



XI

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

ИММВ-2022



СБОРНИК НАУЧНЫХ ТРУДОВ

Том 2

**16-19 мая 2022 г.
Коломна**



XI Международная научно-практическая конференция

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

(16-19 МАЯ 2022 г.)

СБОРНИК НАУЧНЫХ ТРУДОВ

Том 2

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

КОЛОМНА
2022

УДК 004.8 : 681.3.01
ББК 32.813

Организаторы конференции:

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

Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов XI Международной научно-практической конференции (ИММВ-2022, Коломна, 16-19 мая 2022 г.). В 2-х томах. Т2. – М.: РАИИ, 2022. – 332 с.

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

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

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

Секция 4 «Инженерия знаний и онтологии»,

Секция 5 «Когнитивные модели в искусственном интеллекте»,

Секция 6 «Гибридные интеллектуальные системы»,

Секция 7 «Интеллектуальные агенты, роботы и коллективы роботов»,

Секция 8 «Интеллектуальное производство, интеллектуальное управление жизненным циклом».

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

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



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

ПРОГРАММНЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

Сопредседатели

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

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

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

Заместители председателей

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

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

Члены программного комитета

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

ОРГАНИЗАЦИОННЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

Сопредседатели

Гречищев А.В. (Россия, Коломна, Глава Коломенского городского округа)

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

Заместители председателей

Королева М.Н., к.т.н., доц. (Россия, Москва, МГТУ им. Н.Э. Баумана)

Мурзак Н.А., к.э.н., доц. (Россия, Коломна, Коломенский институт (филиал) Московского политехнического университета)

Члены организационного комитета

Ефремова А.П., к.т.н. (Россия, Москва, АО «НИИАС»)

Луферов В.С., к.т.н. (Россия, Смоленск, филиал НИУ МЭИ)

Солопов Р.В., к.т.н., доц. (Россия, Смоленск, филиал НИУ МЭИ)

Федотова А.В., к.т.н., доц. (Россия, Москва, МГТУ им. Н.Э. Баумана)

Аникин М.А. (Россия, Москва, МГТУ им. Н.Э. Баумана)

Ахтямова М.Л. (Россия, Москва, МГТУ им. Н.Э. Баумана)

Мажуга В.М. (Россия, Коломна, ООО «Инлайт»)

Тычинская А.М. (Россия, Смоленск, филиал НИУ МЭИ)

ИНФОРМАЦИОННАЯ ПОДДЕРЖКА КОНФЕРЕНЦИИ

ООО «Лаборатория информационных технологий» (Россия, Смоленск)

УДК 004.8

РАЗРАБОТКА МЕТОДА СОЗДАНИЯ ПАЦИЕНТ-ОРИЕНТИРОВАННЫХ РЕКОМЕНДАЦИЙ ПО УПРАВЛЕНИЮ СЕРДЕЧНО-СОСУДИСТЫМ ЗДОРОВЬЕМ У МУЖЧИН*

*Афанасьева Т.В., д.т.н., доцент
РЭУ им. Г.В. Плеханова
e-mail: afanaseva.tv@rea.ru*

1. ВВЕДЕНИЕ

В настоящее время разработка методов и средств автоматизации управления риском сердечно-сосудистых (СС) событий у мужчин является стратегической задачей российского здравоохранения и ключевой задачей превентивной медицины. Отметим, наибольшее число потерянных лет потенциальной жизни обусловлено смертностью среди мужчин трудоспособного возраста (до 60 лет). При этом риск развития инсульта, инфаркта миокарда и сердечно-сосудистой смерти у мужчин, не имеющих доказанного атеросклероза, выше, чем у женщин в 5,5 раз. В рекомендациях Европейского общества кардиологов 2021 года [1] впервые отмечено, что задачу профилактики СС заболеваний (ССЗ) необходимо решать на двух уровнях: на индивидуальном уровне, целью которого является

* Работа выполнена при финансовой поддержке РФФИ, проекты № 20-07-00672 и № 19-47-730001_p_a

максимально возможная персонификация подходов, и на популяционном, в основу которого должны лечь популяционные механизмы контроля факторов риска.

Одной из проблем при современном подходе к профилактике СС осложнений у мужчин, в отличие от женщин, является их низкая мотивированность к заботе о здоровье, систематическому следованию медицинским рекомендациям и управлению ключевыми показателями. В то же время необходимо принимать во внимание и специфические факторы риска у мужчин, такие как хроническая болезнь почек и эректильная дисфункция. Для оценивания риска преждевременной смерти от СС событий разработаны и рекомендованы для практического использования шкалы SCORE, SCORE2 и SCORE-OP [2]. Шкала суммарного сердечного риска SCORE2 оценивает 10-летний риск смерти и смертельных случаев СС3 (инфаркт миокарда, инсульт) у практически здоровых людей в возрасте 40-69 лет с факторами риска, которые не лечились или были стабильными в течение нескольких лет, в то время как шкала SCORE-OP ориентирована для пациентов 70 лет и старше. Анализ типовой формы и структуры клинических рекомендаций [3], а также стандартных алгоритмов оказания медицинской помощи позволяет сделать вывод, что врачу необходимо за небольшое время приема выполнить физикальный анализ, изучить историю болезни пациента, разнородные данные ранее сделанных исследований и медицинских заключений, получить исчерпывающие ответы от пациента на уточняющие вопросы, вычислить суммарный сердечный риск по релевантной шкале и на этой основе сформировать пациент-ориентированные рекомендации, опираясь на национальные руководства [4] и индивидуальную программу профилактики СС заболеваний.

Отметим, что данными для создания отдельных компонент пациент-ориентированных рекомендаций, не связанных с определением диагноза и плана лечения, такими как уровень артериального давления, окружность талии, симптомы и др. владеет сам пациент. Однако люди обычно принимают неправильные решения, связанные с управлением своим здоровьем, по разным причинам, например, из-за напряженного образа жизни, переоценивания ресурсов здоровья и недостатка знаний. С другой стороны, модификация образа жизни, физической активности, питания и следование медицинским рекомендациям зависит от мотивированности и целевой установки пациента; все это выступает значимыми факторами, обеспечивающими профилактику и снижение СС риска.

Одним из перспективных способов повышения эффективности профилактических методов снижения СС заболеваемости и в итоге смертности мужчин, является создание систем поддержки пациентов, основной целью которых будет информирование пациента о состоянии его СС здоровья и формирование пациент-ориентированных рекомендаций по профилактике сердечно-сосудистых событий. На решение этой проблемы направлены исследования, приведенные в этой статье.

В статье предложен метод, позволяющий автоматизировать создание пациент-ориентированных рекомендаций по темпоральным многомерным данным наблюдения за показателями СС здоровья. Пациент-ориентированность рассматривается относительно класса пациента, его возраста, образа жизни, выраженности симптомов, дескриптивных и предиктивных оценок СС здоровья и социально-психологического профиля, влияющего на мотивацию пациента-мужчины к улучшению здоровья. Предложенный метод соответствует направлениям цифрового здравоохранения, ориентированным на пациента, когда пациент должен стать ключевой фигурой в управлении своим здоровьем для предупреждения СС заболеваний. Вовлекая пациента в управление, мониторинг и принятие решений по улучшению своего СС здоровья, можно достичь большего эффекта как в направлении сохранения трудоспособного населения, так и в направлении оптимизации затрат на здравоохранение в целом.

В то же время применение метода автоматизации при формировании пациент-ориентированных рекомендаций по управлению СС здоровья в процессе оказания первичной медицинской помощи обеспечит повышение ее качества и оперативности.

2. ОБЗОР ПОДХОДОВ К СОЗДАНИЮ ПАЦИЕНТ-ОРИЕНТИРОВАННЫХ РЕКОМЕНДАЦИЙ

Согласно обзору [5] системы рекомендаций по здоровью (Health Recommender System, HRS) предлагают потенциал для мотивации и привлечения пациентов к изменению их поведения путем предоставления практических знаний. В исследованиях, посвященных созданию рекомендаций, следует уточнить, кто является целевым пользователем и к кому относятся рекомендации, что рекомендуется и как рекомендации представлены пользователю.

В обзоре [6] HRS сфокусирована на создание рекомендации, касающиеся различных категорий, таких как диеты, физические нагрузки/спорт, рекомендуемые лечение/лекарства для конкретного заболевания, а также медицинскую информацию/источники, которые

должны мотивировать пользователей вести здоровый образ жизни и улучшать свое самочувствие. Исследования, результаты которого опубликованы в статье [7], посвящены решению задачи адаптации рекомендаций для конечных пользователей на основе контекста здоровья, истории и целей. В исследовании [8] авторы сосредоточились на индивидуальном пользователе, анализируя, как HRS могут быть направлены на изменение его поведения. Чтобы рекомендовать персонализированные виды физической активности в соответствии с конкретными потребностями и личными интересами авторы в статье [9] использовали комбинацию рассуждений на основе правил с предпочтениями пациента. В исследовании, опубликованном в статье [10] авторы объединили демографическую информацию и знания, представленные в виде дерева решений для определения состояния здоровья. Так как описание симптомов СС заболеваний пациентом размыты, неполны и неточны, то при оценивании состояния пациента целесообразно применение нечетких моделей. Как отмечается в работе [11] нечеткость симптомов, представлений и оценок в медицине соответствует особенностям картины мира больного и врача с учетом влияния рефлексии. Методы мягких вычислений, применяемых в рекомендательных системах рассмотрены в обзоре [12]. В этом обзоре рассматриваются достоинства и недостатки применения нечеткого моделирования предпочтений пациентов, а также возможности применения функций нечеткого агрегирования и нечеткого расстояния между понятиями. В исследовании [13] описана интеллектуальная система оптимизации здоровья, которая позволяет динамически контролировать состояние здоровья человека. Содержание рекомендаций, получаемые с использованием базы знаний, обновляются по мере поступления новой информации. Система управления автоматически контролирует регулярность поступления новых данных от пациентов из всех источников, включая медицинские карты здоровья. Авторы статьи [14] предложили структуру системы рекомендаций, которая предоставляет информацию по правильному питанию и физической активности. Предложенный в этой статье метод создания рекомендаций основан на модели знаний в виде дерева решений и профиля состояния здоровья пользователя, который содержит данные о возрасте, весе, аллергии, предпочтениях, образе жизни. В работе [15] автор отмечает целесообразность гранулярных вычислений и лингвистических шкал в анализе медицинских данных. Использование нечетких шкал и грануляции данных на основе многозначного контекста для оценивания СС здоровья мужчин рассмотрено в статье [16]. Результат

оценивания СС здоровья представлен в форме абстрактных пропозиций, которые затем интерпретируются предложениями на естественном языке.

3. ПОСТАНОВКА РЕШАЕМОЙ ЗАДАЧИ

Рассматривая процесс создания, пациент-ориентированных рекомендаций по управлению риском СС событий для его последующей автоматизации необходимо определить концепцию, целевого пользователя, входы и выходы этого процесса. В 2009г Американской Ассоциацией Сердца была разработана модель СС здоровья, введено понятие “идеального”, “среднего” и “слабого” СС здоровья. При оценке СС здоровья учитывались интервалы 7 показателей: уровень глюкозы, холестерина, артериального давления, отказ от курения, индекс массы тела, физическая активность и правильное питание [17]. Данная концепция, основанная на идеи оценивать не СС риск или заболевание, а СС здоровье, привлекательна в задаче генерации рекомендации, ориентированных на мужчин, для которых целью и мотивом часто является стремление добиваться большего и демонстрировать «идеальное» здоровье. Предлагаемый в настоящей статье метод генерации пациент-ориентированных рекомендаций ориентирован на следующие типы пользователей:

1. Пациенты, не имеющие подтвержденных ССЗ и отклонений в показателях, желающих сохранить текущий уровень СС здоровья. Будем называть таких пользователей пациентами класса 1 для краткости.
2. Пациенты, не имеющие подтвержденных ССЗ, но имеющие отклонения в некоторых показателях СС здоровья и выполняющие их мониторинг. Пользователей этой группы в дальнейшем будем именовать пациентами класса 2.
3. Пациенты, имеющие ССЗ, находящиеся под наблюдением врача, выполняющие систематический мониторинг показателей и управление риском снижения уровня СС здоровья. Таких пациентов отнесем к классу 3.
4. Медицинские специалисты, к которым обратился пациент за оказанием медицинской помощи, и которые создают медицинские заключения и рекомендации.

