множества с частотой до 4 Гц. В [Holmes, 2020] метод ARMTD использует заранее рассчитанные множества для безопасной онлайн-оптимизации траекторий манипулятора. Для низкоразгонных космических аппаратов в [Bowerfind, 2024] предложен семплинг-алгоритм с использованием оптимального управления, который ускоряет расчёт множеств и хорошо согласуется с динамикой системы. В [Shao, 2022] рассмотрена задача относительного движения спутника; предлагается полиэдральное приближение множества достижимости с учётом ограничений управления.

Нейросети активно применяются для компенсации неопределённостей и наблюдения. В [Cheng, 2021] описан адаптивный контроллер, использующий онлайн-оценку нейросетью для компенсации нелинейностей, обеспечивая точность и устойчивость. В [Alfaro-Ponce, 2018] предложен наблюдатель на основе дифференциальной нейросети, устойчивый к квантованию выходных данных. В [Hasan, 2023] нейросеть интегрирована в смешанный наблюдатель для оценки состояния автомобиля с экспериментальным улучшением точности. Работа [Xiang, 2018] посвящена верификации нейросетей: предложен метод вычисления выходного диапазона многослойного перцептрона, критичный для безопасного применения ИИ в управлении.

Методы аппроксимации границы множества достижимости

После того, как будут подготовлены тренировочные данные (координаты точек и метки их достижимости), можно применять различные методы классификации для аппроксимации границы множества достижимых состояний. В данной работе рассматриваются три подхода.

Пиксельный метод. Пиксельным методом называется алгоритм прямого вычисления множества достижимости на дискретной сетке состояний. Суть метода заключается в следующем: пространство (x, y, θ, v) (или соответствующие подпространства, например, (x, y, θ)) квантуется с заданным шагом по каждой координате, образуя сетку узлов (пикселей в конечномерном пространстве). Затем запускается итеративный процесс

распространения “волны” достижимых состояний от начального состояния. Изначально помечается достижимым начальный узел, соответствующий $(x(0), y(0), \theta(0), v(0))$. Далее на каждом шаге времени Δt (дискретизация времени) для всех узлов, помеченных как достижимые на предыдущем шаге, вычисляются их возможные переходы за Δt при всех дискретных вариантах управления (например, для угловой скорости ω можно перебрать значения $-\omega_{max}, 0, \omega_{max}$; для скорости v – значения 0 или V_{max} , и т.п.). Конечные узлы этих переходов (после интервала Δt) помечаются как достижимые на новом шаге. Процесс повторяется, пока не исчерпан весь горизонт T (то есть выполнено $T/\Delta t$ итераций). В результате получается множество узлов сетки, помеченных достижимыми.

Для повышения точности шаги Δt и пространственный шаг сетки берутся достаточно мелкими. Вычислительная сложность пиксельного метода зависит от размерности пространства и требуемой точности. Она пропорциональна числу рассматриваемых узлов, числу шагов времени и числу дискретных управлений. Если обозначить через N_x, N_y, N_θ, N_v количество квантований по каждой переменной, то число пространственных узлов N_x, N_y, N_θ, N_v (для моделей 1–2 можно считать $N_v = 1$). Число временных шагов $N_t = \frac{T}{\Delta t}$. Число управляющих воздействий N_u – конечное и даёт сложность порядка $O(N_x N_y N_\theta N_v \cdot N_t \cdot N_u)$. Даже для умеренных значений ($N_x = N_y = 200, N_\theta = 36$ (шаг 10°), $N_t = 100, N_u = 6$) $\approx 3 \times 10^{10}$ операций, что крайне много. Его главное достоинство – концептуальная простота и наглядность: по сути, он моделирует “заливку” области. В наших экспериментах результаты пиксельного метода (при очень мелкой сетке) рассматриваются как эталонное множество достижимости, с которым сравниваются остальные методы. Однако для применения в реальном времени такой метод не подходит из-за больших вычислительных затрат.

Методы машинного обучения. В работе применялись два подхода машинного обучения для классификации точек на достижимые и недостижимые: метод опорных векторов (SVM) и двухслойная полносвязная нейронная сеть (MLP).

При использовании SVM точки обучающей выборки классифицировались при помощи радиально-базисного ядра (RBF), выбранного из-за явно нелинейной и изогнутой границы множества достижимости. Обучение SVM заключалось в решении задачи квадратичной оптимизации, в результате которой получались опорные векторы, определяющие разделяющую поверхность.

Для нейросетевого подхода использовалась архитектура MLP с одним скрытым слоем, количество нейронов которого варьировалось от 5 до 20 и подбиралось по результатам на валидационной выборке. Входом сети являлись координаты точек (x, y) , выходом – один нейрон с сигмоидной

активацией, выдающий вероятность принадлежности точки к достижимому классу. Скрытый слой использовал нелинейные функции активации (сигмоидальную, \tanh , а также ReLU). Обучение сети проводилось методом градиентного спуска с оптимизатором Adam на тех же размеченных данных, что и для SVM.

Математическая модель

Модель «машина Дубинса». В качестве тестовой динамической системы выбрана модель «машина Дубинса» [Munts, 2020] – движение в горизонтальной плоскости с ограничением на скорость поворота и скорость движения, описываемая данной системой:

$$\begin{cases} \dot{x}(t) = v(t)\cos\theta(t) \\ \dot{y}(t) = v(t)\sin\theta(t) \\ \dot{\theta}(t) = \omega(t) \\ \dot{v}(t) = a(t) \end{cases},$$

где управление включает угловую скорость $\omega(t)$ и управление ускорением $a(t)$. Для модели задается начальное состояние системы. В данной работе предполагается фиксированное начальное положение $(x(0), y(0)) = (0, 0)$ и начальный курс $\theta(0) = 0$ (ориентация вдоль оси Ox). Множество достижимости определяется как совокупность всех состояний (x, y, θ, v) или (x, y, θ) , которых система может достичь хотя бы при одной допустимой реализации управления за заданный промежуток времени T . В частности, основной интерес представляет проекция этого множества на плоскость (x, y) – зона достижимости в геометрическом пространстве. В дальнейших рассуждениях под *множеством достижимости* будем понимать именно проекцию на плоскость (x, y) за фиксированное время T , поскольку ориентация и скорость конечного состояния не столь важны для геометрической зоны достижимости, хотя необходимо учитывать, что фактическая размерность задачи имеет значение 4 – две координаты, угол движения и модуль скорости.

Ограничения и параметры. В типичных сценариях задаются численные ограничения, например: максимальная угловая скорость ω_{max} (что эквивалентно минимальному радиусу поворота $R_{min} = V/\omega_{max}$ для варианта 2), максимальная скорость V_{max} (для варианта 3), а также горизонт T (время, за которое рассматривается достижимость). В нашем исследовании можно считать, для определенности, что $V_{max} = 1$ (единичная скорость в безразмерных единицах), $\omega_{max} = 1$ (единица угловой скорости, что дает минимальный радиус поворота $R_{min} = 1$ для варианта 2), а горизонт T – некоторый фиксированный промежуток (например, $T = 1$). Все величины можно считать безразмерными после соответствующей нормализации.

Генерация данных. Для применения методов машинного обучения необходимо сформировать выборку примеров состояний, принадлежащих или не принадлежащих множеству достижимости. Проще говоря, требуется множество точек (x, y) , снабженных меткой класса: 1 (достижимо) или 0 (недостижимо) из начального состояния за время T . Поскольку аналитического описания границы множества достижимости для машины Дубинса (в общем случае) не существует, в работе получены *эталонные данные* численным путем. В качестве эталонного способа определения достижимости реализован пиксельный метод.

Схема построения выборок. Получив эталонное множество достижимости (например, в виде списка достижимых узлов сетки или полигона, охватывающего эту область), формируем обучающие выборки для последующей классификации. Рассматриваются два типа выборок:

1. *Равномерная выборка:* точки (x, y) генерируются равномерно по площади в некоторой области, гарантированно содержащей множество достижимости (например, в круге радиуса $V_{max} \cdot T$ вокруг начального положения). Каждой точке присваивается метка 1, если она попала в достижимую область (по данным эталонного пиксельного метода), и 0 – если не попала. Таким образом, формируется сбалансированный набор примеров внутри и вне достижимой зоны.

2. *Выборка с увеличенной плотностью у границы:* в данной работе применялся алгоритм на основе поиска ближайшего соседа – если для точки в пределах радиуса $r_m = 1$ есть точка противоположного класса, то она считается близкой к границе, в противном случае – далёкой от нее. Далее на такой выборке вместо равномерного распределения бралось распределение, где у всех точек, близких к границе, вероятность попасть в итоговую выборку выше, чем у далёких от границы, в $prob = 10$ раз. В итоге получается выборка, где значительно больше точек расположено вблизи разделяющей поверхности «достижимо/недостижимо». Цель такого подхода – дать обучающему алгоритму больше информации о форме границы, поскольку именно там решается задача классификации (точки далеко внутри или вне области тривиальны для классификатора).

Последовательность шагов, необходимых для генерации и применения выборки, визуально представлена на рис. 1.



Рис. 1. Визуальная блок-схема составления выборки и применения методов

Численные эксперименты

И метод опорных векторов, и нейросеть обучается на данных, полученных пиксельным методом. То есть, они не вычисляют множество достижимости *с нуля*, а именно аппроксимируют уже известное (приближенно) множество. Тем не менее, даже с учетом затрат на предварительную генерацию данных, выигрыш оказывается существенным: данные можно сгенерировать один раз офлайн (например, на этапе проектирования системы), а в процессе работы эксплуатации использовать легковесную модель для классификации новых ситуаций или применять пиксельный метод локального для дообучения моделей.

Для сравнения описанных методов проведены численные эксперименты на модели «машина Дубинса». Эталонное множество достижимости за время T получено пиксельным методом на сетке с шагом 0.01 по (x, y) и 1° по углу θ .

Для обучения моделей SVM и нейросети сгенерировано $\sim 10,000$ точек в квадрате $[-1.2, 1.2]$. Примерно половина точек выбрана равномерно, половина – с упором к границе. Из них 60% использовано для обучения, остальные для контроля и теста (по 20%). SVM настроен с RBF-ядром, параметр ширины ядра и регуляризации подобраны по кросс-валидации на контроле. Нейросеть взята с 10 нейронами скрытого слоя, сигмоидной активацией, обучена на 500 эпох (до сходимости ошибки).

Эти гиперпараметры моделей подбирались методом перебора по сетке (grid search) на валидационной выборке. Реализация моделей выполнена с использованием классической библиотеки Python: Scikit-learn (`sklearn.svm.SVC` с RBF-ядром и `sklearn.neural_network.MLPClassifier` соответствен-

но). Архитектура нейросети выбрана максимально простой (один скрытый слой с 10 нейронами) – это минимально достаточная конфигурация, которая, как показали эксперименты, обеспечивает требуемое качество решения задачи на базовом уровне.

Таблица 1

Сравнение точности классификации и времени вычисления
для разных методов

Т	Выборка	Метод	Точность	Время
15с	Равномерная	SVM	67%	0.001с
15с	Смещённая (R=1, prob=x10)	SVM	72%	0.001с
15с	Равномерная	NN	93%	0.017с
15с	Смещённая (R=1, prob=x10)	NN	91%	0.01с
15с	Равномерная	Пиксельный	*100%	29.8с
15с	Смещённая (R=1, prob=x10)	Пиксельный	*100%	35.1с

Для оценки качества классификации использовалась метрика ассигасу, понимаемая как доля правильно классифицированных точек тестовой выборки. В случае нейросетевой модели для равномерной выборки эта доля составила около 93%. Соответствующая матрица ошибок показывает, что ~30% всех тестовых точек были правильно определены как достижимые (True Positive), ~60% – правильно как недостижимые (True Negative), около 4% оказались ложно недостижимыми (False Negative) и ~3% – ложно достижимыми (False Positive). Эти ошибки распределены относительно равномерно и приходятся в основном на точки, лежащие вблизи истинной границы множества достижимости. Подобное явление типично для задач классификации данного типа, поскольку в окрестности границы даже незначительные отклонения условий могут изменить метку достижимости. Для более явного анализа и интерпретации граничных элементов применимы методы вербального анализа решений; в системе ДИФКЛАСС [Ларичев, 1996] задаётся процедура построения непротиворечивых правил дифференциальной классификации, удобная для проверки устойчивости разметки у границы.

В табл. 1 представлены сводные показатели для трех методов: пиксельного (эталон), SVM и нейросети. Точность классификации оценивалась на независимых тестовых данных и характеризует, какой процент точек правильно определен как достижимый/недостижимый (по сравнению с эталоном). Пиксельный метод потребовал порядка нескольких минут вычислений для построения границы. В то же время, проверка на принадлежность множеству достижимости методом SVM заняла $\sim 10^{-3}$ с (1 мс) на CPU, а нейросетью – $\sim 10^{-2}$ с (10 мс). Иными словами, SVM позволяет получать ответ практически мгновенно, нейросеть – очень быстро

в масштабе реального времени, а вот пиксельный метод непригоден для онлайн-применения (его можно использовать только офлайн или в ограниченном объёме данных). На рис. 1 и 2 представлены примеры разметок тестовой выборки для рассмотренных случаев.