Основываясь на российских рекомендациях по профилактике ССЗ [4] и европейских руководствах по профилактике ССЗ [1], предложено использовать в качестве входных данных следующие группы показателей СС здоровья мужчин:

$$M = \{M1, M2, M3, M4, M5, M6\}, \quad (1)$$

где M_1 – это группа показателей клинических исследований, включающая числовые показатели уровень холестерина, систолического и диастолического давления, триглицериды, фракция выброса и индекс массы левого желудочка, креатинин, скорость клубочковой фильтрации, уровень глюкозы;

M_2 определяет множество бинарных показателей наличия отклонений в ЭКГ, имевшихся сопутствующих заболеваний, выявленных ранее и влияющих на повышение СС риска (хронические болезни сердечно-сосудистой системы и сердечно-сосудистые события, эректильная дисфункция, сахарный диабет 2 типа, хроническая болезнь почек, простатит);

M_3 включает показатели, характеризующие образ жизни, выраженные числовыми индексами социальной, физической и профессиональной активности (индекс профессиональной и (или) общественной занятости, признак курения, индекс физической активности);

M_4 – это множество показателей, характеризующих психологические особенности и ценности пациента, определяющие его поведение и отношение к управлению своего здоровья (стиль принятия решений, ценностные ориентации, уровень депрессии, наличие деменции);

M_5 содержит показатели физиологического и антропометрического статуса (возраст, окружность талии, вес, рост или индекс массы тела);

M_6 определяет группу показателей, характеризующих наличие симптомов сердечно-сосудистых заболеваний, выраженные в виде множества проявлений дискомфорта, связанным с СС заболеваниями.

Источником рассмотренных выше данных, описывающих СС здоровье пациента, могут служить результаты домашних измерений, полученных пациентом с помощью бытовых медицинских приборов и гаджетов, результаты лабораторных и инструментальных медицинских исследований, результаты анкетирования, электронные медицинские карты, знания, содержащиеся в руководствах по профилактике ССЗ, в официальных клинических рекомендациях и экспертные знания. Выходными данными, то есть результатом применения метода генерации пациент-ориентированных рекомендаций, является структурированная рекомендация, включающая план эффективных действий и оценочные знания о СС здоровье пациента. Такие знания должны обладать определенной формой, структурой и содержанием.

Контроль и управление риском СС событий рассматривается как итерационный процесс управления по отклонениям, в рамках

которого на каждом шаге (периоде наблюдения) выполняются анализ и оценивание множества показателей и создаются пациент-ориентированные рекомендации. Исходя из этих целевых установок и анализа типовой структуры клинических рекомендаций [3] предложена модель пациент-ориентированных рекомендаций, которая представлена следующими компонентами, зависящими от времени t :

$$\begin{aligned} Rec(t) &= \{Pers(t), Es(t), Pres(t), Med(t)\} \\ Es(t) &= \{SCO(t), Des(t), Pred(t)\} \\ Pres(t) &= \{EsG(t), EsR(t), Motiv(t), Plan(t), Inf(t)\} \end{aligned} \quad (2)$$

Компонента $Pers(t)$ содержит персонализированную информацию о пациенте. Эта компонента может быть скрыта или удалена в целях деперсонализации при обработке данных. Компонента $Es(t)$ включает дескриптивные оценки $Des(t)$ и предиктивные оценки $SCO(t)$ и $Pred(t)$, полученные по множеству показателей СС здоровья с учетом класса пациента. Дескриптивные оценки $Des(t)$ содержат качественные и количественные значения отклонений в значениях показателей, а предиктивные оценки $Pred(t)$ – качественные значения изменений значений показателей. Предиктивные оценки $SCO(t)$ в числовом и качественном виде прогнозируют суммарный риск СС событий в течение последующих 10-ти лет.

Содержание рекомендаций предлагается дифференцировать по классам пациентов и фиксировать в прескриптивном компоненте $Pres(t)$. Так, рекомендации, ориентированные на пациентов класса 1, основаны на данных, содержащихся в оценочных компонентах $SCO(t)$ и $Pred(t)$. Для пациентов класса 3 содержание компонента $Pres(t)$ определяется компонентами $Des(t)$, $SCO(t)$, $Med(t)$ и $Pred(t)$.

Прескриптивный компонент $Pres(t)$ предложенной модели пациент-ориентированных рекомендаций содержит компоненты для управления риском СС событий. Он включает целевые установки стратегического уровня $EsG(t)$ и тактического уровня $EsR(t)$ содержащие целевые значения и тенденции по каждому показателю, требующему коррекции на следующий временной период. Для поддержки пациента в процессе достижения поставленных целей в компонент $Pres(t)$ предложено включить мотивационный $Motiv(t)$ и информационный $Inf(t)$ компоненты, а также индивидуальны план эффективных действий $Plan(t)$. Этот индивидуальный план может включать рекомендации по образу жизни, по оптимизации и коррекции питания, по релевантным физическим нагрузками, график проведения мониторинговых и скрининговых исследований, консультаций медицинских специалистов, в том числе экстренных. Компонент $Med(t)$ содержит данные, определяемые медицинским

специалистом (или интеллектуальным алгоритмом), например, код заболевания по МКБ-10, предварительные диагнозы, уточнение индивидуального плана и т.д. Очевидно, что для создания пациент-ориентированных рекомендаций $Rec(t)$ указанной структуры необходимы знания Z , полученные в рамках доказательной медицины и используемые в практике медицинских исследований [1-4].

Используя введенные обозначения, сформулируем задачу генерации пациент-ориентированных рекомендаций: имея множество темпоральных многомерных числовых данных $X(t, m)$, $t=1,2, \dots$, $m=1,2,\dots,mk$, характеризующих СС здоровье пациента по показателям M (см. выражение 1), и базу медицинских знаний Z требуется получить пациент-ориентированные рекомендации $Rec(t)$ в виде (2), т.е. построить отображение

$$X(t) \times M \times Z \rightarrow Rec(t). \quad (3)$$

4. МЕТОД АВТОМАТИЗИРОВАННОГО СОЗДАНИЯ ПАЦИЕНТ-ОРИЕНТИРОВАННЫХ РЕКОМЕНДАЦИЙ

В рамках поставленной задачи предложен метод генерации пациент-ориентированных рекомендаций для управления риском ССЗ у мужчин, включающий три основных этапа, которые выполняются на каждом t -ом наблюдении за показателями СС здоровья.

Этап 1. Оценивание состояния и динамики СС здоровья на основе анализа темпоральных многомерных данных $X(t, m)$, $t=1,2, \dots$, $m=1,2,\dots,mk$, характеризующих СС здоровье пациента по показателям M , и применения медицинских знаний Z . Результатом оценивания является создание компонента $Es(t)=\{SCO(t), Des(t), Pred(t)\}$ модели рекомендаций (2). Множество медицинских оценочных знаний Z на этом этапе будем рассматривать как элементы обобщенных ограничений в виде лингвистически именованных интервалов, включающие рекомендованные значения, определенные на множестве значений показателей, и правил вывода качественных оценок тенденций. Для получения таких знаний, опубликованных в российских рекомендациях [4] и европейских руководствах по профилактике ССЗ [1] целесообразно применять текстологические методы. На этом этапе важно сначала на основе анализа данных X определить класс и возраст пациента, так как для каждого класса и возраста пациентов необходим свой набор анализируемых показателей и обобщенных ограничений, что определяет и содержание рекомендаций. Так, рекомендации, ориентированные на пациентов класса 1, основаны на анализе возраста, статуса курильщика, уровня холестерина и уровня систолического артериального давления,

необходимых для получения компонента $SCO(t)$, оценивающего суммарный сердечный риск. Для получения количественных и качественных оценок суммарного сердечного риска мужчин в зависимости от возраста рекомендовано применять медицинские шкалы SCORE2 или SCORE-OP [2]. При этом отмечается в [1,4], что отклонения в показателях мочеполовой системы мужчин, повышенный уровень глюкозы, депрессия и увеличенная окружность талии повышают суммарный сердечный риск. При создании дескриптивного компонента $Des(t)$ для показателей группы М6, характеризующих наличие симптомов сердечно-сосудистой недостаточности, предлагается применить опросник и шкалу оценки клинического состояния пациента (ШОКС) [18]. При оценивании показателей группы М4, характеризующих психологические особенности и ценности пациента, целесообразно использовать тест «Ценностные ориентации» М. Рокича [19] и тест на мотивацию при принятии решений мужчинами ТМД/А (Mehrabian Achieving Tendency Scale, MATS) [20]. Чтобы оценить наличие депрессии предлагается использовать гериатрическую шкалу депрессии GDS-15 [21]. Для получения лингвистического выражения компонент $Des(t)$ и $Pred(t)$ конкретного пациента предлагается применить метод, приведенный в статье [16]. В основе этого метода лежит гранулирование данных и представление оценочных знаний в виде пропозиций и протоформ [22], характеризующих наличие отклонений в состоянии и в тенденции СС здоровья пациента [23].

Этап 2. Создание прескриптивного компонента $Pres(t) = \{EsG(t), EsR(t), Motiv(t), Plan(t), Inf(t)\}$ на основе базы медицинских знаний Z и анализа качественных оценок, сформированных в компоненте $Es(t)$.

Структура цели рекомендации включает стратегическую $EsG(t)$ и тактическую цель $EsR(t)$ на следующий период. Стратегическая цель $EsG(t)$ носит обобщенный характер, она представима несколькими установками на действия: (1) по срочному/немедленному обращению за медицинской помощью, (2) по сохранению текущего уровня СС здоровья при отсутствии отклонений в показателях и/или негативных тенденций, (3) по улучшению уровня СС здоровья в противном случае. Тактическая цель $EsR(t)$ описывает какие показатели требуют улучшения. Для генерации мотивационного компонента $Motiv(t)$ на достижение цели, необходимо использовать оценочные оценки показателей группы М4, чтобы определить наиболее значимые ценности и способы принятия решения мужчинами. Это дает возможность в мотивационной части использовать фразы типа «для того, чтобы новое путешествие доставило максимум удовольствия, рекомендуем выполнить рекомендуемый план эффективных действий

по улучшению СС здоровья и качества жизни» или «чтобы быть результативным на работе долгое время, реализуйте рекомендуемый план эффективных действий по достижению целей сохранения уровня СС здоровья и качества жизни». Индивидуальный план эффективных действий $Plan(t)$ включает целевые значения и тенденции показателей, в которых выявлены отклонения, планируемые сроки их достижения и информацию о способах их достижения, включающую рекомендации по образу жизни, физическим нагрузкам, питанию, по консультациям с медицинскими специалистами и справочно-информационный контент. Для создания компонента $Pres(t)$ используются знания Z , извлеченные из официальных клинических рекомендаций и руководств [1-4, 24, 25], с помощью текстологических методов. Для представления таких знаний предлагается использовать продукционную модель на основе иерархии правил.

Этап 3. Уточнение полученных компонент пациент-ориентированных рекомендаций и использование для поддержки принятия врачебных решений. На этом этапе медицинским специалистом (или интеллектуальным алгоритмом) выполняется анализ и добавление информации в компонент $Med(t)$, например, код заболевания по МКБ-10, предварительные диагнозы и т.д. Этот этап предполагает также коррекцию пунктов плана эффективных действий прескриптивного компонента $Pres(t)$ в случае неполноты, неточности, противоречивости и повторения.

5. ЗАКЛЮЧЕНИЕ

Для профилактики, раннего выявления и управления СС здоровьем предлагается метод, позволяющий автоматизировать процесс создания пациент-ориентированных рекомендаций. В отличие от векторной модели пациента, рассмотренной в статье [26] данные о СС здоровье включают временную компоненту, что позволяет извлекать и учитывать тенденции при формировании рекомендаций. В статье предложены постановка задачи автоматизированного получения пациент-ориентированной рекомендации и метод ее решения, который может быть применен при проектировании рекомендательных систем по управлению СС здоровьем. В основе предложенного метода лежит модель показателей СС здоровья и компонентная модель рекомендаций, ориентированные на пациентов-мужчин. В отличие от методов, используемых в рекомендательных системах [5-10, 14], предложенный метод использует расширенную модель показателей СС здоровья, содержащую в том числе социально-психологические показатели и симптомы СС недостаточности, в модель пациент-ориентированных рекомендаций включены дескриптивный, предиктивный и прескриптивный компоненты. Метод создания пациент-ориентированных рекомендаций опирается на

текстологические и коммуникативные методы извлечения знаний и на методы информационной грануляции данных медицинского наблюдения, позволяющие извлекать знания в виде фактов о состоянии и изменении СС здоровья мужчин-пациентов разных классов. Извлеченные факты используются в продукционных моделях для создания пациент-ориентированных рекомендаций. К ограничению предложенного метода автоматизированного создания пациент-ориентированных рекомендаций по управлению сердечно-сосудистым здоровьем следует отнести необходимость предварительного преобразования гетерогенных данных медицинских наблюдений, хранящихся в различных источниках, в однородные темпоральные многомерные числовые данные.

Литература

1. 2021 ESC Guidelines on cardiovascular disease prevention in clinical practice // *European Heart Journal*. – 2021. – Vol.42 (34). – P. 3227-3337.
2. Шкалы SCORE2 и SCORE2-OP для определения сердечно-сосудистого риска. Доступно по адресу: <https://clincasequest.org/score2-score2-op/>
3. Клинические рекомендации (типовая форма). Приказ Минздрава России от 23 июня 2020 г. N 617Н, Приложение 2, изменено с 4 августа 2020 г. 2020. Доступно по адресу: <https://base.garant.ru/72240714/f7ee959fd36b5699076b35abf4f52c5c/>
4. Кардиоваскулярная профилактика 2017. Российские национальные рекомендации // *Российский кардиологический журнал*. – 2018. – Т. 23 (6). – С.7-122.
5. De Croon, R., Van Houdt, L., Htun, N. N., Štiglic, G., Vanden Abeele, V., & Verbert, K. Health Recommender Systems: Systematic Review // *Journal of medical Internet research*. – 2021. – Vol. 23(6).
6. Tran, T.N.T., Felfernig, A., Trattner, C. et al. Recommender systems in the healthcare domain: state-of-the-art and research issues // *J Intell Inf Syst*. – 2021. – Vol.57 – P.171–201.
7. Schäfer H., Hors-Fraile S., Karumur R.P., Valdez A.C., Said A., Torkamaan H., Ulmer T., Trattner C. Towards Health (Aware) Recommender Systems // *Proceedings of the 2017 International Conference on Digital Health; DH'17; July 2-5, 2017; London, USA: ACM*. – 2017. – P. 157–161.
8. Hors-Fraile S., Rivera-Romero O., Schneider F., Fernandez-Luque L., Luna-Perejon F., Civit-Balcells A., de Vries H. Analyzing recommender systems for health promotion using a multidisciplinary taxonomy: a scoping review // *Int J Med Inform*. – 2018. – Vol. 114. P. 143–55.
9. Ali R., Afzal M., Hussain M., Ali M., Siddiqi M.H., Lee S., Ho Kang B. Multimodal hybrid reasoning methodology for personalized wellbeing services // *Comput Biol Med*. – 2016. – Vol. 69. – P. 10–28.
10. Asthana S., Megahed A., Strong R. A Recommendation System for Proactive Health Monitoring Using IoT and Wearable Technologies // *2017 IEEE International Conference on AI & Mobile Services (AIMS)*. – 2017. – P. 14-21.
11. Kobrinskii B.A. Fuzzy and Reflection in the Construction of a Medical Expert System // *Journal of Software Engineering and Applications*. – 2020. – Vol.2. – P.15-23.

12. Toledo, R.Y., & Martínez, L. Fuzzy Tools in Recommender Systems: A Survey // *Int. J. Comput. Intell. Syst.* – 2017. – Vol.10. – P. 776-803.
13. Kobrinский B.A., Grigoriev O.G., Molodchenkov A.I., Smirnov I.V., Blagosklonov N.A. Artificial Intelligence Technologies Application for Personal Health Management // *IFAC-PapersOnLine.* – 2019. – Vol.52, P.70-74.
14. Ahire, Ms.Shreya & Khanuja, Harmeet. HealthCare Recommendation for Personalized Framework // *International Journal of Computer Applications.* – 2015. – Vol.110. – P.24-26.
15. Кобринский Б.А. Субъективность и объективность в мире медицины: взгляд с позиции представления знаний для гибридной интеллектуальной системы // *Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов X-й Международной научно-технической конференции (ИММВ-2021, Коломна, 17-20 мая 2021 г.). В 2-х томах.* – 2021. – Т1. – С.56-67.
16. Afanasieva T. V., Rodionova T. E. Methodology of patient-oriented assessment of cardiovascular health of men using fuzzy sets and formal conceptual analysis // *World Scientific Proceedings Series on Computer Engineering and Information Science/ Developments of Artificial Intelligence Technologies in Computation and Robotics.* – 2020. – P. 857-865.
17. Imaeva A. E., Shalnova S. A. Cardiovascular Health Evaluation: Former Factors – Novel Model // *Cardiovascular Therapy and Prevention.* – 2015. – Vol. 14(5). P. 87–93.
18. Шкала оценки клинического состояния пациента (ШОКС). Доступно по адресу: <https://cr.minzdrav.gov.ru/scale/644e440d-e401-406c-bac1-e53667d381e8>
19. Тест «Ценностные ориентации» М. Рокича. Доступно по адресу: <https://www.psychologos.ru/articles/view/metodika-cennostnye-orientacii-rokicha>
20. Тест на мотивацию при принятии решений мужчинами ТМД/А (Mehrabian Achieving Tendency Scale, MATS). Доступно по адресу: <https://psytests.org/emotional/matsM.html>.
21. Yesavage J.A., Brink T.L., Rose T.L., et al. Development and validation of a geriatric depression screening scale: A preliminary report // *Journal of Psychiatric Research.* – 1983. – Vol.17. – P.37 – 49.
22. Zadeh L.A. Toward a Theory of Fuzzy Information Granulation and its Centrality in Human Reasoning and Fuzzy Logic// *Fuzzy Sets and Systems.* – 1997. – Vol.90. – P.111-127.
23. Afanasieva T., Perfilieva I., Kozhevnikov V. Approach to Patient Assessment Based on a Spatial-Temporal Model for Decision Support Systems in Cardiology. In: Kovalev S., Tarassov V., Snel V., Sukhanov A. (eds) *Proceedings of the Fifth International Scientific Conference “Intelligent Information Technologies for Industry” (ITI’21).* – 2021.–Vol.330.–P.461-471.
24. Рубрикатор клинических рекомендаций Минздрава России. Доступно по адресу: <https://cr.minzdrav.gov.ru/>
25. Рекомендации для пациентов с хронической сердечной недостаточностью. Доступно по адресу: <https://scardio.ru/content/activities/2021/AZ/Recomendation.pdf>
26. Zywicki P., Wójtowicz A., Stachowiak A., Dyczkowski K. Improving medical decisions under incomplete data using interval-valued fuzzy aggregation// *Proceedings of the 2015 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology.* – 2015. Доступно по адресу: <https://doi.org/10.2991/ifsa-eusflat-15.2015.83>

ПРИМЕНЕНИЕ ОНТОЛОГИЧЕСКОГО ИНЖИНИРИНГА В ОБРАЗОВАТЕЛЬНЫХ СИСТЕМАХ*

***Борисов В.В.**, д.т.н., профессор
филиал ФГБОУ ВО «Национальный исследовательский
университет «МЭИ» в г. Смоленске
e-mail: vbor67@mail.ru*

***Мисник А.Е.**, к.т.н.,
МОУ ВО «Белорусско-Российский университет»
e-mail: anton@misnik.by*

1. ВВЕДЕНИЕ

Современные образовательные учреждения можно рассматривать как сложные кибер-физические системы, особенностями которых являются:

- сложность структуры, многокомпонентность, наличие функциональных подсистем, решающих различные целевые задачи, большое количество параметров, характеризующих процессы системы;
- динамическое изменение структуры и параметров системы;
- неполная информация о функционировании и состоянии системы;
- разнообразие внешних и антропогенных воздействий на систему;
- наличие сложных нелинейных связей между параметрами;
- сложность принятия управленческих решений по повышению эффективности системы;
- ограниченные возможности для экспериментальных исследований системы и происходящих процессов;
- невозможность использования единого подхода к созданию моделей исполнительных и информационно-аналитических процессов, происходящих в таких системах.

С 2006 года в Белорусско-Российском университете идёт разработка и развитие систем автоматизации процессов внутри университета.

* Исследование выполнено при финансовой поддержке Министерства науки и высшего образования РФ (государственное задание № FSWF-2020-0019)

В этом году запускается в эксплуатацию система Электронный университет, построенная на принципах онтологического инжиниринга, которая позволит интегрировать в единую систему все службы университета и учебный процесс.

Онтологический подход обладает рядом ключевых преимуществ при автоматизации процессов в образовательных кибер-физических системах, ведь особенностью таких систем является высокий уровень интеграции процессов и совместное использование данных и знаний.

Проведенные исследования позволили обосновать применение метаграфов для соответствующего класса систем с учётом подобранного инструментария, соединяющего онтологическое и метаграфовое представления в рамках единого инструмента, что не достигается традиционным объединением онтологий.

2. ОНТОЛОГИЧЕСКИЙ ИНЖИНИРИНГ

Организация процесса управления в современных образовательных сложных кибер-физических системах является актуальной задачей, решение которой существенно затрудняется увеличением информационных потоков в таких системах. Источниками данных могут быть как исполнительные подсистемы, так и операторы процессов в системе.

В образовательных кибер-физических системах практически каждый пользователь является одновременно как асинхронным источником данных, так и «потребителем» данных.

Для различных сложных образовательных киберфизических систем, работающих в условиях неопределенности, а также тесного взаимодействия основных информационно-аналитических процессов, процессов управления, процессов поддержки принятий решений, процессов документооборота, процессов обучения и усвоения материала и исполнительных процессов, остро стоит проблема повышения эффективности проектирования информационно-аналитических процессов, к которым относятся процессы сбора обработки, обобщения, оценки и прогнозирования состояния систем, выработка обоснованных управленческих решений и оценка их реализуемости.

Одной из наиболее актуальных проблем развития ИТ на данный момент является доведение до широкой эксплуатации таких технологий, которые позволяют строить действительно сложные модели, решать оптимизационные, аналитические и операционные задачи, для решения которых другие средства не подходят [8].

Однако проектирование и моделирование информационно-аналитических процессов невозможно без построения прикладной онтологии предметной области [3].

При разработке сложных кибер-физических систем существует тенденция интеграции разномодальных подсистем в единую систему. Очевидно, что интеграция мультимодальных систем должна осуществляться с использованием единого концептуального интерфейса между ними. В качестве такого интерфейса предлагается использовать онтологию.

В литературе используется достаточно большое количество определений термина онтология.

Онтология – это точная спецификация определенной области, включающая в себя словарь терминов предметной области и набор отношений, описывающих, как эти термины соотносятся друг с другом.

Онтологический инжиниринг — процесс проектирования и разработки онтологий, объединяющий две основные технологии проектирования сложных киберфизических систем — объектно-ориентированный и структурный анализ. Он включает в себя выделение основных классов сущностей при описании реальных взаимодействующих процессов, отношений между этими классами, а также набора свойств, определяющих их изменение и поведение при взаимодействии [9].

Целями онтологического инжиниринга являются повышение уровня интеграции информации, необходимой для принятия управленческих решений, повышение эффективности информационного поиска, предоставление возможности совместной обработки знаний на основе единого семантического описания пространства знаний [10].