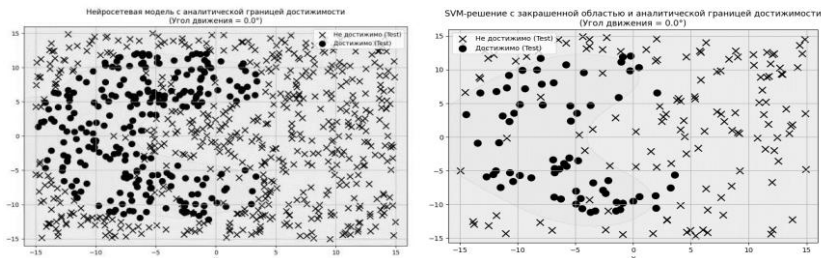


Рис. 2. Аппроксимация границы множества достижимости.
Метод формирования выборки равномерный

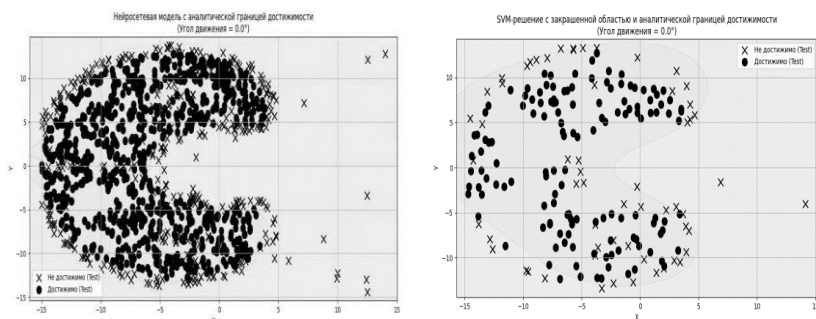


Рис. 3. Аппроксимация границы множества достижимости.
Метод формирования выборки вблизи границы

Выводы

Полученные результаты показывают, что методы машинного обучения способны успешно решать задачу аппроксимации множества достижимости. Основная часть ошибок обеих моделей сконцентрирована вблизи границы области – что естественно, поскольку именно там данные наиболее «сложные»: небольшая погрешность может изменить метку точки. Внутри области и далеко снаружи и SVM, и MLP почти безошибочно (что подтверждается близкой к 100% полнотой/точностью в этих зонах).

В дополнение к базовому сценарию были проведены эксперименты и с другими значениями параметров для нейросетевой модели – в частности, с различными горизонтами времени T и максимальными углами поворота

ω_{max} . В табл. 2 приведены показатели качества модели при нескольких таких комбинациях параметров. Видно, что точность классификации остается практически неизменной при варьировании T и ω_{max} . Однако обнаружено, что модель, обученная при одних параметрах, показывает существенно низкую точность, если применять её напрямую к данным, сгенерированным при других условиях. В то же время при дообучении предварительно обученной сети под новые параметры для достижения аналогичного качества требуется примерно вдвое меньше итераций обучения, чем при обучении «с нуля». Этот результат указывает на потенциал методов трансферного дообучения и обучения с подкреплением для эффективной адаптации модели к изменяющимся условиям.

Таблица 2

Результаты модели на различных горизонтах прогноза T и ограничениях максимального угла поворота

Горизонт T , сек.	Макс. угол поворота, °	Точность, %
5	30	87
15	30	93
15	60	91
30	60	90

Заключение

Результаты исследования показывают, что методы машинного обучения, в частности SVM и относительно простая нейросетевая модель, эффективно аппроксимируют границу множества достижимости для динамической системы с дифференциальными ограничениями. Нейросетевые модели оказались применимы для оценки достижимости с высокой скоростью и достаточной точностью. Таким образом, новизна данного исследования заключается в том, что, несмотря на потенциальную непредсказуемость нейросетевых моделей, они продемонстрировали одновременно высокое быстродействие и приемлемую точность в задаче аппроксимации границы множества достижимости.

Важным преимуществом подхода является возможность применения в реальном времени за счёт быстродействия разработанного классификатора. Например, автономный дрон или спутник могут принимать решения о допустимых маневрах или реагировать на динамические ограничения при помощи этого классификатора.

Однако следует учитывать и ограничения предложенного подхода. Модель (SVM или нейросеть) обучается на определенных условиях (динамика, ограничения, горизонт времени). Если условия изменятся (на-

пример, изменится ресурс времени или динамические параметры), потребуется либо повторная генерация данных и обучение, либо использование более универсальной модели.

Перспективные направления дальнейших исследований включают адаптацию нейросетей к разным условиям и режимам работы, включающим большее число параметров. Также актуальна переносимость моделей, которую можно обеспечить с помощью трансферного обучения. Ещё одно перспективное направление – интеграция дифференциальных нейросетей, описывающих динамику систем напрямую. Вместо последовательных этапов генерации траекторий и классификации достижимых состояний возможна разработка единой модели, воспроизводящей эволюцию множества достижимости.

Список литературы

- Латонов В.В.** Программные стратегии тестирования качества управления линией визирования по видеоизображению // Вестник Московского университета. Серия 1: Математика. Механика. – 2018. – № 6. – С. 51-56.
- Maler O.** Computing Reachable Sets: An Introduction // Hybrid Systems: Computation and Control, LNCS 4981, 2008. – P. 65-76.
- Лотов А.В., Бушенков В.А., Каменев Г.К., Черных О.Л.** Компьютер и поиск компромисса. Метод достижимых целей. – М.: Наука, 1997. – 239 с.
- Горнов А.Ю., Финкельштейн Е.А.** Алгоритм кусочно-линейной аппроксимации границы множества достижимости управляемой системы // Автоматика и телемеханика. – 2015. – № 3. – С. 22-35.
- Allen R.E., Clark A.A., Starek J.A., Pavone M.A.** Machine Learning Approach for Real-Time Reachability Analysis // Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2017.
- Bui M., Lu M., Hojabr R., Chen M., Shriraman A.** Real-Time Hamilton-Jacobi Reachability Analysis of an Autonomous System with an FPGA // Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2021.
- Holmes P., Kousik S., Zhang B., Raz D., Barbalata C., Johnson-Roberson M., Vasudevan R.** Reachable Sets for Safe, Real-Time Manipulator Trajectory Design // Proceedings of Robotics: Science and Systems (RSS), 2020.
- Bowerfind S., Taheri E.** Rapid Approximation of Low-Thrust Spacecraft Reachable Sets within Complex Two-Body and Cislunar Dynamics // Aerospace. – 2024. – 11(5): 380.
- Shao L., Miao H., Hu R., Liu H.** Reachable set estimation for spacecraft relative motion based on bang-bang principle // Chinese Journal of Aeronautics. – 2022. – 36(2). – P. 229-240.
- Cheng L., Wang Z., Jiang F., Li J.** Adaptive neural network control of nonlinear systems with unknown dynamics // Advances in Space Research. – 2021. – 67(3).
- Alfaro-Ponce M., Argüelles A., Chairez I., Avelar A.** Differential Neural Networks Observer for Second Order Systems with Sampled and Quantized Output // IFAC-PapersOnLine. – 2018. – 51(13). – P. 490-495.

- Hasan A.-G., Laghmara H., Ali S. A., Ainouz S., Xing G.** Continuous–Discrete Time Neural Network Observer for Nonlinear Dynamic Systems (Application to Vehicle Systems). – IFAC-PapersOnLine, 2023. – 56(2).
- Xiang W., Tran H.-D., Johnson T.T.** Output Reachable Set Estimation and Verification for Multi-Layer Neural Networks // IEEE Transactions on Neural Networks and Learning Systems. – 2018. – 29(11).
- Munts N.V.** Numerical Study of Different Variants of Dubins' Car Model // Proceedings of the 60th Israel Annual Conference on Aerospace Sciences. – 2020.
- Ларичев О.И., Болотов А.А.** Система ДИФКЛАСС: построение полных и непротиворечивых баз экспертных знаний в задачах дифференциальной классификации // Науч.-техн. информация. Сер. 2: Информационные процессы и системы. – 1996. – № 9. – С. 9-15.

ИНТЕЛЛЕКТУАЛЬНАЯ ПОДДЕРЖКА СИТУАЦИОННЫХ РЕШЕНИЙ НА ОСНОВЕ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ¹

А.П. Шапкин (*shapkinap1999@gmail.com*)

В.В. Борисов (*vbor67@mail.ru*)

Филиал Национального исследовательского университета «МЭИ»
в г. Смоленске, Смоленск

В статье решается задача интеллектуальной поддержки ситуационных решений на основе предложенной разновидности адаптивных нечетких ситуационных моделей (АНСМ) и их обучения с подкреплением. Выполнено обоснование подхода и предложен способ оценки управляющих решений в АНСМ на основе нейросети *Deep Recurrent Q-Network*, позволяющей учитывать ретроспективу принятых ранее управляющих решений, а также результаты структурно-параметрической настройки АНСМ. Приведен пример интеллектуальной поддержки ситуационных решений при управлении энергетической мини-сетью на основе предложенной разновидности АНСМ и обучения с подкреплением.

Ключевые слова: адаптивные нечеткие ситуационные сети, интеллектуальная поддержка ситуационных решений, обучение с подкреплением.

Введение

Нечеткий ситуационный подход эффективно применим для поддержки принятия решений в условиях неполноты и неопределенности данных, с учетом выполнения требований и ограничений при переходе через промежуточные ситуации для достижения целевой ситуации [Мелихов и др., 1990]. Одной из основных особенностей этого подхода является формирование модели рассматриваемой проблемы, совмещенной с моделью управления ею в виде совокупности управляющих решений для достижения целевой ситуации [Борисов и др., 2021].

¹ Работа выполнена в рамках государственного задания (проект № FSWF-2023-0012).

В работе [Мелихов и др., 1990] предложены нечеткие ситуационные сети, состоящие из нечетких ситуаций и дуг, соответствующих переходам между ситуациями при воздействии управляющих решений на соответствующие ситуационные признаки.

В статье [Борисов и др., 2009] предложены нечеткие ситуационно-событийные модели, учитывающие неопределенность воздействия управляющих решений на ситуации за счет использования лингвистических лотерей, а также продолжительность выполнения решений.

Развитием этих моделей являются адаптивные нечеткие ситуационные модели (АНСМ), предложенные способы адаптации которых позволяют изменять их структуру и параметры в зависимости от изменения системных и внешних факторов, стратегий принятия решений и ограничений [Denisenkov et al., 2018.].

Однако определение наилучшей последовательности управляющих решений для перехода из текущей в целевую ситуацию и выбор следующей нечеткой ситуации в соответствии с заданной стратегией для АНСМ представляет собой нетривиальную задачу в условиях необходимости ее структурно-параметрической настройки.

Подход к обучению с подкреплением создает хорошие предпосылки для решения указанной выше проблемы оценки управляющих решений АНСМ для поиска наилучших последовательностей управляющих решений с максимизацией заданной функции полезности с учетом выбранной стратегии и постоянно накапливаемого опыта.

В статье предлагается способ оценки управляющих решений АНСМ на основе обучения с подкреплением, а также рассматривается пример поддержки ситуационных решений при управлении энергетической минисетью на основе предложенной разновидности АНСМ.

1. Адаптивные нечеткие ситуационные модели

Предложенная в работе [Борисов и др., 2025] разновидность АНСМ и дополненная предлагаемым способом оценки управляющих решений представляется в виде:

$$AFSM = \langle P, S, R, U, A, E \rangle,$$

где P, S, U, R – множества нечетких ситуационных признаков, ситуаций, переходов, управляющих решений, соответственно; A – способ адаптации; E – способ оценки управляющих решений.

Нечеткие ситуации представляются в виде нечетких множеств 2-го уровня на совокупности ситуационных признаков:

$$S = \{\widehat{s}_l\}, \quad l = 1, \dots, L,$$

$$\widehat{s}_l = \{(\mu_s(p_i) / p_i)\}, \quad p_i \in P,$$

$$\mu_s(p_i) = \left\{ \left(\mu_{\mu_s(p_i)}(T_j^i) / T_j^i \right) \right\}, \quad i \in \{1, \dots, I\}, \quad j \in \{1, \dots, J_i\},$$

$$p_i = \langle T_i, D_i \rangle, \quad i \in \{1, \dots, I\},$$

$T_i = \{T_1^i, \dots, T_{m_i}^i\}$ – терм-множество ситуационного признака p_i , m_i – число p_i ; D_i – базовое множество p_i ; термы T_j^i , $j \in \{1, \dots, J_i\}$ задаются нечеткими переменными $\langle T_j^i, D_i, C_j^i \rangle$, то есть нечеткими множествами C_j^i на D_i :

$$C_j^i = \left\{ \left(\mu_{C_j^i}(d) / d \right) \right\}, \quad d \in D_i.$$

Управляющие решения представляются в виде:

$$R_k = \langle Lr_k, Tr_k, Dr_k \rangle, \quad k \in \{1, \dots, K\},$$

где $Lr_k = \{Lr_1^k, Lr_2^k, Lr_3^k\}$ – терм-множество направленности воздействия управляющего решения r_k ; $Tr_k = \{Tr_1^k, \dots, Tr_{L_k}^k\}$ – терм-множество силы его воздействия; Dr_k – шкала силы воздействия r_k , $[-1, 1]$.

Множество переходов $U = \{u_1, u_2, \dots, u_Y\}$ между ситуациями сопоставляется с управляющими решениями с учетом их значимости.

Способ адаптации A предназначен для структурно-параметрической настройки АНСМ на основе обучения с подкреплением.

Способ оценки управляющих решений E предназначен для определения управляющих решений, наиболее подходящих для достижения целевой нечеткой ситуации, в соответствии с выбранной стратегией управления.

2. Обоснование подхода к оценке управляющих решений в АНСМ на основе обучения с подкреплением

Политика обучения с подкреплением для ситуационного подхода определяется как отображение ситуаций в управляющие решения в зависимости от контекста и системы предпочтений.

При решении задачи поиска наилучших управляющих решений в АНСМ с неограниченным количеством ситуаций и управляющих решений традиционные алгоритмы обучения с подкреплением, такие как *Q-learning* или *Deep Q-Learning (DQN)*, сталкиваются с принципиальными ограничениями. Так, алгоритм *Q-learning* не масштабируется на многочисленные признаковые пространства из-за необходимости хранения *Q*-таблиц, а алгоритм DQN, хоть и использует нейросетевую аппроксимацию, не учитывает зависимости между смежными ситуациями [Шапкин и др., 2025].

Deep Recurrent Q-Network (DRQN) сочетает преимущества глубокого обучения с механизмом рекуррентных нейросетей, что позволяет эффективно обрабатывать частично наблюдаемые признаковое пространство и учитывать ретроспективу взаимодействия системных и внешних факторов [Zhu et al., 2018], [Moreno-Vera, 2019]. Достоинства применения *DRQN* для оценки управляющих решений в АНСМ заключаются в его способности учитывать историю переходов между нечеткими ситуациями. Помимо этого, *DRQN* позволяет учитывать новые ситуации при адаптации АНСМ, за счет кодирования особенностей АНСМ в скрытых состояниях [Hausknecht et al., 2017].