Онтология предметной области, особенно в образовательных кибер-физических системах, является весьма динамичной. Для построения такой онтологии необходим соответствующий инструментарий.

3. ВЫБОР ИНСТРУМЕНТАРИЯ ДЛЯ ОНТОЛОГИЧЕСКОГО ИНЖИНИРИНГА

Сегодня графы широко используются для представления данных и знаний в различных областях, таких как: электротехника, строительство и логистика, компьютерные и социальные сети, лингвистика и многие другие.

Теоретико-графовые конструкции (сети Петри, синтаксические деревья, модели программ в виде управляющего графа и т.п.) важны в теории программирования и средств его автоматизации. Многие задачи анализа программ, возникающие при оптимизации, трансляции, валидации, тестировании, значительно упрощаются, если рассматривать их на теоретико-графовых моделях [4].

Хватает ли возможностей классических графов для онтологического инжиниринга? Увы, нет.

Стремление повысить адекватность модели приводит к необходимости использования не только бинарных отношений между компонентами системы (вершинами графа), но и отношений, объединяющих более двух вершин. В этом случае в качестве модели используется гиперграф. Ребро гиперграфа, соединяющее набор вершин, соответствует определенному отношению между данным набором вершин. Это, например, может быть совокупность компонентов (вершин) системы, участвующих в выполнении определенной функции, или совокупность данных, описывающих определенный объект, свойства, события, или совокупность элементов, объединенных в одно целое [11].

Гиперграфическое представление имеет аналогии с семантическими сетями и с фреймами, но фреймы трудно формализовать, а семантические сети до сих пор не имеют стандартного определения, хотя и возникли из формального понятия сети, которое по определению связано с гиперграфом.

Гиперграф же имеет четкое формальное определение и теоретически обоснованную процедуру сопоставления (поиска изоморфизма).

В настоящее время гиперграфы широко используются в тех областях, где требуется анализ сложных связей между вершинами графа, в частности, при проектировании аппаратных средств.

Однако в силу определения гиперграф не обладает достаточно удобными возможностями для моделирования иерархий. В отличие от гиперграфа метаграф представляет собой модель иерархического объекта с разными уровнями обобщения [2].

Метаграфы представляют собой обобщение представлений о структурах графов, используемых в разных областях. Метаграф содержит элементы как орграфов, так и гиперграфов. Сам метаграф строится на основе иерархического графа.

Метаграф — это набор направленных отображений множества в множество. Его приложениями являются моделирование отношений данных, каждое из которых рассматривается как отображение набора

ключевых элементов на набор элементов контента, и моделирование задач рабочего процесса, каждое из которых можно рассматривать как отображение набора элементов. входных документов в набор выходных документов [1].

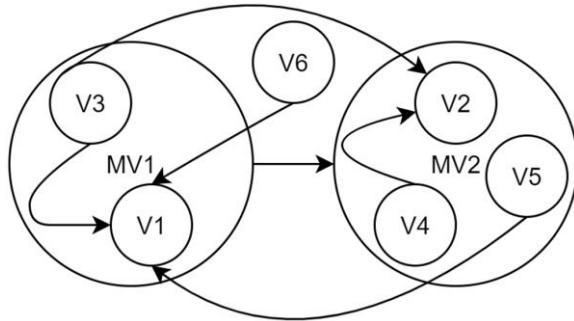


Рис.1. Метаграф

Метаграф - определяется тремя множествами:

$$S = \{V, MB, E\},$$

где V — множество вершин метаграфа, MV — множество метавершин, E — множество ребер. Метавершина метаграфа — это вершина, включающая множество вершин V .

Ребро метаграфа характеризуется исходной и целевой вершинами и направленностью и соединяет две вершины, в отличие от гиперребра гиперграфа, которое может охватывать несколько вершин. Ребро может соединять как вершины, так и метавершины метаграфа.

Таким образом, существуют принципиальные различия между метаграфом и гиперграфом. Гиперграф был изобретен как формализм, позволяющий моделировать сложный порядок обхода вершин графа. Такой подход, в частности, нашел применение при проектировании электронных микросхем. Основное назначение метаграфов — моделирование сложных иерархических объектов и систем.

Вершина метаграфа может быть атрибутивной, то есть с ней может быть связано произвольное количество атрибутов.

Нами предложено обобщить понятия вершины и метавершины до понятия узла метаграфа (N) и расширить определение узла наборами событий и методов, принадлежащих узлу:

$$N = \{N, A, EV, M\},$$

где A — множество атрибутов, связанных с вершиной, EV — набор событий, связанных с узлом метаграфа, M — набор методов, связанных с узлом метаграфа.

Метаграф позволяет определить онтологию системы. Для описания данных системы используются метавершины, позволяющие определять ненаправленные семантические связи между элементами данных. Наличие вложенности, которую обеспечивают метавершины, позволяет моделировать сложные иерархические отношения между элементами данных.

Такой подход к построению метаграфа позволяет включать процессную составляющую в онтологическую модель системы как неотъемлемую ее часть, что позволяет гибко и экономично строить процессные модели на основе сегментов общей онтологической модели.

4. ПОДХОД К ПОСТРОЕНИЮ ИНСТРУМЕНТАРИЯ ДЛЯ ОНТОЛОГИЧЕСКОГО ИНЖИНИРИНГА И УПРАВЛЕНИЯ В КИБЕР-ФИЗИЧЕСКИХ СИСТЕМАХ

Предлагается использовать подход к проектированию информационно-аналитических процессов в условиях необходимости оперативного проектирования, модификации и управления информационно-аналитическими процессами является их создание без привлечения разработчиков [5].

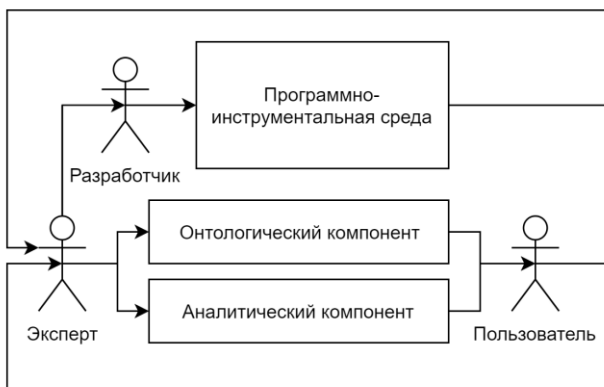


Рис.2. Иллюстрация предлагаемого подхода к проектированию онтологии и процессов в кибер-физической системе

Разработчики создают программно-инструментальную среду на основе онтологического подхода (средства такой среды могут непосредственно оперировать с информационными сущностями

предметной области), а специалисты, используя возможности среды, сами проектируют информационно-аналитические процессы [7].

Преимуществом такого подхода является преодоление «семантического разрыва» между экспертами, архитекторами и разработчиками информационно-аналитических процессов. При этом привлечение разработчиков необходимо только в ситуациях, когда требуется разработать новые или настроить существующие инструменты среды. Для реализации информационно-аналитических процессов специалисты в данной области должны иметь лишь базовые навыки автоматизированного проектирования процессов.

5. СПЕЦИАЛИЗИРОВАННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Разработана специализированная программно-инструментальная среда, в рамках которой возможно моделирование исполнительных процессов и реализация информационно-аналитических процессов сложной киберфизической системы со следующими модулями:

- модуль онтологического инжиниринга, формирующий онтологическую модель сложной кибер-физической системы;
- модуль ввода и хранения пользовательских данных и данных, полученных из внешних источников;
- модуль супервизора нейронной сети для проверки внешних и аналитических данных;
- модуль построения модели информационно-аналитического процесса с использованием механизмов конструкторов на основе темпоральных нейро-нечетких сетей Петри;
- модуль гибкой настройки приоритетного управления информационно-аналитическими процессами, конкурирующими за ресурсы, на основе аппарата темпоральных нейро-нечетких сетей Петри.

Важным компонентом этой среды является модуль онтологического инжиниринга, реализующий принципы построения онтологической модели системы как метаграфа с атрибутивными узлами, включающими события и методы.

Этот модуль позволяет строить структуру метаграфа, задавать наборы атрибутов для его узлов, включать в узел множество узлов метаграфа, включая возможность рекурсивного использования узлов, создавать методы обработки данных и указывать необходимые события для их активации.

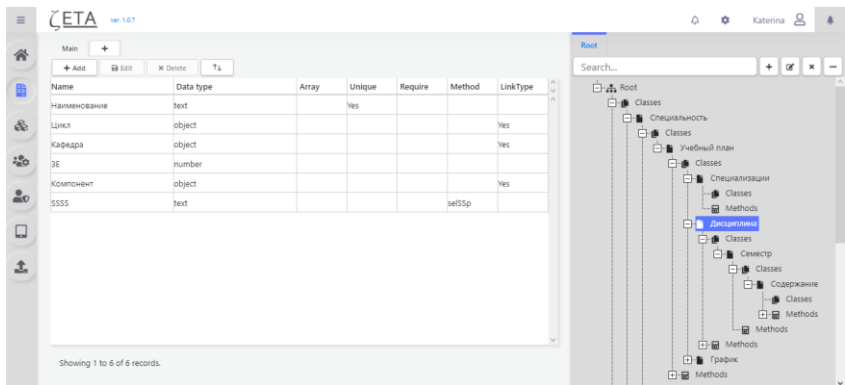


Рис.3. Пример экранной формы модуля онтологического инжиниринга

Разработав прикладную онтологию системы с помощью данного модуля, мы получаем готовый инструмент обработки данных и основу для быстрого формирования моделей системных процессов [6].

Динамичность онтологии системных процессов и траекторий обучения во много продиктована динамичностью системных процессов и индивидуальных траекторий обучения в современных образовательных кибер физических системах.

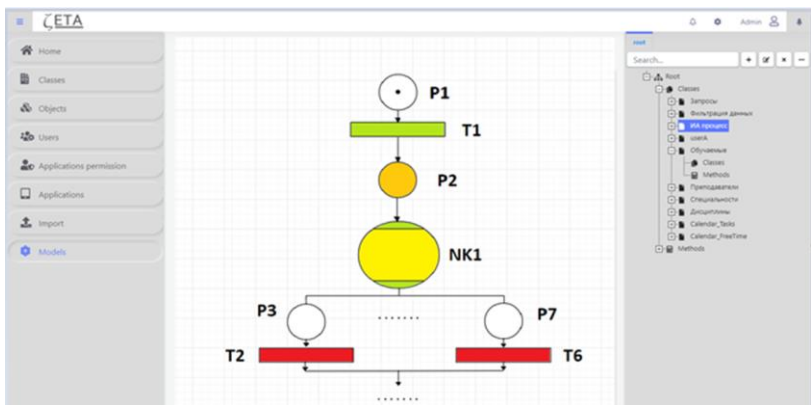


Рис.4. Пример экранной формы построения модели информационно-аналитического процесса с использованием механизмов конструкторов на основе темпоральных нейронечетких сетей Петри

Таким образом, разработанная прикладная онтология используется для построения темпоральных нейро-нечетких сетей Петри для моделирования и управления процессами в системе.

Программно-инструментальная среда в целом и модуль онтологического инжиниринга используют систему управления базой данных PostgreSQL для реализации подсистемы хранения и обработки данных. PostgreSQL — это наиболее полнофункциональная бесплатная система управления базами данных с открытым исходным кодом. Эта система управления базами данных обладает всеми функциями, которые нужны большинству клиентов. PostgreSQL активно используется во всем мире для создания критически важных бизнес-систем под большой нагрузкой.

Использование реляционных СУБД для реализации онтологий, конечно, подвергается определенной критике из-за недостаточно высокой производительности при обходе иерархических моделей, однако в реальной жизни решение вопросов нахождения решений на онтологии занимают весьма небольшое время, в то время как обработка соответствующих массивов данных и их сортировка часто является куда более сложной и ресурсоёмкой задачей и здесь реляционные СУБД существенно выигрывают у альтернатив, рассчитанных только на поиск решений по иерархии отношений.

Фронтенд системы разработан на ReactJS, бэкенд — на PHP с использованием MVC Phalcon.

6. ЗАКЛЮЧЕНИЕ

Для эффективного решения задачи онтологического инжиниринга требуются универсальные средства представления иерархии, атрибутов, событий и методов внутри онтологии.

Рассмотрены графовые инструменты построения онтологий.

Предложены модификации метаграфов для построения прикладной онтологии сложной образовательной киберфизической системы.

Разработанное специализированное программное обеспечение для онтологического инжиниринга рассматривается на примере системы «Электронный университет» Белорусско-Российского университета.

Литература

1. Basu A., Blanning R. *Metagraphs and their applications*. Springer, 2007. 174 p.
2. Самохвалов Э. Н., Ревунков Г. И., Гапанюк Ю. Е. Использование метаграфов для описания семантики и прагматики информационных

- систем // Вестник МГТУ им. Н.Э. Баумана. Серия «Приборостроение». 2015. №1 (100). с. 83-99.
3. Gribova V., Shalfeeva E. Ontological infrastructures for intelligent problem solving // CEUR Workshop Proceedings: Russian Advances in Fuzzy Systems and Soft Computing: Selected Contributions to the 10th International Conference on «Integrated Models and Soft Computing in Artificial Intelligence», IMSC 2021. — 2021. — Vol. 2965.
 4. Bobryakov, A.V. A Method for Managing Engineers Training Processes using Swarm Intelligence Algorithms / A.V. Bobryakov, S.P. Yanukovich, K.V. Zakharchenkov, V.V. Borisov // 2020 5th International Conference on Information Technologies in Engineering Education, Inforino 2020 – Moscow, Russia. 2020. DOI: 10.1109/Inforino48376. 2020.9111788.
 5. Bobryakov A., Misnik A. and Prakapenka S. Management of industrial and technological processes of complex systems based on modified neuro-fuzzy Petri nets // CEUR Workshop Proceedings: Russian Advances in Fuzzy Systems and Soft Computing: Selected Contributions to the 10th International Conference on «Integrated Models and Soft Computing in Artificial Intelligence», IMSC 2021. — 2021. — Vol. 2965. — P. 276-283
 6. Борисов В.В., Мисник А.Е., Прокопенко С.А., Кугузов В.В., Захарченков К.В./ Моделирование образовательных процессов на основе нейронечетких темпоральных сетей Петри // Прикладная информатика. – 2021. – Т. 16. – № 4(94). – С. 35-47. – DOI 10.37791/2687-0649-2021-16-4-35-47.
 7. Bobryakov A. V., Krutalevich S. K., Misnik A. E. and Prokopenko S. A. Modeling of Industrial and Technological Processes in Complex Systems Based on NeuroFuzzy Petri Nets // Journal of Physics: Conference Series, Volume 2096, International Conference on Automatics and Energy (ICAE 2021) 7-8 October 2021, Vladivostok, Russia
 8. Vajnilovich Yu., Zaharchenkov K. and Zakharova A. Integrated System Approach to Improving the Efficiency of IT Projects Management Based on Evolutionary Modeling // Computing in Physics and Technology. pp. 309-314
 9. Gomez-Perez A., Fernández-López M., Corcho O. (2004). Ontological Engineering: With Examples from the Areas of Knowledge Management, E-Commerce and the Semantic Web..
 10. Aguado De Cea, G., Gómez-Pérez, A., Montiel-Ponsoda, E., and Suárez-Figueroa, M.C. Natural Language-Based Approach for Helping in the Reuse of Ontology Design Patterns. In Proceedings of the 16th International Conference on Knowledge Engineering (EKAW),pp. 32-47. (2008).
 11. Soshnikov D. Software Toolkit for Building Embedded and Distributed Knowledge-Based Systems // In Proceedings of the 2nd International Workshop on Computer Science and Information Technologies CSIT' 2000. Ufa, 2000. P. 103-111.

ОСНОВНЫЕ ПОЛОЖЕНИЯ МНОГОМЕРНО-МЕТАГРАФОВОЙ МОДЕЛИ ДАННЫХ И ЗНАНИЙ*

*Гапанюк Ю.Е., к.т.н., доцент
МГТУ им. Н.Э. Баумана
e-mail: garyu@bmstu.ru*

1. ВВЕДЕНИЕ

В современных информационных системах традиционно используются разнородные информационные модели данных, знаний и процессов. Такая ситуация сложилась исторически, потому что раньше мощность вычислительных систем была невысока и во главу угла ставилась производительность обработки информационной модели. Вопросы интеграции информационных систем и унификации информационных моделей оставались на втором плане.

В настоящее время ситуация изменилась. Появление и активное развитие технологий обработки больших данных, расширение круга информационно-аналитических задач привело к тому, что в качестве обрабатываемых данных вполне могут выступать знания, ситуации, процессы. Это требует новых подходов к интеграции информационных систем, к унификации информационных моделей.

В настоящее время в области интеграции информационных ресурсов доминирует сервис-ориентированный подход, в том числе, в его современном микросервисном варианте. Этот подход до определенной степени решает задачу интеграции информационных систем, но при этом каждая система функционирует как черный ящик. Интеграция возможна только на уровне элементов, которые вынесены в интерфейс сервиса. У смежных систем и систем более высокого уровня нет прямого доступа к данным, знаниям и используемым алгоритмам обработки конкретной системы.

Например, если в целях решения информационно-аналитических задач мы хотим использовать OLAP-куб для хранения и агрегации не чисел, а ситуаций или процессов, то использование сервис-ориентированного подхода не поможет, требуется интеграция на уровне единой информационной модели.

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

В данной статье предлагается использование многомерно-метаграфовой модели данных и знаний информационной системы. При этом процесс обработки информации может рассматриваться как последовательность переходов между элементами модели с использованием метаграфовых процессов.

Основой многомерно-метаграфовой модели является модель на основе сложных графов – метаграфовая модель. Метаграфовая модель была изначально предложена А. Базу и Р. Блэннингом в монографии [1] и в дальнейшем получила ряд расширений, независимо предлагаемых различными группами исследователей. В данной статье метаграфовая модель используется в форме аннотируемой метаграфовой модели, предложенной в [2].

Статья организована следующим образом: в разделе 2 рассматриваются основные понятия сложных графов, гиперграфовая и гиперсетевая модели; в разделе 3 – основные положения аннотируемой метаграфовой модели; в разделе 4 – основные принципы многомерно-метаграфовой модели и ее связь с метаграфовыми процессами.

2. ОСНОВНЫЕ ПОНЯТИЯ И РАЗНОВИДНОСТИ СЛОЖНЫХ ГРАФОВ

В настоящее время термины *«сложная сеть»* или *«комплексная сеть»*, которые являются различными переводами англоязычного термина «complex network»), и термин *«сложный граф»* (англ. «complex graph») часто употребляются как синонимы. В работе [3, стр. 14] отмечается, что термин «сложная сеть», как правило, употребляется для обозначения реальной исследуемой системы, в то время как термин «сложный граф» обычно используют для обозначения математической модели такой системы. Как правило, термин «сложный» трактуется в двух вариантах:

I. Плоские графы (сети) очень большой размерности. Такие сети могут включать миллионы и более вершин. Ребра, соединяющие вершины, могут быть ненаправленными или направленными. Иногда используется модель мультиграфа, в этом случае две вершины могут соединяться не одним, а несколькими ребрами.

II. Сложные графы, в которых используется сложное (комплексное) описание вершин, ребер и/или их расположения. Обычно в таких моделях отказываются от плоского расположения вершин и ребер. При этом вершины или ребра могут включать вложенные объекты, создавая эффект эмерджентности. Именно подобные модели могут быть наиболее полезны при описании сложных моделей данных и знаний. На сегодняшний день наиболее

широко известны три подобных модели: *гиперграф*, *гиперсеть* и *метаграф*.

Формализованную модель *гиперграфа* в соответствии с [4] можно представить следующим образом: $HG = \langle V, HE \rangle, he_j = \{v_i\}$, $v_i \in V, he_j \in HE$, где HG – гиперграф; V – множество вершин гиперграфа; HE – множество непустых подмножеств V , называемых гиперребрами; v_i – вершина гиперграфа; he_j – гиперребро гиперграфа.

Гиперграф может быть направленным или ненаправленным. Гиперребра ненаправленного гиперграфа показывают включение вершин, в то время как гиперребра направленного гиперграфа задают порядок обхода вершин.

В соответствии с определением гиперграфа операция иерархической вложенности для гиперребер не определена явным образом, можно говорить только о пересечении множеств вершин, вложенных в гиперребра. Поэтому, хотя гиперграф и содержит гиперребра, но не позволяет моделировать сложные иерархические зависимости и не является полноценной «сложной сетью с эмерджентностью».

Далее рассмотрим более «богатую» гиперсетевую модель, которая строится на основе гиперграфовой модели. Гиперсетевая модель (с одинаковым названием) была открыта дважды: В.К. Попковым и Дж. Джонсоном.

Гиперсетевая модель (по В.К. Попкову) предложена профессором В.К. Попковым. Первые публикации можно отнести к 1980-м годам. В качестве математического аппарата используются теория графов и отчасти теория категорий. Модель применяется в различных прикладных задачах, особенно много применений в области организации транспортных перевозок.

Гиперсетевая модель (по Дж. Джонсону) предложена профессором Джеффри Джонсоном. Основная монография профессора Дж. Джонсона [5], посвященная гиперсетям, вышла в 2013 году. В качестве математического аппарата используются в основном методы топологии (симплициальные комплексы, топологические пространства), метод Q-анализа (предложенный Р. Аткиным). Модель применяется в основном в области социологии.

Рассмотрим определение *абстрактной гиперсети* (по В.К. Попкову) в соответствии с [6].

Пусть даны гиперграфы $PS \equiv WS_0, WS_1, WS_2, \dots, WS_k$. Гиперграф PS или WS_0 называется первичной сетью. Гиперграф WS_i называется вторичной сетью i -го порядка.

Пусть также задана последовательность отображений между сетями различных уровней $\{\Phi_i\}: WS_K \xrightarrow{\Phi_K} WS_{K-1} \xrightarrow{\Phi_{K-1}} \dots WS_1 \xrightarrow{\Phi_1} PS$.

Тогда иерархическую абстрактную гиперсеть порядка K можно определить как $AS^K = \langle PS, WS_1, \dots, WS_K; \Phi_1, \dots, \Phi_K \rangle$.

Эмерджентность в гиперсети (по В.К. Попкову) возникает при переходе между уровнями за счет использования отображений между «слоями» гиперребер. Эмерджентность в гиперсети (по Дж. Джонсону) возникает при переходе между уровнями за счет возникновения гиперсимплексов. В статье [7] профессор К.В. Анохин приводит удачное неформальное определение *гиперсимплекса*: «основание гиперсимплекса содержит множество элементов одного уровня, а его вершина образуется описанием их отношений и приобретает интегральные свойства, делающие ее элементом сети более высокого уровня».

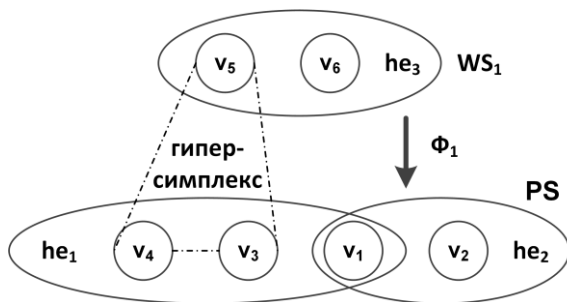


Рис. 1. Пример описания гиперсети

Пример описания гиперсети приведен на рисунке 1. Первичная сеть PS образуется вершинами гиперребер he_1 и he_2 . Первый уровень вторичной сети образуется вершинами гиперребра he_3 . Штрихпунктирной линией выделен гиперсимплекс $v_3-v_4-v_5$ (по Дж. Джонсону). Показано отображение Φ_1 (по В.К. Попкову).