3. Оценка управляющих решений в АНСМ на основе Deep Recurrent Q-Network

3.1. Способ оценки управляющих решений для АНСМ

Необходимость в оценке управляющих решений для АНСМ возникает при определении наилучшей последовательности управляющих решений для достижения целевой нечеткой ситуации. Поэтому, перед оценкой управляющих решений определяется целевая нечеткая ситуация s_{targ} , например, в виде значений функций принадлежности по каждому признаку нечеткой ситуации.

Для получения оценки управляющих решений применяется глубокая рекуррентная нейросеть E , основанная на *DRQN*.

Предлагаемый способ оценки управляющих решений для АНСМ включает в себя перечисленные этапы.

Этап 1. Идентификация текущей нечеткой ситуации s_{cur} .

Этап 2. Определение возможных управляющих решений для нечеткой ситуации s_{cur} . Результатом этого этапа является маска m_{cur} , представляющая собой массив бинарных чисел по числу нечетких ситуаций в АНСМ. Каждой ситуации, в которую возможен переход соответствует «1», а недостижимым ситуациям – «0» [Shengyi et al., 2020].

Этап 3. Формирование массива входных данных для сети E , который включает в себя информацию о текущей нечеткой ситуации s_{cur} и целевой нечеткой ситуации s_{targ} .

Этап 4. Входной массив подается на вход нейросети E .

Этап 5. На выходе нейросети E получается множество оценок управляющих решений, размер которого зависит от количества нечетких ситуаций.

Этап 6. Для исключения оценок несуществующих переходов из текущей нечеткой ситуации s_{cur} , ко множеству оценок управляющих решений применяется маска m_{cur} .

Результатом работы способа является множество оценок управляющих решений, в котором нечеткая ситуация, в которую необходимо совершить переход с учетом выбранной стратегии имеет наивысшее значение оценки.

При необходимости оценки управляющих решений в АНСМ, моделирующих разные сложные системы одного класса, применяется нормализация данных. Информация о весах нечетких переходов не подается на вход явным образом, а кодируется через функцию вознаграждения.

3.2. Обучение нейросети, основанной на Deep Recurrent Q-Network

Ключевое преимущество применения *DRQN* для оценки управляющих решений заключается в его способности интегрировать особенности заданной стратегии управления непосредственно в процесс обучения. Это достигается через соответствующую параметризацию функции вознаграждения $R(s, a, s')$, которая кодирует критерии оптимальности пути (минимизацию количества переходов, избегание нежелательных ситуаций). В результате обученная Q -функция $Q(s, a; \theta)$ позволяет получить оценку действий, которые отражают долгосрочные последствия решений в контексте выбранной стратегии управления [AlMahamid et al., 2022]. Благодаря этому свойству, в каждой ситуации s_t выбирается наилучшее управляющее действие a_t , позволяющее достичь целевую ситуацию s_T .

Дополнительно для обеспечения устойчивости обучения используется целевая сеть – нейросеть с идентичной структурой основной нейросети *DRQN*, но отличающейся частотой обновления параметров.

Периодическое обновление параметров целевой нейросети θ^- позволяет стабилизировать обучение, уменьшая корреляцию между текущими Q -значениями и целевыми. Интервал обновления подбирается эмпирически: слишком частые обновления приводят к нестабильности, а слишком редкие – к замедлению обучения.

В контексте ситуационного подхода политика, полученная в результате обучения, представляет собой систему предпочтений, отражающую стратегию управления. Формально политика $\pi(a|s)$ определяет вероятность выбора действия a в ситуации s . В случае ϵ -жадной стратегии с вероятностью $(1 - \epsilon)$ выбирается действие с максимальным Q -значением, а с вероятностью ϵ – исследует альтернативные варианты.

Процесс обучения начинается с инициализации весов *DRQN*. Входом сети является текущая ситуация s_t , представленная в виде множества признаков, а выходом – Q -значения для всех возможных управляющих решений $a \in A(s_t)$.

При последовательном переходе из одной нечеткой ситуации в другую формируется награда r_t за каждый такой переход. Награда определяется в соответствии со стратегией управления: например, если стратегия требует минимизации суммарного веса переходов, то награда может быть определена как отрицательный вес нечеткого перехода. В случае, если переход в определенную ситуацию нежелателен при выбранной стратегии, то может быть добавлен значительный штраф при переходе в эту нечеткую ситуацию.

Рекуррентный слой нейросети *DRQN* позволяет учитывать не только текущую ситуацию, но и контекст предыдущих переходов, что особенно важно для работы АНСМ. На каждом шаге скрытое состояние h_t обновляется в соответствии с выбранным действием и наблюдением новой ситуации, формируя представление, которое влияет на принятие решений.

Обучение происходит путем минимизации функции потерь:

$$L(\theta) = E_{(s,a,r,s') \sim D} \left(\left(r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s, a; \theta) \right)^2 \right),$$

где θ – параметры обучаемой нейросети, θ^- – параметры целевой нейросети, обновляемые периодически, γ – коэффициент дисконтирования, а D – база воспроизведения опыта.

Во время обучения из D случайным образом выбирается последовательность управляющих решений, затем для каждого из них, используя целевую нейросеть, вычисляется целевое Q -значение. Используя градиент ошибки для выбранной последовательности управляющих решений обновляются параметры основной нейросети.

4. Пример поддержки ситуационных решений с использованием АНСМ на основе обучения с подкреплением

Рассмотрим пример оценивания управляющих решений на примере АНСМ, предназначенной для моделирования процесса управления энергетической мини-сетью [Борисов и др., 2025]. Определим множество признаков $P = \{p_1, p_2, p_3, p_4\}$, где p_1 – уровень заряда аккумуляторов, p_2 – производство энергии солнечными панелями, p_3 – потребление энергии, p_4 – количество топлива резервного генератора.

Определим терм-множества ситуационных признаков и соответствующие им базовые множества:

$$\begin{aligned} T_{p1} &= \{T_1^{p1} = \text{«Низкий»}, T_2^{p1} = \text{«Средний»}, T_3^{p1} = \text{«Высокий»}\}, D_{p1} = [0, 40]. \\ T_{p2} &= \{T_1^{p2} = \text{«Низкое»}, T_2^{p2} = \text{«Среднее»}, T_3^{p2} = \text{«Высокое»}\}, D_{p2} = [0, 100]. \\ T_{p3} &= \{T_1^{p3} = \text{«Низкое»}, T_2^{p3} = \text{«Среднее»}, T_3^{p3} = \text{«Высокое»}\}, D_{p3} = [0, 100]. \\ T_{p4} &= \{T_1^{p4} = \text{«Низкое»}, T_2^{p4} = \text{«Среднее»}, T_3^{p4} = \text{«Высокое»}\}, D_{p3} = [0, 700]. \end{aligned}$$

Нечеткие ситуации задаются с учетом соответствия степеней принадлежности значениям признаков. Структура АНСМ для рассматриваемого примера представлена на рис. 1.

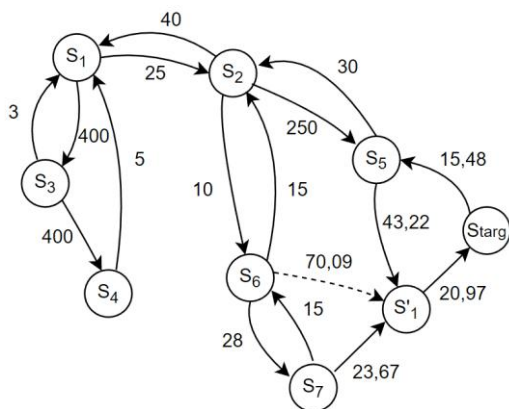


Рис. 1. АНСМ для моделирования процесса управления энергетической мини-сетью

Пусть энергетическая мини-сеть находится в нечеткой ситуации s_6 , а до этого – в нечеткой ситуации s_2 . Стоит задача ее перехода в нечеткую ситуацию s_{targ} . Выбрана стратегия минимизации стоимости соответствующих управляющих решений. Однако переход через некоторые ситуации является нежелательным. Например, непосредственный переход в нечеткую ситуацию s'_1 является более «дорогим», чем переход в эту ситуацию через нечеткую ситуацию s_7 , однако, ситуация s_7 характеризуется повышенными эксплуатационными рисками.

Массив исходных данных для нейросети *DRQN* формируется из описания целевой нечеткой ситуации, которое формируется по значениям функций принадлежности признаков нечеткой ситуации:

$$s_{targ} = (0,9; 0,2; 0,4; \dots; 0,2)$$

и описания текущей нечеткой ситуации:

$$s_6 = (0,2; 0,4; 0,7; \dots; 0,9) .$$

После обработки массива исходных данных, нейросеть формирует значения оценок для всех управляющих решений АНСМ, к которому применяется маска доступных нечетких переходов из нечеткой ситуации s_6 :

$$M_{s_6} = (0; 1; 0; 0; 0; 0; 1; 1; 0) .$$

Массив итоговых значений оценки выглядит следующим образом:

$$Q(s_6) = (0; 3,4; 0; 0; 0; 12,7; 15,8; 0) .$$

Таким образом, оценка управляющего решения, приводящего к переходу в нечеткое состояние s'_1 выше, чем у управляющего решения, приводящего к переходу в нечеткое состояние s_7 . Такая оценка соответствует выбранной стратегии управления.

Заключение

Применение обучения с подкреплением для интеллектуальной поддержки ситуационных решений является перспективным направлением развития теории и практики принятия решений.

В статье поставлена и решается задача интеллектуальной поддержки ситуационных решений на основе предложенной разновидности АНСМ и их обучения с подкреплением.

Выполнено обоснование подхода и предложен способ оценки управляющих решений в АНСМ на основе нейросети *DRQN*, позволяющего учитывать ретроспективу принятых ранее управляющих решений и взаимодействия системных и внешних факторов, а также результаты динамической структурно-параметрической настройки АНСМ, обеспечивать выбор наилучших управляющих решений в соответствии с заданной стратегией.

Приведен пример интеллектуальной поддержки ситуационных решений при управлении энергетической мини-сетью на основе предложенной разновидности АНСМ и обучения с подкреплением.

Дальнейшие исследования направлены на расширение класса решаемых задач с использованием предложенной разновидности АНСМ, включая многокритериальную оптимизацию и интеллектуальную поддержку ситуационных решений в мультиагентных системах.

Список литературы

- [Борисов и др., 2009] Борисов В.В., Зернов М.М. Реализация ситуационного подхода на основе нечеткой иерархической ситуационно-событийной сети // Искусственный интеллект и принятие решений. – 2009. – № 1. – С. 17-30.
- [Борисов и др., 2021] Борисов В.В., Авраменко Д.Ю. Нечеткое ситуационное управление сложными системами на основе их композиционного гибридного моделирования // Системы управления, связи и безопасности. – 2021. – № 3. – С. 207-237.
- [Борисов и др., 2025] Борисов В.В., Шапкин А.П. Нечеткие ситуационные сети с адаптацией на основе обучения с подкреплением // Вестник Московского энергетического института. – 2025. – № 3. – С. 135-143.
- [Мелихов и др., 1990] Мелихов А.Н., Берштейн Л.С., Коровин С.Я. Ситуационные советующие системы с нечеткой логикой. – М.: Наука, 1990. – 272 с.
- [Шапкин и др., 2025] Шапкин А.П., Борисов В.В. Способы внедрения обучения с подкреплением в адаптивные нечеткие ситуационные сети // XXII Международная научно-техническая конференция студентов и аспирантов (Смоленск, 23–24 апреля 2025 г.): Труды конференции. В 2-х т. Т. 1. – Смоленск: Универсум, 2025. – С. 470-474.

- [AlMahamid et al., 2022] AlMahamid F., Grolinger K. Reinforcement Learning Algorithms: An Overview and Classification [Электронный ресурс] // arXiv.org. – 2022. URL: <https://arxiv.org/abs/2209.14940> (дата обращения: 16.03.2024).
- [Denisenkov et al., 2018] Denisenkov M.A., Borisov V.V. Modeling the behavior of intelligent agents based on adaptive fuzzy situational networks // Proc. 3th Russian-Pacific Conference on Computer Technology and Application (RPC). Vladivostok, Russia, 18-25 Aug. 2018. – DOI: 10.1109/RPC.2018.8482217.
- [Hausknecht et al., 2017] Hausknecht M., Stone P. Deep Recurrent Q-Learning for Partially Observable MDPs // arXiv.org. – 2017. – URL: <https://arxiv.org/abs/1507.06527> (дата обращения: 16.03.2024).
- [Moreno-Vera, 2019] Moreno-Vera F. Performing Deep Recurrent Double Q-Learning for Atari Games [Электронный ресурс] // arXiv.org. – 2019. – URL: <https://arxiv.org/abs/1908.06040> (дата обращения: 16.03.2024).
- [Shengyi et al., 2020] Shengyi H., Ontañón S. A Closer Look at Invalid Action Masking in Policy Gradient Algorithms [Электронный ресурс] // arXiv.org. – 2020. – URL: <https://arxiv.org/abs/2006.14171> (дата обращения: 16.03.2024).
- [Zhu et al., 2018] Zhu P., Li X., Poupart P., Miao G. On Improving Deep Reinforcement Learning for POMDPs [Электронный ресурс] // arXiv.org. – 2018. – URL: <https://arxiv.org/abs/1804.06309> (дата обращения: 16.03.2024).

ПОДДЕРЖКА ПРИНЯТИЯ РЕШЕНИЙ ПРИ ПРЕРВАННОЙ ПОСАДКЕ С ИСПОЛЬЗОВАНИЕМ МЕТОДА ВЫВОДА НА ОСНОВЕ ПРЕЦЕДЕНТОВ

Б.Е. Федунцов (*boris_fed@gosniias.ru*)^A
А.А. Липатов (*alexey.lipatov75@mail.ru*)^A

^A ФАУ «ГосНИИАС», Москва

В докладе представлены фрагменты базы знаний бортовой оперативно советующей экспертной системы, предназначенной для поддержки принятия решений экипажем самолёта в ситуации прерванной посадки. Система должна выдавать рекомендации об уходе на второй круг и повторной попытке посадки или уходе на запасной аэродром на основании анализа причин, приведших к прерыванию посадки. Содержимое базы знаний формируется за счёт накопления опыта успешного принятия и реализации подобных решений, формализованного с помощью метода вывода на основе прецедентов. Разработан прототип экспертной системы, в котором для принятия решений используются признаки, характеризующие метеоусловия (скорость попутного ветра), состояние взлётно-посадочной полосы (коэффициент сцепления) и характер движения самолёта при неудачной попытке посадки (установившееся продольное ускорение). Приведён пример работы системы.