Таким образом, в обеих моделях гиперсеть является конструкцией, содержащей слои гиперграфов, при этом только соседние слои могут быть «сцеплены». Разница между моделями состоит в том, как «сцепляются» слои. В модели В.К. Попкова «сцепление» производится за счет отображений Φ . В модели Дж. Джонсона «сцепление» производится за счет использования гиперсимплексов.

3. ОСНОВНЫЕ ПОЛОЖЕНИЯ АННОТИРУЕМОЙ МЕТАГРАФОВОЙ МОДЕЛИ

Аннотируемая метаграфовая модель, предложенная в [2], расширяет идеи исходной модели А. Базу и Р. Блэннинга [1]. Определим *метаграф* следующим образом: $MG = \langle V, MV, E, ME \rangle$, где MG – метаграф; V – множество вершин метаграфа; MV – множество метавершин метаграфа; E – множество ребер метаграфа; ME – множество метаребер метаграфа.

Вершина метаграфа характеризуется *множеством атрибутов*: $v_i = \{atr_k\}$, $v_i \in V$, где v_i – вершина метаграфа; atr_k – атрибут.

Ребро метаграфа характеризуется множеством атрибутов, исходной и конечной вершиной и признаком направленности: $e_i = \langle v_S, v_E, eo, \{atr_k\} \rangle$, $e_i \in E$, $eo = true \mid false$, где e_i – ребро метаграфа; v_S – исходная вершина (метавершина) ребра; v_E – конечная вершина (метавершина) ребра; eo – признак направленности ребра ($eo=true$ – направленное ребро, $eo=false$ – ненаправленное ребро); atr_k – атрибут.

Фрагмент метаграфа в общем виде может содержать произвольные вершины (метавершины) и ребра (метаребра): $MG_i = \{ev_j\}$, $ev_j \in (V \cup E \cup MV \cup ME)$, где MG_i – фрагмент метаграфа; ev_j – элемент, принадлежащий объединению множеств вершин (метавершин) и ребер (метаребер) метаграфа.

Метавершина метаграфа в дополнение к свойствам вершины включает вложенный фрагмент метаграфа: $mv_i = \langle \{atr_k\}, MG_j \rangle$, $mv_i \in MV$, где mv_i – вершина метаграфа; atr_k – атрибут; MG_j – фрагмент метаграфа.

Наличие у метавершин собственных атрибутов и связей с другими вершинами является важной особенностью метаграфов. Это соответствует принципу эмерджентности, то есть приданию понятию нового качества, несводимости понятия к сумме его составных частей. Фактически, как только вводится новое понятие в виде метавершины, оно «получает право» на собственные свойства, связи и т.д., так как в соответствии с принципом эмерджентности новое понятие обладает новым качеством и не может быть сведено к подграфу базовых понятий.

Таким образом, метаграф можно охарактеризовать как «сложный граф с эмерджентностью» или «сложную сеть с

эмерджентностью», то есть фрагмент сети, состоящий из вершин и связей, может выступать как отдельное целое.

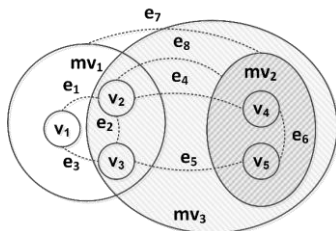


Рис. 2. Пример описания метаграфа в аннотируемой метаграфовой модели

Пример описания метаграфа показан на рис. 2. Данный метаграф содержит вершины, метавершины и ребра. На рис. 2 показаны три метавершины: mv_1 , mv_2 и mv_3 . Метавершина mv_1 включает вершины v_1 , v_2 , v_3 и связывающие их ребра e_1 , e_2 , e_3 . Метавершина mv_2 включает вершины v_4 , v_5 и связывающее их ребро e_6 . Ребра e_4 , e_5 являются примерами ребер, соединяющих вершины v_2 - v_4 и v_3 - v_5 , включенные в различные метавершины mv_1 и mv_2 . Ребро e_7 является примером ребра, соединяющего метавершины mv_1 и mv_2 . Ребро e_8 является примером ребра, соединяющего вершину v_2 и метавершину mv_2 . Метавершина mv_3 включает метавершину mv_2 , вершины v_2 , v_3 и ребро e_2 из метавершины mv_1 , а также ребра e_4 , e_5 , e_8 , что показывает холоническую структуру метаграфа.

Метаребро метаграфа в дополнение к свойствам ребра включает вложенный фрагмент метаграфа:

$$me_i = \langle v_s, v_E, eo, \{atr_k\}, MG_j \rangle, me_i \in ME, eo = true | false,$$

где me_i – метаребро метаграфа, принадлежащее множеству метаребер ME ; v_s – исходная вершина (метавершина) метаребра; v_E – конечная вершина (метавершина) метаребра; eo – признак направленности метаребра ($eo=true$ – направленное метаребро, $eo=false$ – ненаправленное метаребро); atr_k – атрибут; MG_j – фрагмент метаграфа.

Пример описания метаребра метаграфа представлен на рисунке 3. Метаребро содержит метавершины $mv_s, \dots, mv_i, \dots, mv_E$ и связывающие их ребра. Исходная метавершина содержит фрагмент метаграфа. В процессе преобразования исходной метавершины mv_s в конечную метавершину mv_E происходит дополнение содержимого метавершины, добавляются новые вершины, связи, вложенные метавершины.

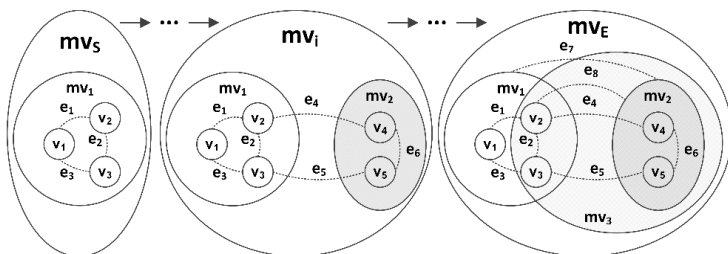


Рис. 3. Пример описания метаребра метаграфа в аннотируемой метаграфовой модели

Если метавершины предназначены прежде всего для описания данных и знаний, то метаребра предназначены в большей степени для описания процессов. Таким образом, аннотируемая метаграфовая модель позволяет в рамках единой модели описывать данные, знания и процессы.

Для того, чтобы объединить метаграфовую модель данных с агентами для ее обработки, введем понятие **активного метаграфа**: $MG^{ACTIVE} = \langle MG^D, AG^{MG} \rangle$, $AG^{MG} = \{ag_i\}$, где MG^{ACTIVE} – активный метаграф; MG^D – метаграф данных; AG^{MG} – множество метаграфовых агентов ag_i , связанных с метаграфом данных.

Рассмотрим особенности моделей гиперграфа, гиперсети и метаграфа с точки зрения реализации на их основе более сложных моделей:

- Гиперграф не позволяет моделировать сложные иерархические зависимости, не является полноценной «сложной сетью с эмерджентностью», что затрудняет создание на его основе более сложных моделей.
- Гиперсеть является «послойным» описанием графов. Предполагается, что слой-гиперграфы идут последовательно и имеют регулярную структуру. Метаграф же позволяет с помощью метавершин группировать произвольные элементы, наличие регулярных уровней не обязательно. Фактически, каждый гиперсимплекс может быть представлен отдельной метавершиной.
- Гиперсеть состоит из разнородных элементов (гиперграфов и отображений по В.К. Попкову, гиперграфов и гиперсимплексов по Дж. Джонсону). Метаграф позволяет с помощью метавершин обеспечивать связь как между элементами одного уровня, так и между элементами различных уровней (при этом не обязательно соседних). Таким образом, метаграфовый подход позволяет

рассматривать сеть не только в виде «горизонтальных» слоев, но и в виде «вертикальных» колонок.

Поэтому именно метаграфовый подход используется для создания на его основе более сложной многомерной (многомерно-метаграфовой) модели.

4. ОСНОВНЫЕ ПРИНЦИПЫ МНОГОМЕРНО-МЕТАГРАФОВОЙ МОДЕЛИ

Общие вопросы многомерной организации метаграфовой информации рассматривались в работах [8, 9], в данной статье полученные ранее результаты систематизируются и объединяются с моделью метаграфовых процессов.

Классическая многомерная модель данных, предложенная Эдгаром Ф. Коддом, позволяет работать с числовыми данными (мерами), привязывая их к иерархическим таксономиям (измерениям) [10]. Многомерная модель данных является ядром информационных систем OLAP (online analytical processing). Было предложено множество вариантов формализации такой модели, но в данном разделе мы используем нашу собственную упрощенную версию формализации в соответствии с [8], которая поможет описать предлагаемую многомерно-метаграфовую модель.

Многомерный гиперкуб (как классический, так и метаграфовый) может быть описан следующим образом:

$$HC = \langle MSR, HCD, HCF, HCR, HCP \rangle, \quad MSR = \{msr_i\}, \quad HCD = \{hcd_i\}, \\ HCF = \{hcf_i\}, \quad HCR = \{hcr_i\}, \quad HCP = \{hcp_i\}, \text{ где } HC - \text{ гиперкуб; } MSR$$

– множество мер (msr_i – мера); HCD – множество измерений (hcd_i – измерение); HCF – множество фактов (hcf_i – факт); HCR – множество правил агрегации (hcr_i – правило); HCP – множество метаграфовых процессов (hcp_i – процесс), данный компонент присутствует только в многомерно-метаграфовой модели (и отсутствует в классической).

Для классической многомерной модели определения могут быть раскрыты следующим образом. Определение измерения гиперкуба:

$$hcd_i = \langle \{hcd_i^k\}, \prec \rangle, \quad hcd_i^k - \text{ элемент измерения гиперкуба; } \prec - \text{ отношение}$$

частичного порядка на множестве элементов измерения гиперкуба.

Определение факта: $hcf_j = \langle \{hcd_i^{ref}\}, \{msr_n\} \rangle$, где hcd_i^{ref} – ссылка на элемент измерения гиперкуба; msr_n – значение меры. Определение правила агрегации:

$$hcr_k : \{hcf_{OUT}\} = agf(\{hcf_{IN}\}, HCD^{ag}), \quad HCD^{ag} \subset HCD,$$

где hcf_{OUT} – выходные (агрегированные) факты; agf – функция агрегации; hcf_{IN} – входные (неагрегированные) факты; HCD^{ag} –

подмножество измерений гиперкуба, принимающее участие в агрегации. Типичными функциями агрегации являются определение суммы, среднего, минимального и максимального значений меры.

На основе формализации для классической многомерной модели может быть предложено описание многомерно-метаграфовой модели.

Мерой в предлагаемой модели является фрагмент метаграфа: $msr_i \equiv MG_j$, где msr_i – мера; MG_j – фрагмент метаграфа. Таким образом, модель может обрабатывать произвольные данные, знания и процессы, представленные в форме метаграфа.

Определение измерения: $hcd_i = \langle \{hcd_i^k\}, \prec \rangle$, $hcd_i \in MV$, $hcd_i^k \in (V \cup MV)$, измерение может быть представлено в виде метавершины, а элемент измерения гиперкуба в виде вложенной вершины или метавершины. Таким образом, измерения гиперкуба могут быть представлены в виде иерархически организованных метавершин. Элементы измерения гиперкуба, соответствующие листьям дерева, могут быть представлены в виде вершин, а элементы более высоких уровней – в виде метавершин.

Определение факта: $hcf_j = \langle \{hcd_i^{ref}\}, \{msr_n\} \rangle$, $hcd_i^{ref} \in (V \cup MV)$, $msr_n \equiv MG_j$. В соответствии с рассмотренными ранее определениями, ссылка на элемент измерения гиперкуба hcd_i^{ref} представляет собой связь с вершиной или метавершиной, которая принадлежит измерению, а мерой msr_n может быть произвольный фрагмент метаграфа.

Определение правила агрегации: $hcr_k : \{hcf_{OUT}\} = ag^{MG}(\{hcf_{IN}\}, HCD^{ag})$, $HCD^{ag} \subset HCD$, вместо функции агрегации agf используется агрегирующий метаграфовый агент ag^{MG} , который выполняет задачу агрегации для произвольного фрагмента метаграфа MG_j , который является мерой.

Можно дать следующее определение метаграфового процесса: $hcr_i = \langle v_S, v_E, eo = true, \{atr_k\}, MG_j [mv_{node} \equiv MG_{node}^{ACTIVE}] \rangle$, $mv_{node} \in MV$, таким образом, метаграфовый процесс является метаребром, в котором узлы mv_{node} вложенного фрагмента метаграфа MG_j являются активными метаграфами MG_{node}^{ACTIVE} . Агенты активного метаграфа обрабатывают данные на соответствующем шаге метаребра, соответствующего метаграфовому процессу. Метаграфовые процессы могут быть использованы для агрегации метаграфовых данных (в этом случае они являются аналогами метаграфовых правил агрегации), для обработки данных на одном уровне многомерно-метаграфовой модели, для детализации данных с переходом на нижние уровни.

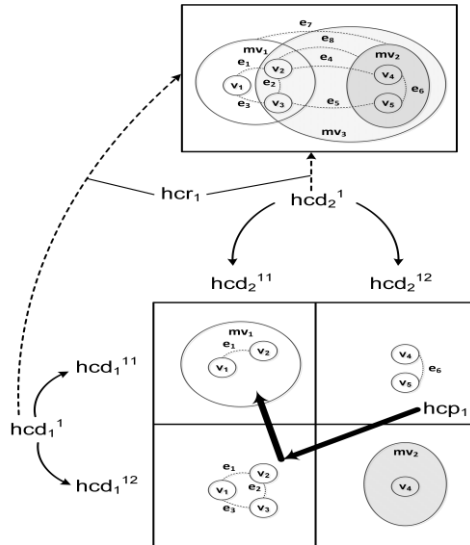


Рис. 4. Пример многомерно-метаграфовой модели

Пример многомерно-метаграфовой модели показан на рис. 4. На рисунке изображен простой гиперкуб с двумя измерениями hcd_1 и hcd_2 . Факты гиперкуба соответствуют элементам измерения гиперкуба. Комбинации элементов hcd_1^{11} - hcd_2^{11} , hcd_1^{11} - hcd_2^{12} , hcd_1^{12} - hcd_2^{11} , hcd_1^{12} - hcd_2^{12} соответствуют фактам гиперкуба нижнего уровня. Комбинация hcd_1^1 - hcd_2^1 соответствует агрегированному факту гиперкуба с правилом агрегации hcr_1 . В процессе агрегации изменяются не только количественные характеристики, но и структура метаграфа для ячеек, соответствующих фактам гиперкуба. Работа правила агрегации hcr_1 показана пунктирными стрелками. Шаги метаграфового процесса hcr_1 (который выполняется на нижнем уровне многомерно-метаграфовой модели,) показаны утолщенными стрелками.

Таким образом, многомерно-метаграфовая модель может обрабатывать произвольные данные, знания и процессы, представленные в форме метаграфа, с использованием агрегации между уровнями модели или с применением метаграфовых процессов.

5. ЗАКЛЮЧЕНИЕ

Среди когнитивных моделей в искусственном интеллекте традиционно важную роль играют графовые модели представления

знаний. В настоящее время повышенное внимание привлекают модели на основе сложных сетей или сложных графов.

Гиперграф не позволяет моделировать сложные иерархические зависимости и не является полноценной «сложной сетью с эмерджентностью». Гиперсеть является «послойным» описанием графов и состоит из разнородных элементов (гиперграфов и отображений или гиперграфов и гиперсимплексов). Метаграфовая модель позволяет с помощью метавершин обеспечивать связь как между элементами одного уровня, так и между элементами различных уровней (при этом не обязательно соседних). Метаграфовый подход позволяет рассматривать сеть не только в виде «горизонтальных» слоев, но и в виде «вертикальных» колонок.

Многомерно-метаграфовая модель включает меры, измерения, факты, правила агрегации и метаграфовые процессы. Метаграфовые процессы могут быть использованы для агрегации метаграфовых данных (в этом случае они являются аналогами метаграфовых правил агрегации), для обработки данных на одном уровне многомерно-метаграфовой модели, для детализации данных с переходом на нижние уровни.

Литература

1. Basu A., Robert W. Blanning. *Metagraphs and their applications.* – New York: Springer, 2007.
2. Гапанюк Ю.Е. Этапы развития метаграфовой модели данных и знаний // В сборнике: Интегрированные модели и мягкие вычисления в искусственном интеллекте (ИММВ-2021). Сборник научных трудов X-й Международной научно-технической конференции. В 2-х томах, Т1. Смоленск, 2021. С.190-200.
3. Chapela V., Regino Criado, Santiago Moral, Miguel Romance. *Intentional risk management through complex networks analysis.* – Springer, 2015: SpringerBriefs in optimization.
4. Voloshin V. I. *Introduction to graph and hypergraph theory.* – New York: Nova Science Publishers, 2009.
5. Johnson J. *Hypernetworks in the science of complex systems.* – London, Hackensack NJ: Imperial College Press, 2013.
6. Попков В.К. *Математические модели связности.* Новосибирск: ИВ-МиМГ СО РАН, 2006.
7. Анохин К.В. Когнитом: гиперсетевая модель мозга // *Нейроинформатика-2015. XVII Всероссийская научно-техническая конференция. Сборник научных трудов.* Ч. 1. М.: МИФИ. 2015. С. 14-15.
8. Chernenkiy V.M., Gapanyuk Y.E., Nardid A.N., Gushcha A.V., Fedorenko Y.S. *The hybrid multidimensional-ontological data model based on metagraph approach // Lecture Notes in Computer Science, Springer.* – 2018. – Vol. 10742. – P. 72-87.
9. Gapanyuk Y.E. *Metagraph approach to the information-analytical systems development // CEUR Workshop Proceedings.* – 2019.–Vol. 2514. – P. 428-439.
10. *Providing OLAP (on-line analytical processing) to user-analysts: An IT mandate.* Technical report, E.F. Codd & Associates, 1993.

МОДЕЛЬ ОНТОЛОГИИ ДЛЯ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ НА ОСНОВЕ НЕЧЕТКОГО ЛОГИЧЕСКОГО ВЫВОДА*

*Гуськов Г.Ю., к.т.н., доцент
e-mail: guskovgleb@gmail.com*

*Романов А.А., к.т.н., доцент
e-mail: romanov73@gmail.com*

*Скалкин А.М., аспирант, м.н.с.
e-mail: anton.skalkin73@gmail.com*

*Строева Ю.В., аспирант, м.н.с.
e-mail: stroeva95@mail.ru*

*Филиппов А.А., к.т.н., доцент
e-mail: al.filippov@ulstu.ru*

Ульяновский государственный технический университет

1. ВВЕДЕНИЕ

Потребность в принятии решений возникает в процессе деятельности любой организации или отдельного человека. Для принятия решения необходимо обладать знаниями об особенностях контекста. Контекст принятия решения определяет условия и ограничения предметной области (ПрО), а также свойства анализируемого объекта и его отношения с другими объектами ПрО. В большинстве случаев человеку проще описать контекст с помощью качественных, а не количественных величин.

Для повышения эффективности процесса принятия решений используются системы поддержки принятия решений (СППР). СППР представляют собой программные системы, целью которых является помощь лицу, принимающему решения (ЛПР), в сложных условиях, для полного и объективного анализа объекта/объектов ПрО. СППР принимает на вход количественные или качественные значения свойств объекта и выдают рекомендации для принятия решений. СППР позволяют скрыть сложность контекста от конечного

* Исследование выполнено при финансовой поддержке РФФИ и Ульяновской области в рамках научных проектов № 19-47-730003 и № 19-47-730005

пользователя и сосредоточиться на процессе управления объектом. ЛППР получает объективную информацию и не вводится в заблуждение большим объемом данных.

Основным компонентом СППР является модуль логического вывода, позволяющий получить результат на основе анализа данных об объекте управления в рамках условий и ограничений некоторого контекста. В некоторых случаях, сведения об объекте управления и/или контексте могут быть представлены с высокой степенью неопределенности на уровне отдельных лингвистических термов: мало, много, средне и т. д. Для работы с такими сведениями используются системы с нечетким логическим выводом (НЛВ).

НЛВ активно применяется в различных интеллектуальных системах. В процессе НЛВ формируются заключения о состоянии некоторого объекта на основе анализа его текущего состояния.

В данной работе рассмотрены новая модель онтологии для поддержки принятия решений на основе НЛВ. Также в работе рассмотрен алгоритм НЛВ, а именно следующие этапы НЛВ: фазификация, агрегирования и активизация.

В качестве примера рассмотрена реализация СППР в задаче балансировки производственных мощностей на основе предложенной модели онтологии для поддержки принятия решений на основе НЛВ.

2. МОДЕЛЬ ОНТОЛОГИИ ДЛЯ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

Онтологические модели представления знаний позволяют объединить преимущества декларативных и продукционных моделей [1]. Онтологии позволяют описать особенности ПрО и набор правил для логического вывода. Атомы правил представляют собой объекты ПрО из состава онтологии.

В основе онтологий лежат различные языки дескрипционной логики (ДЛ). ДЛ позволяют формализовать описание ПрО, гарантировать логическую целостность и непротиворечивость онтологии. Также ДЛ обладают свойствами разрешимости и относительно невысокой вычислительной сложностью основных логических проблем. Данные особенности делают возможным применение ДЛ на практике, обеспечивая компромисс между выразительностью и разрешимостью. Существуют различные по уровню выразительности ДЛ. Базовым семейством ДЛ является *ALC*. Расширения ДЛ *ALC* формируют новые виды логики с более высоким уровнем выразительности. С позиции ДЛ любая онтология

представляет собой совокупность терминологии и утверждений (фактов) [2]:

$$O = \{TBox, ABox\},$$

где $TBox$ – набор терминологических аксиом, представляющих общие знания о понятиях онтологии;

$ABox$ – набор утверждений (фактов) об индивидуальностях онтологии.

В настоящее время в качестве формального языка для представления онтологий используется стандарт OWL 2. В рамках данной работы применяется разновидность языка OWL 2 OWL 2 DL, обеспечивающий выразительность ДЛ $SHOIN(D)$ [3]. Язык SWRL и его расширение SQWRL [4] используются для описания набора логических правил в OWL 2.

Новая модель онтологии для поддержки принятий решений позволяет описывать:

- сущности для организации поддержки принятия решений,
- особенности проблемной области.