Ключевые слова: этап посадки самолёта, поддержка принятия решений, база знаний, процедуры базы знаний, рекомендации экипажу, вывод на основе прецедентов.

Введение

Посадка самолёта является одним из наиболее сложных этапов полёта, связанным с высоким риском авиационных происшествий. Значительную часть среди этих происшествий составляют случаи выкатывания самолётов за пределы взлётно-посадочной полосы (ВПП). В целом, к группе авиационных происшествий, связанных с безопасностью на ВПП, за период с 2016 по 2020 г. относились 53% авиационных происшествий и 41% катастроф, на которые пришлось 20% погибших от общего числа жертв авиационных катастроф [Анализ, 2021]. Существенный вклад в эту группу

происшествий вносят выкатывания. Исследователи отмечают различные причины выкатывания за пределы ВПП [Мозоляко, 2014], которые можно разделить на несколько групп:

- ошибки пилотирования;
- неблагоприятные метеоусловия (сдвиг ветра, попутный ветер, боковой ветер);
- неблагоприятное состояние ВПП (залитая водой, заснеженная, обледенелая), приводящее к снижению коэффициента сцепления колёс самолёта с ВПП;
- снижение эффективности или отказ устройств торможения;
- несвоевременное принятие экипажем решений об уходе на второй круг или запасной аэродром.

Предотвращению выкатываний уделяется большое внимание в рамках деятельности, направленной на повышение безопасности полётов [Государственная политика, 2017, EPAS, 2024]. Для достижения этой цели принимаются меры организационного и технического характера. Организационные меры направлены на исключение предпосылок для выкатывания и главным образом включают в себя повышение уровня подготовки пилотов и совершенствование технологии работы экипажей. Технические меры заключаются в разработке систем, обеспечивающих ситуационную осведомлённость экипажа о положении самолёта на ВПП и предупреждение о возникшей опасности выкатывания, а также бортовых оперативно советующих систем, предоставляющих экипажу рекомендации по действиям в такой ситуации [Мозоляко, 2014], [Бородкин, 2023].

Системы осведомлённости и предупреждения об опасности выкатывания за пределы ВПП – Runway Overrun Alerting and Awareness System (ROAAS) – предназначены для снижения риска выкатывания во время посадки путём обеспечения своевременного и чёткого предупреждения лётного экипажа о нехватке остающейся длины ВПП для остановки самолёта [Georges, 2017]. К данному классу относятся такие системы, как Runway Awareness and Advisory System (RAAS) корпорации Honeywell Aerospace, США [Product Description, 2009], Runway Overrun Prevention System (ROPS) корпорации Airbus [Safety Innovation, 2022], Runway Situation Awareness Tool (RSAT) компании Boeing [Goodwill, 2014] и др. В нашей стране так же ведутся разработки в этой области, примеры которых представлены в работах [Коконцева, 2025], [Методика, 2022], [Аспидова, 2022], [Юневич, 2024].

Анализ причин выкатывания самолётов за пределы ВПП, способов и средств предупреждения таких событий и минимизации их последствий показывает, что основную роль в этом играет своевременное принятие экипажем решений об использовании средств торможения или прерывании посадки с последующим уходом на второй круг или запасной аэро-

дром. Однако, учитывая сложность условий работы экипажа при выполнении посадки, весьма актуальной является поддержка экипажа со стороны бортового оборудования, заключающаяся в своевременном предоставлении ему объективных оценок складывающейся ситуации, предупреждений об угрозах безопасному выполнению посадки и рекомендаций по принятию решений, направленных на предотвращение выкатывания.

В упомянутых выше разработках основное внимание уделено следующим вопросам:

- выявления угрозы выкатывания самолёта за пределы ВПП на воздушном и наземном этапе посадки и предупреждения экипажа об этой угрозе;
- формирования рекомендаций о прерывании посадки при угрозе выкатывания;
- формирования рекомендаций по использованию устройств торможения при угрозе выкатывания.

В то же время прерывание посадки предполагает два варианта возможных действий: уход на второй круг и повторную попытку посадки или уход на запасной аэродром. Выбор одного из них обусловлен причинами, вызвавшими угрозу выкатывания. Вопрос о поддержке принятия соответствующего решения в настоящее время проработан слабо.

В связи с этим в данной работе рассматривается ситуация, когда экипажем было принято и реализовано решение о прерывании посадки, а дальнейшие действия допускают две указанные выше альтернативы.

Цель работы: повышение безопасности полётов на этапе посадки за счёт поддержки принятия решения об уходе на второй круг или на запасной аэродром после неудачной попытки посадки.

Для достижения поставленной цели были решены следующие задачи:

- анализа факторов, влияющих на принятие решения об уходе на второй круг или запасной аэродром;
- выбора метода накопления и формализации знаний, используемых для принятия данного решения;
- разработки архитектуры базы знаний (БЗ) бортовой оперативно советующей экспертной системы;
- разработки прототипа бортовой оперативно советующей экспертной системы;
- исследования прототипа бортовой оперативно советующей экспертной системы.

1. Анализ факторов, влияющих на принятие решения об уходе на второй круг или запасной аэродром, и выбор метода формализации знаний

На основании проведённых консультаций с экспертом – профессиональным пилотом был сделан вывод от том, что решение об уходе на второй круг или запасной аэродром после прерванной посадки принимается пилотами самолёта с учётом многих факторов, среди которых можно отметить следующие:

- требования руководящих документов;
- запас топлива на борту самолёта;
- текущие и прогнозируемые метеоусловия в аэропорту посадки и в запасном аэропорту;
- состояние ВПП в аэропорту посадки;
- метеоминимумы членов экипажа, самолёта и ВПП;
- субъективную оценку экипажем своей готовности выполнить повторную посадку в текущих условиях и др.

Источником экспертных знаний о способах принятия указанных решений может являться накопленный опыт успешного решения аналогичных задач при различных сочетаниях перечисленных факторов. В связи с этим было принято решение о разработке БЗ для принятия решения об уходе на второй круг или запасной аэродром на основе метода вывода по прецедентам [Федунов, 2023].

Для решения по прецеденту некоторой прикладной задачи необходимо найти в предметной области, в которой возникла данная задача, аналогичные задачи (АЗ), которые были успешно решены в прошлом. Эти задачи образуют множество прецедентов. Каждый прецедент характеризуется условиями (совокупностью факторов), в которых он был реализован. Для построения БЗ эти условия формализуются в виде ситуационного вектора SV .

В [Федунов, 2023] показано, как каждый прецедент представляется своим нечётким множеством (НМ-прецедента), определённом на декартовом произведении наборов значений координат ситуационного вектора. Данное декартово произведение представляет собой универсальное множество нечётких множеств прецедентов (УМ-прецедентов). Конкретные значения координат ситуационного вектора, описывающие условия возникновения задачи, которую нужно решать – это точка в УМ-прецедентов. Значение функции принадлежности каждого НМ-прецедента в этой точке – это приоритет прецедента.

2. Разработка базы знаний бортовой оперативно советующей экспертной системы

В качестве прикладной задачи в данной работе рассматривается ситуация, когда при выполнении посадки после касания ВПП системой ситуационной осведомлённости [Product Description, 2009, Safety Innovation, 2022, Goodwill, 2014, Юневич, 2024, Коконцева, 2025] была выявлена угроза продольного выкатывания самолета, и экипажем было принято решение о прерывании пробега по ВПП и взлете. После отрыва самолёта от ВПП экипаж должен принять решение либо о повторной попытке посадки (уход на второй круг), либо об уходе на запасной аэродром. Необходимо разработать БЗ бортовой оперативно советующей экспертной системы, обеспечивающую формирование соответствующей рекомендации экипажу на основе накопленного положительного опыта принятия подобных решений.

Для разработки БЗ проведен поиск и анализ реальных случаев принятия таких решений и результатов их выполнения, которые составили множество АЗ. В результате их анализа были выявлены два прецедента:

- уход на второй круг;
- уход на запасной аэродром.

Также на основе анализа АЗ были выявлены факторы, значимые для принятия одного из перечисленных решений. Для прототипа экспертной системы было принято решение ограничиться учётом трёх факторов, которые представлены в виде ситуационного вектора **SV**, включающего следующие элементы:

- скорость попутной составляющей ветра;
- величина установившегося квазипостоянного отрицательного продольного ускорения самолёта при пробеге по ВПП;
- допустимое время ожидания улучшения условий посадки (ослабления попутного ветра и (или) повышения коэффициента сцепления ВПП), ограниченное запасом топлива для ухода на запасной аэродром.

Следует отметить, что установившееся продольное ускорение самолёта при торможении на ВПП включено в вектор **SV**, как величина, косвенно характеризующая реальный коэффициент сцепления с ВПП в момент неудачной посадки. Проведенные стендовые эксперименты показали, что при торможении самолёта на ВПП, как правило, наблюдается участок, на котором производная ускорения близка к нулю [Юневич, 2024].

Элементам вектора **SV** поставлены в соответствие лингвистические переменные (ЛП), представленные в табл. 1.

Таблица 1

Имя ЛП	Смысл	Терм-множество		Универсальное множество	
l_1	Скорость попутного ветра	t_{11}	Слабый	U_1	3..15 м/с
		t_{21}	Умеренный		
		t_{31}	Сильный		
l_2	Продольное ускорение	t_{12}	Слабое	U_2	-4..-0,1 м/с ²
		t_{22}	Среднее		
		t_{32}	Нормальное		
l_3	Допустимое время ожидания	t_{13}	Малое	U_3	5..30 мин
		t_{23}	Среднее		
		t_{33}	Большое		

Для каждого термина было сформировано нечёткое множество. Универсальные множества представлены в табл. 1, а обобщённый вид функций принадлежности показан на рис. 1.

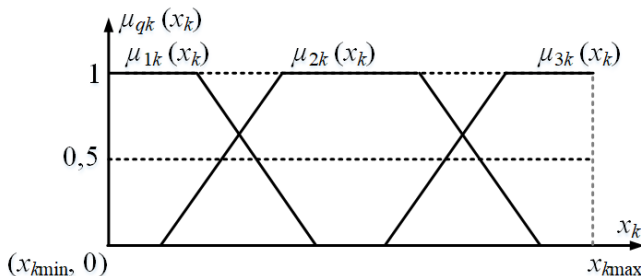


Рис. 1. Нечёткие множества ЛП, соответствующих элементам ситуационного вектора

На рис. 1 обозначены: x_{kmin} и x_{kmax} — соответственно нижняя и верхняя границы универсального множества U_k , $k = \overline{1,3}$, $\mu_{qk}(x_k)$ — функция принадлежности термина t_{qk} , $q = \overline{1,3}$, $k = \overline{1,3}$. В соответствии с методом вывода на основе прецедентов [Федунов, 2023] была построена экспериментальная матрица, фрагмент которой представлен в табл. 2. Экспериментальная матрица содержит известные из опыта состояния ситуационного вектора, при которых был реализован тот или иной прецедент.

Таблица 2

№ блока	№ АЗ	l_1	l_2	l_3	Прецедент
		Слабый Умеренный Сильный	Слабое Среднее Нормальное	Малое, Среднее, Большое	
1	1.1	Слабый	Среднее	Большое	Второй круг
	2.1	Умеренный	Нормальное	Среднее	
2	2.1	Сильный	Слабое	Малое	Запасной аэродром
	2.2	Умеренный	Среднее	Малое	

3. Пример решения задачи в базе знаний бортовой оперативно-советующей экспертной системы

Вид процедурной матрицы БЗ и процесс логического вывода на ней проиллюстрируем на примере решения конкретной задачи. Условия задачи определяются состоянием вектора **SV**, представленного в табл. 3.

Таблица 3

SV	x_1	x_2	x_3
Значение	5 м/с	-3 м/с ²	20 мин

В результате вычисления функций принадлежности термов ЛП из табл. 1 при текущем состоянии вектора **SV** процедурная матрица принимает вид, представленный в табл. 4.

Таблица 4

Процедурная матрица знаний							
№ блока	№ АЗ	Ситуационный вектор SV Значения функций принадлежности			min по строке	max по блоку	Прецеденты
		$\mu_{q1}(x_1)$	$\mu_{q2}(x_2)$	$\mu_{q3}(x_3)$			
1	1.1	1	0	0.5	0	0,2	Второй круг
	1.2	0,4	1	0,2	0,2		
2	2.1	0	0	0	0	0	Запасной аэродром
	2.2	0,4	0	0	0		

Из табл. 4 видно, что в данном примере в качестве рекомендуемого решения будет выбран прецедент №1 – уход на второй круг с повторной попыткой посадки.

Использованный в докладе метод построения БЗ реализован в экспериментальном прототипе бортовой оперативно советующей экспертной системы. Были проведены исследования данного прототипа на стенде комплекса бортового оборудования самолета МС-21 [Юневич, 2024], [Коконцева, 2025].

Заключение

Проведённый анализ причин продольного выкатывания самолётов за пределы ВПП при посадке и известных способов предотвращения таких происшествий указывает на необходимость развития средств поддержки принятия экипажем самолёта решений при возникновении угрозы выкатывания. В данной работе рассмотрена задача принятия решения об уходе на второй круг или запасной аэродром в ситуации прерывания посадки после касания ВПП.

Для разработки БЗ бортовой оперативно советующей экспертной системы, вырабатывающей соответствующие рекомендации, выбран метод вывода на основе прецедентов.

В ходе работы был сформирован набор аналогичных задач, отражающих опыт успешного принятия решений в подобной ситуации. Эти знания были формализованы в виде нечёткого ситуационного вектора и экспериментальной матрицы знаний. Для реализации вывода по прецедентам построена процедурная матрица знаний, разработаны алгоритмы вывода.

Разработанная БЗ реализована в прототипе бортовой оперативно советующей экспертной системы, поддерживающей принятие решений при прерванной посадке. Проведены стендовые исследования созданного прототипа системы.

Список литературы

- [Анализ, 2021] Анализ состояния безопасности полетов в гражданской авиации Российской Федерации в 2020 году. – М.: Федеральное агентство воздушного транспорта. Управление инспекции по безопасности полетов, 2021.
- [Мозоляко, 2014] Мозоляко А.В., Акимов А.Н., Воробьев В.В. Проблемы предотвращения выкатывания гражданских воздушных судов на этапе пробеге по ВПП // Научный вестник МГТУ ГА. – 2014. – № 204. – С. 74-77.
- [Государственная политика, 2017] Государственная политика в области предотвращения авиационных происшествий, связанных с использованием ВПП. Семинар по безопасности полетов, Москва, Внуково, 2017. Федеральное агентство воздушного транспорта. 2017.
- [EPAS, 2024] European Plan for Aviation Safety (EPAS) 2024. – 13th ed. 23 Jan 2024. EASA. 2024. [Электронный ресурс] // EASA. – URL: <https://www.easa.europa.eu/en/document-library/general-publications/european-plan-aviation-safety-epas-2024#group-easa-downloads> (дата обращения: 28.05.2025).
- [Бородкин, 2024] Бородкин С.Ф., Волынчук А.И., Киселев М.А., Петров Ю.В. Возможности и недостатки систем предупреждения выкатывания воздушного судна за пределы взлетно-посадочной полосы // Научный вестник МГТУ ГА.

- 2023. – № 3. [Электронный ресурс]. – URL: <https://cyberleninka.ru/article/n/vozmozhnosti-i-nedostatki-sistem-preduprezhdeniya-vykatyvaniya-vozdushnogo-sudna-za-predely-vzletno-posadochnoy-polosy> (дата обращения: 07.04.2025).
- [Georges, 2017]** Georges P. EUROCAE ED-250: ROAAS MOPS. Runway Overrun Alerting and Awareness System Minimum Operational Performance Specifications. GRSS 2017. Lima, Peru. [Электронный ресурс] // ICAO. – 2017. – URL: https://www.icao.int/Meetings/GRSS-2/Documents/Panel%207/7.1_EUROCAE%20ED-250%20v7%20light.pdf (дата обращения: 09.04.2025).
- [Product Description, 2009]** Product Description, Flight Safety Functions of the Enhanced Ground Proximity Warning System. Honeywell International Inc. 2009. [Электронный ресурс] // Honeywell. – 2009. – URL: <https://skybrary.aero/sites/default/files/bookshelf/898.pdf> (дата обращения: 09.04.2025).
- [Safety Innovation, 2022]** Safety Innovation #5: Runway Overrun Prevention System (ROPS) and Runway Safety Suite. [Электронный ресурс] // Airbus. – 2022. – URL: <https://www.airbus.com/en/newsroom/stories/2022-10-safety-innovation-5-runway-overrun-prevention-system-rops-and-runway> (дата обращения: 09.04.2025).
- [Goodwill, 2014]** Goodwill S. Runway situation awareness tools (RSAT). Flight Technical and Safety [Электронный ресурс] // The Boeing Company. – 2014. – URL: [https://www.icao.int/SAM/Documents/2014-UNSTAPPC/BOEING%20Runway%20situation%20awareness%20tools%20\(RSAT\).pdf](https://www.icao.int/SAM/Documents/2014-UNSTAPPC/BOEING%20Runway%20situation%20awareness%20tools%20(RSAT).pdf) (дата обращения: 09.04.2025).
- [Коконцева, 2025]** Коконцева С.Ю., Федунев Б.Е., Юневич Н.Д. Бортовые интеллектуальные системы тактического уровня на фазах посадки воздушного судна // Известия Российской Академии Наук. Теория и системы управления. – 2025. – № 1. – С. 121-139. – doi: 10.31857/S0002338825010105.
- [Методика, 2022]** Себряков Г.Г., Мужичек С.М., Тестова Т.М., Маслова Л.А. Методика предупреждения выкатывания летательного аппарата со взлетно-посадочной полосы и предупреждения его повреждения при движении по аэродрому // Труды ГосНИИАС. Серия: Вопросы авионики. – 2022. – Вып. 4(59). – С. 46-56.
- [Аспидова, 2022]** Аспидова А.М., Еремин А.И., Федунев Б.Е. Интеллектуальная поддержка решения задачи оперативного целеполагания экипажем самолета на этапе «Посадка»: коллизия «Пробег по ВПП - угроза «Слабое торможение» // Известия Российской Академии Наук. Теория и системы управления. – 2022. – № 6. – С. 133-149.
- [Юневич, 2024]** Юневич Н.Д., Липатов А.А., Коконцева С.Ю. Демонстрационный образец бортовой оперативно советующей экспертной системы этапа полета «Посадка»: первая версия базы знаний, результаты ее испытаний на стенде КБО МС-21 // Труды ГосНИИАС. Серия: Авиационные системы. – 2024. – № 1(64). – С. 19-32.
- [Федунев, 2023]** Федунев Б.Е. Решение задач по прецеденту в базах знаний бортовых интеллектуальных систем тактического уровня на этапах выполнения миссии подвижным объектом // Известия Российской академии наук. Теория и системы управления. – 2023. – № 1. – С. 137-147. – doi: 10.31857/S0002338823010018.

Abstracts of Sectional Reports

Section 5

Fuzzy Models, Soft Measurements and Computing, Bioinspired Methods

AGENT-BASED ALGORITHM FOR SOLVING SYSTEMS OF LINEAR ALGEBRAIC EQUATIONS OF LARGE DIMENSION

D.A. Bereza (*dbereza@sfedu.ru*)
L.A. Gladkov (*lagladkov@sfedu.ru*)
N.V. Gladkova (*nvgladkova@sfedu.ru*)

Southern Federal University, Taganrog

The paper solves the problem of constructing an effective method for solving systems of linear algebraic equations (SLAE) of large dimension. The high relevance and practical significance of solving the problem was noted. The analysis of various approaches to the construction of mathematical models of design objects is carried out. The mathematical formulation of the problem is given. A mechanism for reducing the dimension of the solved system of equations is proposed. The idea of integrating evolutionary algorithms and optimization methods based on the theory of Pareto optimality is proposed. A hybrid algorithm for solving SLAE based on genetic search and multi-agent organization has been developed. The architecture of the software subsystem for solving SLAE based on the use of a service-oriented approach has been developed. A software implementation of the proposed hybrid algorithm was performed and a series of computational experiments were conducted. The results obtained generally confirmed the effectiveness of the proposed approach and allowed us to identify promising areas for further research development.

Keywords: computer-aided design systems, ultra-large integrated circuits, multi-agent systems, service-oriented architectures, evolutionary computing.

SOFT SITUATIONAL-COGNITIVE MODELS FOR INTELLECTUAL DECISION SUPPORT

V.V. Borisov (*vbor67@mail.ru*)
A.S. Fedulov (*fedulov_a@mail.ru*)
S.A. Fedulova (*svfed67@mail.ru*)

Branch of the National Research University
“Moscow Power Engineering Institute”, Smolensk

The Soft Situational-Cognitive Models (SSCM) are proposed, based on polymorphic hybridization of fuzzy situational and cognitive models, providing: their structural and parametric correspondence; the interdependence of construction and configuration pro-

cedures within a single polymorphic model; their joint interdependent use for intellectual support of situational decision-making. The developed method of constructing and configuring the SSCM is considered, in which it is proposed to use the results of scenario modeling in the cognitive network structure of the SSCM to assess the impact of situational decisions on dependent features and take into account their time-distributed effects. An example of using SSCM for intellectual decision support is considered.

Keywords: Soft Situational-Cognitive Model, polymorphic hybridization, decision support.

A STRUCTURALLY SENSITIVE APPROACH TO THE ORGANIZATION OF INTELLIGENT MONITORING BASED ON FUZZY EVOLVING MODELS

A.I. Dolgiy (*a.dolgiy@vniias.ru*)
S.M. Kovalev (*ksm.1954@yandex.ru*)

JSC NIIAS, Moscow

The paper presents a new principle of organization of the structurally sensitive mechanism of functioning of fuzzy evolving models used in intelligent monitoring systems. The proposed approach ensures synchronization of the dynamics of the system model with the progress of the technological process and allows, based on the analysis of structural changes occurring in the model, to detect characteristic features in the behavior of this process. A formal criterion for the structural sensitivity of the model is proposed for the practical use of the structure-sensitive approach in monitoring algorithms. Based on this criterion, an algorithmically simple procedure for analyzing the structural sensitivity of a system model has been developed in order to use it for analytics and machine learning.

Keywords: incremental learning, structural sensitivity, fuzzy evolving model, intelligent monitoring.

MULTI-LEVEL HIERARCHICAL EVOLUTIONARY ALGORITHM OF RATIONAL SELECTION OF PARAMETERS OF THE SYSTEM OF MANAGEMENT OF PROMOTION IN SOCIAL NETWORKS

K.V. Zakharchenkov (*zakharchenkovkv@mail.ru*)
A.E. Misnik (*anton@misnik.by*)
A.M. Sharaburka (*sheren@tut.by*)

Belarusian-Russian University, Belarus, Mogilev

The paper describes a multi-level hierarchical evolutionary algorithm of rational selection of parameters of the system of management of promotion in social networks. Within the framework of evolutionary modeling, a recursive hierarchical genetic algorithm has been implemented, allowing to form optimal sets of characteristics of publications and to choose rational solutions based on the target function. The mechanisms of

genetic operators (breeding, crossing, mutation, elitist selection) provide the search for a global optimum. A distinctive feature of the proposed algorithm is the rational selection of the values of publication characteristics at the first level and the rational selection of the set of publications for the content plan at the second level. The proposed algorithm can be used to improve the effectiveness of promoting goods, works, and services on social networks.

Keywords: Social Media Marketing, Characteristics of Publications, Content Plan Formation, Rational Solution, Evolutionary Modelling, Genetic Algorithm.

METHOD AND PROGRAMMING LIBRARY IMPLEMENTATION FOR DEALING WITH STRATEGIES OF FUZZY NUMBER UNIVERSE RESTRICTING

M.M. Zernov (*zmmioml@eandex.ru*)^{A,B}

P.P. Petrov (*petr@petrov.ru*)^{A,B}

N.A. Makeenkov (*n.a.makeenkov@yandex.ru*)^A

^A JSC Radiozavod, Penza

^B Branch of the National Research University Moscow Power Engineering
Institute in Smolensk, Smolensk

The paper represents method for dealing with strategies of fuzzy number universe restricting, based on higher order functions, aimed to manage fuzzy numbers interpretation due to limitations of universe. Offered method was implemented in a new C++ library for fuzzy calculations. Library classes hierarchy is build around 3 design patterns: “strategy”, CRTP, “visitor”. Additional technique for input elements, been out of fuzzy number support discarding, applied for membership function evaluation for ordered input numbers arrays. Performance evaluation tests made, proved method realization efficiency.

Keywords: higher order functions, fuzzy mathematics, fuzzy numbers.

IMPLEMENTATION OF THE FUZZY GRAPH SEPARATION ALGORITHM FOR THE FORMATION OF THE OBJECTS INFLUENCE ZONES IN COMPLEX TECHNICAL SYSTEMS

E.R. Zyablova (*zero32f@mail.ru*)^A

A.O. Ermakov (*alee@sfedu.ru*)^A

E.S. Shevketova (*shevketova@sfedu.ru*)^A

^A Southern Federal University, Taganrog

The paper considers the implementation features of a modified algorithm for dividing a fuzzy graph with different types of vertices and multiple edges of different types. The use of such a graph for modeling complex technical systems (CTC) with different types of information flows is substantiated. The advantage of the modified separation

algorithm is to preserve the computational complexity of the original algorithm, taking into account the possibility of processing a fuzzy graph with different types of vertices and multiple edges of different types. A distinctive feature of the implemented algorithm is the possibility of dividing the graph into proportional subgraphs (by the number of vertices). Approaches to the formation of zones of influence of system objects using a separation algorithm and an algorithm for calculating graph metrics are described. As an example of an STS, an extended perimeter security system is considered, for which agent influence zones are formed depending on the metrics of the subgraphs.

Keywords: fuzzy graph, different types of vertices, different types of multiple edges, graph separation, security system, software implementation.

APPROACHES TO CONSIDERING THE UNCERTAINTY OF THE PARAMETER VALUES OF FUZZY COGNITIVE MAPS

R.A. Isaev (*Ruslan-Isaev-32@yandex.ru*)^{A,B}

K.I. Rychenkova (*k.ri4enkova@yandex.ru*)^B

I.A. Kopeliovich (*igor.copeliovitch@yandex.ru*)^B

A.G. Podvesovskii (*a.podvesovski@gmail.com*)^{A,B}

^A V.A. Trapeznikov Institute of Control Sciences
of Russian Academy of Sciences, Moscow

^B Bryansk State Technical University, Bryansk

The paper describes a problem related to the insufficient validity of the parameter values of fuzzy cognitive models using the example of V.B. Silov's fuzzy cognitive maps. The reasons for the existence of this problem are considered, and the example demonstrates that when using a cognitive model to develop management strategies, lack of confidence in the values of parameters can lead to unreliable decisions. The disadvantages of the applied approaches to reducing the influence of uncertainty in the values of cognitive model parameters on modeling results are described, and new ways of solving the problem are proposed. The first of them involves the modernization of the mathematical apparatus of fuzzy cognitive maps of Silov, in which uncertainty about the values of parameters is interpreted as a subjective probability of these values, while the parameters are presented in the form of random variables. The second approach is based on an analysis of the stability of the model parameters, which is understood as a study of the dependence of the modeling results on possible errors in the parameter values. The key research directions for the development of each of the proposed approaches are presented.

Keywords: cognitive modeling, fuzzy cognitive map, parametric identification, uncertainty of parameters, stability analysis, random variable.

INTERPRETABLE MULTILAYER CLASSIFIER BASED ON A FUZZY COGNITIVE NETWORK FOR FAULT DETECTION AND DIAGNOSIS IN INDUSTRIAL SYSTEMS

Y.S. Napadailo (*napadailo@skynapp.com*) ^A

V.A. Tailakov (*vic7519@mail.ru*) ^A

G.V. Rybina (*gvrybina@yandex.ru*) ^A

^A National Research Nuclear University MEPhI, Moscow

This paper proposes an interpretable multilayer classifier based on a fuzzy cognitive model for fault detection and diagnosis, with a particular focus on wind turbine applications. The proposed method is evaluated in terms of fault detection accuracy and overall performance against a state-of-the-art recurrent neural network classifier. Furthermore, several approaches to interpreting the output of black-box models for fault detection and diagnosis, including those enhanced with a fuzzy cognitive model for improved explainability, are compared with the authors' proposed model.

Keywords: fault detection and diagnosis, interpretability, fuzzy cognitive model, intelligence decision support systems.

UTILIZING FUZZY SETS FOR THE DEVELOPMENT OF NEXT-GENERATION HYBRID EXPLAINABLE ARTIFICIAL INTELLIGENCE

Y.V. Trofimov (*ura_trofim@bk.ru*)

A.L. Averkin (*averkin2003@inbox.ru*)

A.K. Alekseev (*aak.24@uni-dubna.ru*)

A.V. Shevchenko (*leviathan0909@gmail.com*)

Dubna State University, Dubna

The paper presents an overview of contemporary set-theoretic formalisms used to represent uncertainty in explainable artificial intelligence (XAI). It surveys classical fuzzy sets introduced by L. Zadeh, their extensions – intuitionistic fuzzy sets proposed by K. Atanassov, neutrosophic sets developed by F. Smarandache – as well as the most recent plithogenic set formalism. These approaches are shown to model not only degree-based uncertainty but also knowledge incompleteness and inconsistency. The study further examines the role of these formalisms in the evolution of XAI from the first generation (XAI 1.0) to the more advanced paradigms of XAI 2.0 and XAI 3.0.

Keywords: XAI, Artificial Intelligence, Fuzzy Sets.

DEVELOPMENT OF A PROTOTYPE RECURSIVE EXPLAINABLE ARTIFICIAL INTELLIGENCE SYSTEM BASED ON BEHAVIORAL-LOGICAL MODELS

Y.V. Trofimov (*ura_trofim@bk.ru*)
A.L. Averkin (*averkin2003@inbox.ru*)
I.P. Muravyov (*mip.22@uni-dubna.ru*)
A.K. Alekseev (*aak.24@uni-dubna.ru*)
E. M. Kuznetsov (*Kot454556@yandex.ru*)

Dubna State University, Dubna

The paper introduces a prototype of a recursive explainable artificial intelligence (XAI) system that integrates behavioral-logical models of reasoning with state-of-the-art neural-network technologies. The architecture follows a cognitive dual-system paradigm. System 1 is implemented as a hybrid graph-capsule neural network that carries out fast, intuitive recognition. System 2 is built on a differentiable logical reasoned – the Neural Theorem Prover – which performs formal deduction in a continuous feature space. Mode switching is governed by a Kolmogorov-Arnold Network (KAN) that dynamically allocates confidence between the two subsystems. To maintain coherence across the neuro-symbolic hierarchy, fuzzy logic acts as an intermediate binding layer. The proposed architecture exemplifies the principles of XAI 2.0, delivering a high degree of interpretability without sacrificing predictive accuracy. Each inference is accompanied by an internal logical justification, providing operational transparency and robustness to uncertainty in complex cognitive tasks.

Keywords: XAI, Graph Neural Networks, Capsule Neural Networks, Neural Theorem Prover, Kolmogorov-Arnold Network, fuzzy logic.

ON QUANTITATIVE CHARACTERISTICS OF UNCERTAINTY IN FINITE BLURRY MODELS

G.E. Yakhyaeva (*gul_nara@mail.ru*)
O.D. Palchunova (*o.palchunova@g.nsu.ru*)

Novosibirsk State University, Novosibirsk

Blurry model theory represents an alternative way to formalize imprecision and/or incompleteness of knowledge compared to the fuzzy set approach proposed by Lotfi Zadeh. When blurry models are used to formalize knowledge about subject domains, there is a need to quantitatively describe the degree of uncertainty and fuzziness of these models. In this paper, various quantitative characterizations that take into account the uncertainty of finite blurry models have been discussed. The paper presents different methods for formalizing the degree of uncertainty of a blurry model: a probabilistic approach, which takes into account the degree of randomness of an event in the subject domain; a fuzzy approach, which is related to the uncertainty of concepts in the domain; and an object approach, which describes the uncertainty at the object level. The separability coefficient of the blurry model is also analyzed, which shows how many independent submodels a given blurry model can be partitioned into.

Keywords: blurry model, entropy, separability coefficient.

Section 6

Intelligent Agents, Robots, Computer Vision

CLASSIFICATION OF BARN-RAISED POULTRY CONDITION BASED ON THE USE OF YOLO

D.G. Bukhanov (*dgbukh@gmail.com*)^A

S.V. Chernikov (*serg4315@mail.ru*)^B

Yu.A. Koshlich (*sergey@sergeev.ru*)^A

^ABelgorod State Technological University named after V.G. Shukhov, Belgorod

^BEntersoft LLC, Belgorod

Analysis of the research status in the field of automatic determination of the functional state of floor-raised poultry has shown that existing approaches do not take into account the specific conditions of real-world poultry farm production sites. The difficulty of deploying full-scale workstations or establishing remote access to surveillance cameras within poultry houses necessitates the use of low-performance or single-board computers for data analysis. A modification of a YOLO-based classifier is proposed, involving the separation of image preprocessing into a distinct stage, executed once per input image. Experiments were conducted on classification of floor-raised poultry, detection of mortality, and assessment of feeder fullness. As a result of the experiments, performance improvements of 3% and 12.5% were achieved for a GPU-equipped workstation and a Raspberry Pi 5 single-board computer, respectively. Based on the experimental results, a conclusion was drawn regarding the applicability of the proposed approach on single-board or low-performance computers, with the goal of improving the performance of functional state analysis systems for floor-raised poultry.

Keywords: video analysis, YOLO, Artificial Neural Networks, computer vision.

ACCELERATION OF DATA EXCHANGE NEAR OBSTACLES IN MULTI-AGENT FORAGING

D.I. Zvorykin (*zuev.di@phystech.edu*)

P.S. Sorokoumov (*petr.sorokoumov@yandex.ru*)

NRC Kurchatov Institute, Moscow

It is very important to correctly organize the exchange of information between the group of foraging agents. In a decentralized system, communications can be organized by pairwise interactions of agents. The paper considers the role of obstacles and bottlenecks in the formation of a favorable environment for numerous exchanges. The reinforcement learning of groups with different organizations of exchange processes and various forms of obstacles showed a weak effect of randomly located obstacles and a

more significant effect of extended obstacles. The role of data exchange turned out to be higher at the initial stage of the modelling and in changing environments. The results of the work can be used to intensify data exchange processes in multiagent groups.

Keywords: multi-agent foraging, reinforcement learning, information distribution.

SOME BIO-INSPIRED AGENT-BASED NEST BUILDING MODELS

V.E. Karpov (*karpov.ve@gmail.com*)

National Research Centre “Kurchatov Insitute”, Moscow

Two agent-based models of construction activity are considered. Using the example of the construction of an ant nest analogue, it is shown that agents are able to erect a dome and simulate tunneling using the basic law of group movement without the need for active interagent interaction and without using planning procedures. The basic motion control mechanism is based on the generalized potential method, when positive and negative potentials are formed for different classes of objects. Their summation, taking into account the values of the vector of factor coefficients, makes it possible to realize various motion effects.: such as the desire to stick together, repulsion, movement along a bearing or to a selected point, etc. Switching between forms of movement is carried out by a finite automaton. The construction of the dome is based on the ability of agents to climb on top of each other, and the area of interest of the coordinated movement of the group is the multitude of agents in terminal states. Tunnel construction is a more complex task, for which it is necessary to determine both the general direction (bearing) of movement, but also to determine the conditions for its start/completion.

Keywords: agent, behavioral models, method of potentials, nest building, finite state machine-based control.

INTERACTIVE SERVICE FOR VISUALIZING IMAGES OF COMPLEX OBJECTS USING A UNIFIED GRAPHIC VISUALIZATION OF ACTIVITY

M.V. Kuznetsov (*kurt21.89@mail.ru*)^A

V.A. Uglev (*uglev-v@yandex.ru*)^{A,B}

^A Department of Applied Physics and Space Technologies,
Siberian Federal University, Zheleznogorsk

^B Department of Intelligent Systems in the Humanities
of the Russian State University for the Humanities, Moscow

The paper describes a tool solution for one of the Data Visualization approaches from Data Mining, the Unified Graphic Visualization of Activity (UGVA) method. A generalized model of the initial data formalization is given and the technology of their display on the part of an anthropomorphic image is briefly described. The interactive features of the service and the possibilities of its integration into analytical systems are

discussed. Using the example of the formation of images of employees of a scientific and educational organization, it is shown how the UGVA method can be applied in intelligent decision support systems.

Keywords: Data Mining, cognitive visualization, Unified Graphic Visualization of Activity, intelligent decision support system.

ALGORITHMS FOR MULTI-AGENT PATHFINDING

A.I. Lutsenko (*ailutsenko@edu.hse.ru*)^A

K.S. Yakovlev (*kyakovlev@hse.ru*)^{A,B}

^A National Research University "Higher School of Economics", Moscow

^B Federal Research Center "Computer Science and Control"
of the Russian Academy of Sciences, Moscow

This paper studies the problem of multi-agent pathfinding on rectangular grid. This model is useful when routing agents in warehouses. However, it takes time for the robots to make a turn and this time is rather considerable and cannot go unnoticed. That is why this paper studies a slight modification of the problem: multi-agent pathfinding with turn actions, which assumes that the time agent spends on turning is equal to the time it spends on going forward. Several fast algorithms are proposed for solving this problem. Results of the experimental research are given at the end of the article.

Keywords: multi-agent planning, graph, robot, shortest path, graph pathfinding.

NAVIGATION OF INTERCHANGEABLE AGENTS IN CONTINUOUS SPACE

A.V. Mikryukova (*avmikryukova@edu.hse.ru*)^A

S.A. Dergachev (*sadergachev@hse.ru*)^{A,B}

^A National Research University "Higher School of Economics", Moscow

^B Federal Research Center "Computer Science and Control"
of the Russian Academy of Sciences, Moscow

This work addresses the problem of unlabeled multi-agent pathfinding in continuous space. Unlike the classical case, the target positions in this formulation are not assigned to specific agents. A review of modern multi-agent planning methods is conducted, highlighting the *GAP* algorithm, which uses a continuous space representation and imposes minimal requirements on input data. The algorithm was experimentally compared to state-of-the-art discrete approaches. The results demonstrate that switching to a continuous representation significantly improves solution quality. However, the algorithm's constraints on input data may reduce the success rate as agent density increases.

Keywords: multi-agent navigation, multi-agent pathfinding, interchangeable agents navigation, unlabeled navigation.

ENSEMBLES OF NEURAL NETWORKS FOR IMAGE PREPROCESSING IN CLINICAL DECISION SUPPORT SYSTEMS

I.I. Mishchenko (*sombra74@yandex.by*)

A.E. Misnik (*anton@misnik.by*)

Inter-state educational institution of higher education
«Belarusian-Russian University» Belarus, Mogilev

The paper deals with the module of X-ray image preprocessing developed for use in medical decision support systems oriented to the diagnosis of Hip-Spine syndrome. The module is based on an ensemble of convolutional neural networks implemented according to a hierarchical architecture: the controlling supervisor network analyzes the characteristics of the input image and activates specialized submodules responsible for the correction of brightness, contrast and restoration of local features. This approach provides adaptive processing depending on the quality of visual data. Experiments have shown an 18% increase in classification accuracy compared to basic processing methods. The detailed architecture of the module, principles of its functioning, results of efficiency evaluation and prospects for further development in the direction of 3D reconstruction integration are presented.

Keywords: Medical decision support system, Hip-Spine syndrome, pre-processing, neural networks, X-rays.

REGRESSION-BASED NEURAL NETWORKS FOR UAV POSE ESTIMATION DRIVEN BY LOSS AND WEIGHT REGULARIZATION

Ya.S. Pikalyov (*i@pikaliov.ru*)

B.V. Pavlenko (*bogdanpavl2000@mail.ru*)

Federal State Budgetary Scientific Institution
"Institute of Artificial Intelligence Problems", Donetsk

This paper presents the architecture of the regression neural network RegrUAV, designed for accurate estimation of UAV camera orientation angles from aerial images. A modified loss function, Focal Huber Loss (FHL), is proposed, combining the advantages of Huber Loss and Focal Loss to make model more robust while compensating for data imbalance. To further improve regression accuracy, the Feature Distribution Smoothing (FDS) method is integrated, enabling feature distribution smoothing and calibration of statistics during training. Experiments were conducted on the GTA-UAV and VisLoc datasets using different backbone architectures (StripNet-small, InceptionNext-tiny). The results demonstrate that the combination of FHL and FDS reduces the MAE and enhances the model's robustness to data imbalance and variability. On the GTA-UAV dataset, the StripNet-small backbone achieved the highest accuracy.

Keywords: convolutional neural networks, regression, UAV.

TRANSITION FROM OFFLINE TO ONLINE REINFORCEMENT LEARNING FOR ROBOTIC MANIPULATION WITH TRANSFORMER-BASED MODELS

D.I. Poyarkov (*poiarkov.di@phystech.edu*)^{A,B}

A.V. Staroverov (*alstar8@yandex.ru*)^{B,C}

A.I. Panov (*panov.ai@mipt.ru*)^{A,B,C}

^A MIPT, Dolgoprudny

^B FRC CSC RAS, Moscow

^C AIRI, Moscow

Transformer-based models excel in robotic manipulation with large offline datasets but need online fine-tuning for real-world reliability. This study compares two offline pre-training methods—behavior cloning and offline RL with hierarchical actor-critic learning—to accelerate online RL. Experiments show that offline RL pre-training achieves target performance with 32% fewer environment interactions, highlighting its potential to reduce sample complexity while preserving transformer benefits for robotic control. The code is available at https://gitlab.com/cogmod_mr/isaacclab_mod.

Keywords: machine learning, artificial intelligence, transformer, offline learning, online learning, reinforcement learning.

POINT-GOAL NAVIGATION OF A MOBILE ROBOT USING TOPOLOGICAL MAPS AND LEARNED MOTION METHODS

V.A. Romanenko(*viromanenko_1@edu.hse.ru*)^A

K.F. Muravyev (*muraviev@isa.ru*)^B

^A National Research University “Higher School of Economics”, Moscow

^B Federal Research Center “Computer Sciences and Control”

of Russian Academy of Sciences, Moscow

A necessary condition for the autonomous operation of a mobile robot is the ability to perform autonomous navigation (i.e., reaching a given target point). There are many classical and learnable navigation algorithms that operate with high accuracy when information about the target point's position relative to the robot is available. Typically, such algorithms are tested in a simulation environment using precise or lightly noised relative positions from the simulator. This paper presents a navigation system that relies solely on robot observations and a pre-built topological map of the environment. The system is based on trainable methods for generating actions to reach the target point, as well as on determining the robot's location and planning a path to the goal using a compact topological map. Experiments conducted in large simulated indoor environments without using positional information demonstrate successful target point achievement.

Keywords: autonomous navigation, mobile robots, topological mapping, reinforcement learning.

HYBRID METHOD OF OBJECT RECOGNITION IN ROOM POINT CLOUDS BASED ON GEOMETRIC SEGMENTATION AND DEEP LEARNING

O.I. Khristodulo (*o-hristodulo@mail.ru*)

A.Y. Baychugurov (*trappistt@yandex.ru*)

L.R. Mukhametdinova (*2mlr4@mail.ru*)

Ufa University of Science and Technology, Ufa

A hybrid approach to processing indoor point clouds is proposed, integrating geometric segmentation (RANSAC) and density clustering (DBSCAN) algorithms with neural network classification based on the PointNet architecture. This combination ensures efficient recognition of structural elements and objects in a scene while significantly reducing the requirements for the volume of labeled data due to the transition from point-by-point semantic segmentation to classification of pre-segmented clusters.

Keywords: point cloud, segmentation, clustering, PointNet neural network, classification, semantic segmentation.

Section 7

Intelligent Control and Decision Support

INVESTIGATION OF INTERACTION STRATEGIES WITH LARGE LANGUAGE MODELS FOR OPTIMAL MEETING TIME SELECTION

A.A. Agafonov (*agafonov.a@spcras.ru*)

St. Petersburg Federal Research Center of the Russian Academy of Sciences,
St. Petersburg

The paper explores various strategies for interacting with large language models (LLMs) to solve the problem of selecting an optimal meeting time based on user constraints expressed in natural language. This problem represents a special case of decision support under implicit and dynamic user preferences. Two fundamentally different strategies are analyzed: (1) an agent-based approach, which involves generating formal constraints and solving the resulting optimization problem using an external algorithm; and (2) direct generation of solutions in the form of a set of time slots, time intervals, or a binary vector. The results show that the agent-based approach provides superior accuracy and stability, especially as the number of preferences increases. At the same time, direct solution generation methods prove effective when the number of constraints is small and the formulations are simple, highlighting the importance of selecting an appropriate interaction format with LLMs in applied intelligent systems.

Keywords: large language models, constraint programming, AI agents, decision support, preference elicitation, meeting scheduling.

DECISION SUPPORT SYSTEM IN CHILDREN'S SPORTS BASED ON COMPUTER VISION AND NEURO-FUZZY ANALYSIS

D.G. Arseniev (*arsenjevasp@yandex.ru*)^A

A.A. Velkov (*6339743@mail.ru*)^B

A.E. Misnik (*anton@misnik.by*)^C

M.A. Shalukhova (*shalukhova.m@yandex.ru*)^{A,C}

^A Peter the Great St.Petersburg Polytechnic University, St. Petersburg,

^B Federal State Budget Educational Institution of Higher Education
«Tchaikovsky State Physical Education and Sport Academy», Tchaikovsky

^C Inter-state educational institution of higher education
“Belarusian-Russian university”, Belarus, Mogilev

The paper deals with the concept of an intelligent decision support system for children's sports. The proposed architecture incorporates a computer vision module, a digital athlete profile (i.e. a digital twin), an ontological knowledge base and a neuro-fuzzy module for the purpose of making recommendations. The proposed system is capable of facilitating remote monitoring of exercise technique, personalising the load plan in ac-

cordance with age and training, and identifying deviations in training performance in real time. A comparison is made between the conventional approach to coaching practice and the digital control methods under consideration. The analysis encompasses the system's fundamental components, potential implementation scenarios, and the algorithms employed for data processing within the system. The integration of such systems in the domain of paediatric sports has been demonstrated to enhance the objectivity of progress assessment, facilitate the effective construction of individual development trajectories, mitigate the risk of injuries, and unveil novel applied opportunities in the field of decision support systems.

Keywords: early sports specialisation; intelligent decision support systems; computer vision; digital twin; neuro-fuzzy systems; convolutional neural networks; personalised training.

MANEUVER DECISION-MAKING WITH TRAJECTORY STREAMS PREDICTION FOR AUTONOMOUS VEHICLES

M. Jamal (*mayssjamal@phystech.edu*)^A

A.I. Panov (*panov@airi.net*)^{A,B,C}

^A MIPT, Dolgoprudny, Russia

^B FRC CSC RAS, Moscow, Russia

^C AIRI, Moscow, Russia

Decision-making, motion planning, and trajectory prediction are key components of autonomous driving systems. This paper introduces FFStreams++, a hybrid decision-making and motion planning framework that integrates sampling-based trajectory generation with heuristic search. Maneuvers such as unprotected left turns, overtaking, and lane-keeping are modeled in the Planning Domain Definition Language (PDDL) and solved using the Fast-Forward planner. The system iteratively refines the initial state via maneuver-specific trajectory streams until an optimal plan is found. FFStreams++ incorporates a query-connected network to predict trajectories of surrounding obstacles with associated probabilities. Evaluations on the CommonRoad benchmark demonstrate the planner's effectiveness in executing complex maneuvers while maintaining safety.

Keywords: Autonomous Driving, Behavior Planning, Integrated Task and Motion Planning, Maneuver Planning, Trajectory Prediction.

APPLICATION OF LARGE LANGUAGE MODELS IN DECISION SUPPORT SYSTEMS

A.A. Kulinich (*alexkul@rambler.ru*)

Institute of Control Sciences of the Russian Academy of Sciences, Moscow

The decision support system is presented as a complex cognitive system that includes the decision maker's intelligence in the form of a model of his mental space and a pre-trained large language model. The language model is used to explain decisions formally obtained in the mental space model. A method for assessing the quality and use-

fulness of interaction between a person and a large language model is proposed. An example of assessing the language model by a group of hypothetical respondents is given. The example shows the growth of understanding of the essence of the problem by the decision maker, thanks to the explanations of the language model. The proposed model of the decision support system and the method for assessing the quality of interaction between a person and a large language model can be useful for preliminary assessment of such systems.

Keywords: decision support, large language model, explanation of decision alternatives, assessment of the quality of the decision explanation.

SYNTHESIS OF ADAPTIVE CONTROL IN MULTICHANNEL NAVIGATION RECEIVERS BASED ON INTELLIGENT PROCESSING OF PRE-CORRELATED SIGNALS

M.A. Likhachev (*likhachevma@cps.tver.ru*)

R&D Institute Centerprograms system, Tver

The article presents a two-stage synthesis of adaptive control solutions for the directional pattern of a digital antenna array (DAs) weighting coefficients of a navigation receiver under conditions of intentional noise and simulated interference, consisting in an intelligent spatio-temporal analysis of received satellite navigation signals. The proposed approach is aimed at increasing the noise immunity and information security of navigation due to noise suppression in the total pre-correlation signal. At the first stage of synthesis, weights are formed that provide deep dips in the radiation pattern in the directions of active interference sources whose power exceeds the noise level. At the second stage, a new set of weighting coefficients is formed based on the identification of the event of the presence of spoofing signals and the assessment of the angular directions of their radiation. Prediction of the coordinates of the source of spoofing signals (azimuth and elevation) is carried out using an LSTM neural network trained on data from a cyclic correlation matrix characterizing the cyclostationary properties of authentic navigation signals. Thus, the synthesized intelligent solutions for managing a set of adaptive weighting coefficients of receiving channels make it possible to minimize the impact of interference of various natures and ensure the integrity and reliability of received navigation signals for their further correlation processing and tracking.

Keywords: adaptive control, intelligent signal processing, global navigation satellite systems (GNSS), navigation receiver, digital antenna array (DA), interference suppression, spoofing, cyclostationarity, LSTM neural network.

AUTONOMOUS MULTIAGENTIC SYSTEM FOR CONTROLLING A GROUP OF ROBOTS

B.K. Lebedev (*lebedev.b.k@gmail.com*)

O.B. Lebedev (*lebedev.ob@mail.ru*)

M.I. Beskhmel'nov (*m_beskhmel'nov@mail.ru*)

MIREA – Russian University of Technology, Moscow

The paper considers methods of controlling a group of unmanned aerial vehicles with autonomous properties. The main control strategies and their features are analyzed. General principles and mechanisms necessary for developing an algorithm that provides group control of a team of aerial robots are defined, allowing each robot to autonomously navigate in space. At the same time, each robot must independently, without using GPS, establish its location, using data from a camera or lidar to identify obstacles, as well as form the most rational route and make decisions aimed at achieving the goal and completing the task.

Keywords: unmanned aerial vehicle, autonomy, artificial intelligence, decentralized control, swarm strategy.

REINFORCEMENT LEARNING WITH AN EXTERNAL TEACHER FOR ENERGY FACILITIES CONTROL

O.Yu. Maryasin (*maryasin2003@list.ru*)

A.N. Plohotnyuk (*admin@nixson.ru*)

Yaroslavl State Technical University, Yaroslavl

The article considers the problem of energy facility control using Reinforcement Learning algorithms. To tackle the problem of lengthy agent training, the authors propose an approach involving an external teacher. Various methods of using an external teacher have been described, and the energy facility control problem has been set, which allows the use of popular reinforcement learning algorithms with an external teacher. The results of training agents in a virtual environment show that, for all reinforcement learning algorithms, the use of an external teacher enables achieving much lower energy consumption within a fixed timeframe compared to similar algorithms without a teacher. Thus, applying reinforcement learning algorithms with an external teacher can accelerate agent training significantly and improve energy management efficiency at the initial learning stage.

Keywords: energy facility management, reinforcement learning, external teacher, energy consumption optimization.

APPROXIMATION OF THE REACHABILITY SET BOUNDARY BY MACHINE LEARNING METHODS

A.A. Petrov (*aleksand.a.petrov@yandex.ru*)^A
V.V. Latonov (*WLatonov@gmail.com*)^B

^A Moscow State University M.V. Lomonosov, Moscow

^B PJSC Sberbank

In this paper, we solve the problem of estimating the reachability set of a dynamic system using machine learning methods and neural networks. These methods allow us to solve this problem faster than classical numerical methods, such as the pixel method. The paper presents two methods for approximating the boundary of the reachability set: the support vector machine (SVM) and a fully connected neural network with one hidden layer, which are compared with the pixel method. A mathematical model of the Dubins machine in various settings is given, which is used as an example of searching for the reachability set. A method for generating data using the Chebyshev method and schemes for forming training samples are described. Based on numerical experiments, an analysis of the accuracy and speed of the methods, their behavior in the vicinity of the boundary and sensitivity to the sampling method was carried out. The final considered neural network method provided an accuracy of 93% relative to the pixel method, while increasing the speed of operation by more than 3000 times, which indicates the applicability of neural network models for assessing the reachability set in real time (for example, for satellites or autonomous systems).

Keywords: reachability set; Dubins car; boundary approximation; pixel method; support vector machine; neural networks; differential neural networks.

INTELLIGENT SUPPORT FOR SITUATIONAL DECISION-MAKING BASED ON REINFORCEMENT LEARNING

A.P. Shapkin (*shapkinap1999@gmail.com*)
V.V. Borisov (*vbor67@mail.ru*)

Branch of the National Research University
"Moscow Power Engineering Institute", Smolensk

The paper addresses the task of intelligent support for situational decision-making based on a proposed variant of adaptive fuzzy situational models (AFSM) and their reinforcement learning. The study justifies the proposed approach and presents a Deep Recurrent Q-Network-based method for assessing control decisions in AFSM, which takes into account both the history of past control actions and the results of AFSM's structural-parametric adaptation. An example of intelligent support for situational decision-making in the control of a mini power grid is provided, demonstrating the application of the proposed AFSM variant.

Keywords: adaptive fuzzy situational networks, intelligent support of situational decisions, reinforcement learning.

DECISION SUPPORT FOR BALKED LANDING USING PRECEDENTS-BASED REASONING METHOD

B.E. Fedunov (*boris_fed@gosniias.ru*)
A.A. Lipatov (*alexey.lipatov75@mail.ru*)

GosNIIAS, Moscow

In the paper, fragments of the knowledge base for a board-based, rapidly advising expert system are presented. The system is designed to support decision-making by the aircraft crew in a situation of balked landing. The system is supposed to provide recommendations on going around for a second attempt or moving to an alternate airport. The system knowledge base uses the reasoning by precedents method. The content of the knowledge base are formed by accumulating experience of successful decision-making and implementation of such decisions. The knowledge are formalized using the method of decision-making based on precedents. The expert system knowledge base contains the signs that characterize weather conditions (speed of the tailwind), condition of the runway (coefficient of friction) and character of the aircraft movement during an unsuccessful landing attempt (constant longitudinal acceleration). An example of the system's operation is provided.

Keywords: balked landing, decision support, knowledge base, procedures of the knowledge base, recommendations to the crew, reasoning based on precedents.

АВТОРСКИЙ УКАЗАТЕЛЬ

Аверкин А.Н.	97, 108	Латонов В.В.	331
Агафонов А.А.	251	Лебедев Б.К.	307
Алексеев А.К.	97, 108	Лебедев О.Б.	307
Арсеньев Д.Г.	264	Липатов А.А.	352
Байчугуров А.Я.	239	Лихачев М.А.	297
Берёза Д.А.	8	Луценко А.И.	174
Бесхмельнов М.И.	307	Максенков Н.А.	53
Борисов В.В.	20, 343	Марьясин О.Ю.	320
Буханов Д.Г.	132	Микрюкова А.В.	183
Велков А.А.	264	Мисник А.Е.	43, 195, 264
Гладков Л.А.	8	Мищенко И.И.	195
Гладкова Н.В.	8	Муравьев И.П.	108
Дергачев С.А.	183	Муравьев К.Ф.	228
Долгий А.И.	33	Мухаметдинова Л.Р.	239
Ермаков А.О.	65	Нападайло Я.С.	86
Жамал М.	275	Павленко Б.В.	205
Захарченков К.В.	43	Пальчунова О.Д.	120
Зворыкин Д.И.	141	Панов А.И.	216, 275
Зернов М.М.	53	Петров А.А.	331
Зяблова Е.Р.	65	Пикалёв Я.С.	205
Исаев Р.А.	75	Плохотнюк А.Н.	320
Карпов В.Э.	153	Подвесовский А.Г.	75
Ковалев С.М.	33	Поярков Д.И.	216
Копелиович И.А.	75	Романенко В.А.	228
Кошлич Ю.А.	132	Рыбина Г.В.	86
Кузнецов Е.М.	108	Рыченкова К.И.	75
Кузнецов М.В.	164	Сорокоумов П.С.	141
Кулинич А.А.	285	Староверов А.В.	216

Тайлаков В.А.	86	Шалухова М.А.	264
Трофимов Ю.В.	97, 108	Шапкин А.П.	343
Углев В.А.	164	Шевкетова Э.С.	65
Федулов А.С.	20	Шевченко А.В.	97
Федулова С.А.	20	Шеробурко Е.Н.	43
Федунов Б.Е.	352	Яковлев К.С.	174
Христодуло О.И.	239	Яхьяева Г.Э.	120
Черников С.В.	132		

СОДЕРЖАНИЕ

Секция 5. НЕЧЕТКИЕ МОДЕЛИ, МЯГКИЕ ИЗМЕРЕНИЯ И ВЫЧИСЛЕНИЯ, БИОИНСПИРИРОВАННЫЕ МЕТОДЫ

Д.А. Берёза, Л.А. Гладков, Н.В. Гладкова АГЕНТНО-ОРИЕНТИРОВАННЫЙ АЛГОРИТМ РЕШЕНИЯ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ БОЛЬШОЙ РАЗМЕРНОСТИ.....	8
В.В. Борисов, А.С. Федулов, С.А. Федулова МЯГКИЕ СИТУАЦИОННО-КОГНИТИВНЫЕ МОДЕЛИ ДЛЯ ИНТЕЛЛЕКТУАЛЬНОЙ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ.....	20
А.И. Долгий, С.М. Ковалев СТРУКТУРНО-ЧУВСТВИТЕЛЬНЫЙ ПОДХОД К ОРГАНИЗАЦИИ ИНТЕЛЛЕКТУАЛЬНОГО МОНИТОРИНГА НА ОСНОВЕ НЕЧЕТКИХ ЭВОЛЮЦИОНИРУЮЩИХ МОДЕЛЕЙ	33
К.В. Захарченков, А.Е. Мисник, Е.Н. Шеробурко МНОГОУРОВНЕВЫЙ ИЕРАРХИЧЕСКИЙ ЭВОЛЮЦИОННЫЙ АЛГОРИТМ РАЦИОНАЛЬНОГО ВЫБОРА ПАРАМЕТРОВ СИСТЕМЫ УПРАВЛЕНИЯ ПРОДВИЖЕНИЕМ В СОЦИАЛЬНЫХ СЕТЯХ.....	43
М.М. Зернов, Н.А. Максеенков СПОСОБ И ПРОГРАММНАЯ БИБЛИОТЕКА УЧЕТА СТРАТЕГИИ ОГРАНИЧЕНИЙ БАЗОВОГО ДИАПАЗОНА НЕЧЕТКИХ ЧИСЕЛ.....	53
Е.Р. Зяблова, А.О. Ермаков, Э.С. Шевкетова РЕАЛИЗАЦИЯ АЛГОРИТМА РАЗДЕЛЕНИЯ НЕЧЕТКОГО ГРАФА ДЛЯ ФОРМИРОВАНИЯ ЗОН ВЛИЯНИЯ ОБЪЕКТОВ СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМ.....	65
Р.А. Исаев, К.И. Рыченкова, И.А. Копелиович, А.Г. Подвесовский ПОДХОДЫ К УЧЕТУ НЕОПРЕДЕЛЕННОСТИ ЗНАЧЕНИЙ ПАРАМЕТРОВ НЕЧЕТКИХ КОГНИТИВНЫХ КАРТ	75
Я.С. Нападайло, В.А. Тайлаков, Г.В. Рыбина ИНТЕРПРЕТИРУЕМЫЙ МНОГОСЛОЙНЫЙ КЛАССИФИКАТОР НА ОСНОВЕ НЕЧЕТКО-НЕЙРОСЕТЕВОЙ КОГНИТИВНОЙ МОДЕЛИ В ЗАДАЧАХ ТЕХНИЧЕСКОЙ ДИАГНОСТИКИ	86
Ю.В. Трофимов, А.Н. Аверкин, А.К. Алексеев, А.В. Шевченко ИСПОЛЬЗОВАНИЕ НЕЧЕТКИХ МНОЖЕСТВ ДЛЯ ПОСТРОЕНИЯ ГИБРИДНОГО ОБЪЯСНИТЕЛЬНОГО ИСКУССТВЕННОГО ИНТЕЛЛЕКТА НОВОГО ПОКОЛЕНИЯ.....	97

Ю.В. Трофимов, А.Н. Аверкин, И.П. Муравьев, А.К. Алексеев, Е.М. Кузнецов	
СОЗДАНИЕ ПРОТОТИПА РЕКУРСИВНОГО ОБЪЯСНИМОГО ИСКУССТВЕННОГО ИНТЕЛЛЕКТА НА ОСНОВЕ ПОВЕДЕНЧЕСКО-ЛОГИЧЕСКИХ МОДЕЛЕЙ	108

Г.Э. Яхьяева, О.Д. Пальчунова	
О ЧИСЛОВЫХ ХАРАКТЕРИСТИКАХ НЕОПРЕДЕЛЕННОСТИ В КОНЕЧНЫХ РАЗМЫТЫХ МОДЕЛЯХ	120

Секция 6. ИНТЕЛЛЕКТУАЛЬНЫЕ АГЕНТЫ, РОБОТЫ, КОМПЬЮТЕРНОЕ ЗРЕНИЕ

Д.Г. Буханов, С.В. Черников, Ю.А. Кошлич	
КЛАССИФИКАЦИЯ СОСТОЯНИЯ ПТИЦЫ НАПОЛЬНОГО СОДЕРЖАНИЯ НА ОСНОВЕ ИСПОЛЬЗОВАНИЯ YOLO	132

Д.И. Зворыкин, П.С. Сорокоумов	
УСКОРЕНИЕ ОБМЕНА ДАННЫМИ ВОЗЛЕ ПРЕПЯТСТВИЙ ПРИ ГРУППОВОЙ МНОГОАГЕНТНОЙ ФУРАЖИРОВКЕ	141

В.Э. Карпов	
НЕКОТОРЫЕ БИОИНСПИРИРОВАННЫЕ АГЕНТНЫЕ МОДЕЛИ СТРОИТЕЛЬСТВА ГНЕЗД	153

М.В. Кузнецов, В.А. Углев	
ИНТЕРАКТИВНЫЙ СЕРВИС ВИЗУАЛИЗАЦИИ ОБРАЗОВ СЛОЖНЫХ ОБЪЕКТОВ МЕТОДОМ УНИФИЦИРОВАННОГО ГРАФИЧЕСКОГО ВОПЛОЩЕНИЯ АКТИВНОСТИ	164

А.И. Луценко, К.С. Яковлев	
АЛГОРИТМЫ МНОГО-АГЕНТНОГО ПЛАНИРОВАНИЯ ТРАЕКТОРИИ С УЧЕТОМ ВРЕМЕНИ, ЗАТРАЧИВАЕМОГО НА ПОВОРОТЫ	174

А.В. Микрюкова, С.А. Дергачев	
НАВИГАЦИЯ ГРУППЫ ВЗАИМОЗАМЕНЯЕМЫХ АГЕНТОВ В НЕПРЕРЫВНОЙ СРЕДЕ	183

И.И. Мищенко, А.Е. Мисник	
АНСАМБЛЬ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ ИЗОБРАЖЕНИЙ В РАМКАХ СИСТЕМ ПОДДЕРЖКИ ПРИНЯТИЯ ВРАЧЕБНЫХ РЕШЕНИЙ	195

Я.С. Пикалёв, Б.В. Павленко	
РЕГРЕССИОННАЯ НЕЙРОННАЯ СЕТЬ ДЛЯ ОЦЕНКИ ПОЛОЖЕНИЯ БПЛА НА ОСНОВЕ РЕГУЛЯРИЗАЦИИ ФУНКЦИИ ПОТЕРЬ И ВЕСОВЫХ КОЭФФИЦИЕНТОВ	205

Д.И. Поярков, А.В. Староверов, А.И. Панов ПЕРЕХОД С ОФФЛАЙН НА ОНЛАЙН ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ ТРАНСФОРМЕРНЫХ МОДЕЛЕЙ ДЛЯ РОБОТОТЕХНИЧЕСКОЙ МАНИПУЛЯЦИИ	216
В.А. Романенко, К.Ф. Муравьев НАВИГАЦИЯ МОБИЛЬНОГО РОБОТА ДО ЦЕЛЕВОЙ ТОЧКИ С ПОМОЩЬЮ ТОПОЛОГИЧЕСКОЙ КАРТЫ И ОБУЧАЕМЫХ МЕТОДОВ ДВИЖЕНИЯ	228
О.И. Христодуло, А.Я. Байчугуров, Л.Р. Мухаметдинова ГИБРИДНЫЙ ПОДХОД ОБРАБОТКИ ОБЛАКОВ ТОЧЕК ПОМЕЩЕНИЙ НА ОСНОВЕ ГЕОМЕТРИЧЕСКОЙ СЕГМЕНТАЦИИ, КЛАСТЕРИЗАЦИИ И ГЛУБОКОГО ОБУЧЕНИЯ	239

Секция 7. ИНТЕЛЛЕКТУАЛЬНОЕ УПРАВЛЕНИЕ И ПОДДЕРЖКА ПРИНЯТИЯ РЕШЕНИЙ

А.А. Агафонов ИССЛЕДОВАНИЕ СТРАТЕГИЙ ВЗАИМОДЕЙСТВИЯ С БОЛЬШИМИ ЯЗЫКОВЫМИ МОДЕЛЯМИ В ЗАДАЧЕ ОПТИМАЛЬНОГО ВЫБОРА ВРЕМЕНИ ВСТРЕЧИ.....	251
Д.Г. Арсеньев, А.А. Велков, А.Е. Мисник, М.А. Шалухова СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В ДЕТСКОМ СПОРТЕ НА ОСНОВЕ КОМПЬЮТЕРНОГО ЗРЕНИЯ И НЕЙРО-НЕЧЕТКОГО АНАЛИЗА	264
М. Жамал, А.И. Панов ПРИНЯТИЕ РЕШЕНИЙ О ПРОВЕДЕНИИ МАНЕВРА С ПРЕДСКАЗАНИЕМ ТРАЕКТОРНЫХ ПОТОКОВ ДЛЯ БЕСПИЛОТНОГО ТРАНСПОРТНОГО СРЕДСТВА	275
А.А. Кулинич ПРИМЕНЕНИЕ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ В СИСТЕМАХ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ	285
М.А. Лихачев СИНТЕЗ АДАПТИВНОГО УПРАВЛЕНИЯ В МНОГОКАНАЛЬНЫХ НАВИГАЦИОННЫХ ПРИЕМНИКАХ НА ОСНОВЕ ИНТЕЛЛЕКТУАЛЬНОЙ ОБРАБОТКИ ДОКОРРЕЛЯЦИОННЫХ СИГНАЛОВ	297
Б.К. Лебедев, О.Б. Лебедев, М.И. Бесхмельнов АВТОНОМНАЯ МУЛЬТИАГЕНТНАЯ СИСТЕМА УПРАВЛЕНИЯ ГРУППОЙ РОБОТОВ	307

О.Ю. Марьясин, А.Н. Плохотнюк	
ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ С ВНЕШНИМ УЧИТЕЛЕМ ДЛЯ УПРАВЛЕНИЯ ЭНЕРГООБЪЕКТАМИ	320
А.А. Петров, В.В. Латонов	
АППРОКСИМАЦИЯ ГРАНИЦЫ МНОЖЕСТВА ДОСТИЖИМОСТИ МЕТОДАМИ МАШИННОГО ОБУЧЕНИЯ	331
А.П. Шапкин, В.В. Борисов	
ИНТЕЛЛЕКТУАЛЬНАЯ ПОДДЕРЖКА СИТУАЦИОННЫХ РЕШЕНИЙ НА ОСНОВЕ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ	343
Б.Е. Федунев, А.А. Липатов	
ПОДДЕРЖКА ПРИНЯТИЯ РЕШЕНИЙ ПРИ ПРЕРВАННОЙ ПОСАДКЕ С ИСПОЛЬЗОВАНИЕМ МЕТОДА ВЫВОДА НА ОСНОВЕ ПРЕЦЕДЕНТОВ	352
Abstracts of Sectional Reports	361
АВТОРСКИЙ УКАЗАТЕЛЬ	379

Научное издание

Двадцать вторая Национальная конференция
по искусственному интеллекту с международным участием
КИИ-2025
Труды конференции в 3-х томах
Том 2

Подписано в печать 23.09.2025 г.
Формат 60x84¹/₁₆. Тираж 300 экз. Усл. печ. л. 22,4.

Издательство СПб ФИЦ РАН

ISBN 978-5-6052274-4-1



9 785605 227441 >