Модель онтологии для поддержки принятия решений можно представить в виде следующих аксиом ДЛ:

$$\begin{aligned}
 & Decision \sqsubseteq \top \qquad \qquad \qquad Domain \sqsubseteq \top \qquad \qquad \qquad Inference \sqsubseteq \top \\
 & Recommendations \sqsubseteq Decision \qquad States \sqsubseteq Decision \\
 & Entities \sqsubseteq Domain \qquad \qquad \qquad Objects \sqsubseteq Domain \\
 & Decision \sqcap Domain \sqsubseteq \perp \\
 & Recommendations \sqcap States \sqsubseteq \perp \\
 & Entities \sqcap Objects \sqsubseteq \perp \\
 & Decision \sqsubseteq \exists hasDescription.String \sqcap \forall hasDescription.String \sqcap \\
 & \qquad \qquad \qquad \sqcap = 1 hasDescription.String \\
 & Domain \sqsubseteq \exists hasName.String \sqcap \forall hasName.String \sqcap \\
 & \qquad \qquad \qquad \sqcap = 1 hasName.String \\
 & Entities \sqsubseteq \forall connectToObject.Objects \sqcap \\
 & \qquad \qquad \qquad \sqcap \forall notConnectToObject.Objects \\
 & Objects \sqsubseteq \forall connectToEntity.Entities \sqcap \\
 & \qquad \qquad \qquad \sqcap \forall notConnectToEntity.Entities \sqcap \\
 & \qquad \qquad \qquad \sqcap \exists hasRecommendation.Recommendations \sqcap \\
 & \qquad \qquad \qquad \sqcap \forall hasRecommendation.Recommendations \sqcap \\
 & \qquad \qquad \qquad \sqcap \forall hasNumericValue.Double \sqcap \\
 & \qquad \qquad \qquad \sqcap = 1 hasNumericValue.Double \sqcap \exists hasState.State \sqcap \\
 & \qquad \qquad \qquad \sqcap \forall hasState.State \\
 & Dis(connectToObject, notConnectToObject) \\
 & Dis(connectToEntity, notConnectToEntity) \\
 & Inv(connectToObject) \sqsubseteq connectToEntity \\
 & Inv(notConnectToObject) \sqsubseteq notConnectToEntity,
 \end{aligned}$$

где *Decision* – родительский класс для описания сущностей поддержки принятия решений. Класс *Decision* имеет следующие свойства:

- *hasDescription* – обязательное функциональное свойство для определения текстового описания сущности поддержки принятия решений;
- класс *Decision* объявлен непересекающимся с классом *Domain*. При использовании непересекающихся классов одна и та же индивидуальность не может одновременно принадлежать таким классам;

Domain – родительский класс для описания сущностей проблемной области. Класс *Domain* имеет следующие свойства:

- *hasName* – обязательное функциональное свойство для определения имени сущности предметной области;
- класс *Domain* объявлен непересекающимся с классом *Decision*;

Inference – родительский класс для описания сущностей, описывающих состояния анализируемого объекта и рекомендаций по его управлению;

Recommendations – класс для описания рекомендаций по управлению анализируемым объектом. Класс *Recommendations* наследует свойства класса-родителя *Decision*. Класс *Recommendations* объявлен непересекающимся с классом *States*;

States – класс для описания состояний анализируемого объекта. Класс *States* наследует свойства класса-родителя *Decision*. Класс *States* объявлен непересекающимся с классом *Recommendations*;

Entities – класс для описания сущностей проблемной области, которые оказывают влияние на анализируемый объект. Класс *Entities* имеет следующие свойства:

- класс *Entities* наследует свойства класса-родителя *Domain*;
- класс *Entities* объявлен непересекающимся с классом *Objects*;
- *connectToObject* – свойство для определения связи между индивидуальностями классов *Entities* и *Objects*;
- *notConnectToObject* – свойство для определения отсутствия связи между индивидуальностями классов *Entities* и *Objects*;
- свойство *connectToObject* объявлено непересекающимся со свойством *notConnectToObject*. При использовании непересекающихся свойств одни и те же индивидуальности не

могут быть значениями данных свойств одновременно в рамках одной индивидуальности;

- свойство *connectToObject* является обратным свойству *connectToEntity* класса *Objects*. При использовании обратных свойств достаточно создать экземпляр лишь одного свойства. Экземпляр второго свойства для двух индивидуальностей будет сформирован автоматически в процессе логического вывода;

Objects – класс для описания анализируемых объектов. Класс *Objects* имеет следующие свойства:

- класс *Objects* наследует свойства класса-родителя *Domain*;
- класс *Objects* объявлен непересекающимся с классом *Entities*;
- *connectToEntity* – свойство для определения связи между индивидуальностями классов *Objects* и *Entities*;
- *notConnectToEntity* – свойство для определения отсутствия связи между индивидуальностями классов *Objects* и *Entities*;
- *hasRecommendation* – обязательное свойство для определения рекомендаций по управлению анализируемым объектом;
- *hasNumericValue* – функциональное свойство для определения количественного значения свойства анализируемого объекта. Если анализируемый объект содержит несколько свойств, то каждое свойство определяется в качестве класса-потомка для текущего класса анализируемого объекта;
- *hasState* – обязательное свойство для определения состояния анализируемого объекта или его конкретного свойства;
- свойство *connectToEntity* объявлено непересекающимся со свойством *notConnectToEntity*;
- свойство *connectToEntity* является обратным свойству *connectToObject* класса *Entities*.

3. МЕХАНИЗМ ФОРМИРОВАНИЯ РЕКОМЕНДАЦИЙ

В данном разделе рассмотрен механизм формирования рекомендаций на основе онтологии поддержки принятия решений. В основе механизма формирования рекомендаций лежит множество логических правил на языках SWRL и SQWRL.

Рассмотрим некоторые типы правила и примеры их реализации:

$\sqcap \exists membershipEntity.String \sqcap \forall membershipEntity.String$
Term $\sqsubseteq \exists membershipFunction.MembershipFunction \sqcap$
 $\sqcap \forall membershipFunction.MembershipFunction \sqcap$
 $\sqcap \exists membershipEntity.String \sqcap \forall membershipEntity.String$
 $\top \sqsubseteq \leq 1 Inv(hasMembershipValue).MembershipValue,$

где *FuzzySet* – класс для описания нечетких множеств. Класс *FuzzySet* имеет следующие свойства:

- *hasName* – обязательное функциональное свойство для определения имени нечеткого множества;
- *hasTerm* – свойство для связи нечеткого множества и лингвистического термина;
- класс *FuzzySet* объявлен непересекающимся с классами *MembershipFunction*, *MembershipValue* и *Term*;

MembershipFunction – класс для описания функции принадлежности. Класс *MembershipFunction* имеет следующие свойства:

- класс *MembershipFunction* покрыт классом *Triangular*;
- *triangularValues* – свойство для определения параметров функции принадлежности;
- класс *MembershipFunction* объявлен непересекающимся с классами *FuzzySet*, *MembershipValue* и *Term*;

Triangular – класс для описания треугольной функции принадлежности. Класс *Triangular* имеет следующие свойства:

- *triangularValueMin*, *triangularValueMid*, *triangularValueMax* – функциональные свойства для определения параметров треугольной функции принадлежности. Данные свойства являются потомками свойства *triangularValues*;

MembershipValue – класс для описания степени принадлежности нечеткому множеству. Класс *MembershipValue* имеет следующие свойства:

- *membership* – обязательное свойство для определения значения степени принадлежности;
- *membershipEntity* – обязательное свойство для определения лингвистического термина. В качестве термина может выступать любая сущность онтологии;
- класс *MembershipValue* объявлен непересекающимся с классами *MembershipFunction*, *FuzzySet* и *Term*;

Term – класс для описания связи между лингвистическим термом и функцией принадлежности. Класс *Term* имеет следующие свойства:

- *membershipFunction* – обязательное свойство для определения функции принадлежности;

- *membershipEntity* – обязательное свойство для определения лингвистического термина. В качестве термина может выступать любая сущность онтологии;
 - класс *Term* объявлен непересекающимся с классами *MembershipFunction*, *FuzzySet* и *MembershipValue*;
- hasMembershipValue* – обратное функциональное свойство для определения степени принадлежности любой сущности онтологии нечеткому множеству.

5. АЛГОРИТМ НЕЧЕТКОГО ЛОГИЧЕСКОГО ВЫВОДА

В рамках текущего исследования были реализованы следующие этапы нечеткого вывода: фаззификация, агрегирования и активизация.

Алгоритм этапа фаззификации нечеткого логического вывода:

1. Для каждого количественного показателя, для которого необходимо выполнить фаззификацию, пользователь указывает нечеткое множество. Нечеткое множество определяется названием конкретной индивидуальности класса *FuzzySet*.
2. Для каждой индивидуальности класса *Term*, указанной в качестве значения свойства *hasTerm* нечеткого множества, выполняется расчет степени принадлежности количественного показателя. Для расчета степени принадлежности используется реализация функции принадлежности, указанная в качестве значения свойства *membershipFunction* текущего лингвистического термина.
3. Результат фаззификации записывается через создание индивидуальностей класса *MembershipValue*. Свойство *membership* определяет степень принадлежности, а значение свойства *membershipEntity* копируется из соответствующей индивидуальности класса *Term*.

Алгоритм этапов агрегирования и активизации нечеткого логического вывода:

1. Формируется дерево зависимостей *FT* логических правил [5].
2. Начиная с уровня с наибольшим индексом глубины $i = td(FT)$ производится перебор всех правил данного уровня $\widetilde{FT} = \{ FT_j \in FT \mid j = i \}$.
3. Для каждого правила $\widetilde{FT}_j = \langle A \Rightarrow C \rangle$ из множества \widetilde{FT} определяется:
 - степень истинности условия с помощью операции пересечения нечетких множеств атомов условия: $A^P =$

$\max(A_1, A_2, \dots, A_n)$. Если для атома отсутствует нечеткое множество, то атом не учитывается;

- степень истинности заключения устанавливается равной степени истинности условия: $C^P = A^P$. Если в условии невозможно посчитать степень истинности, то данный шаг пропускается.
4. При переходе на более высокий уровень дерева ($i = i - 1$) определяется степень истинности условий и заключений с учетом расчетов, выполненных на предыдущем шаге.
 5. Алгоритм продолжает работу до достижения корневого узла дерева зависимостей ($i = 0$).

С учетом значений степеней истинности результатов SQWRL правил необходимо отсортировать строки в порядке убывания степени истинности и вывести их.

6. ИЛЛЮСТРАТИВНЫЙ ПРИМЕР ПРИМЕНЕНИЯ РАЗРАБОТАННОЙ МОДЕЛИ ОНТОЛОГИИ

В рамках данного примера будет рассмотрено создание системы поддержки принятия решений для формирования рекомендаций по модификации производства в части балансировки производственных мощностей [6, 7].

В данном примере балансировка производственных мощностей будет производиться на основе числовых значений индикаторов производительности оборудования *ToolPower* и сотрудников *EmployeePower*.

Необходимо добавить следующие аксиомы в онтологию поддержки принятия решений:

- состояния значений индикаторов производства: низко, средне, высоко:

$High \sqsubseteq Inference$ $Low \sqsubseteq Inference$ $Middle \sqsubseteq Inference$;

- состояния производства: плохо или хорошо:

$Bad \sqsubseteq Inference$ $Good \sqsubseteq Inference$;

- текстовые представления значений индикаторов производства, например, для состояния *High*:

$StateHigh: States$ (*StateHigh*, "Высокое значение"): *hasDescription*;

- текстовые представления рекомендаций для ЛПП, например, при состоянии *Bad* индикатора *ToolPower*:

$ToolPowerBadState: Recommendations$

(*ToolPowerBadState*, "Текст рекомендации"): *hasRecommendation*;

- индикаторы производства и их значения:

$EmployeePower \sqsubseteq Objects$ $ToolPower \sqsubseteq Objects$

employeePower: EmployeePower

(employeePower, 4610.41): hasNumericValue

toolPower: ToolPower

(toolPower, 2700.0): hasNumericValue;

- определение нечетких множеств для НЛВ (рис. 1), например, для нечеткого множества *Low* индикатора производства *ToolPower*:

FuzzyToolPower: FuzzySet (Tr1,0): triangularValueMin

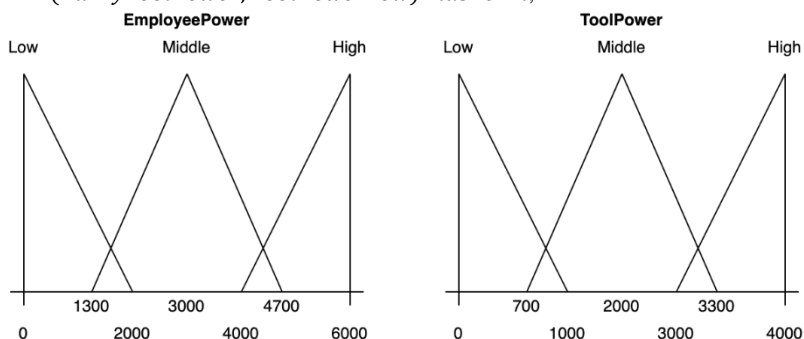
(Tr1,0): triangularValueMid (Tr1,1000): triangularValueMax

ToolPowerLow: Term

(ToolPowerLow,Tr1): membershipFunction

(ToolPowerLow,»Low«): membershipEntity

(FuzzyToolPower,ToolPowerLow): hasTerm;



- создание логических правил, например для индикатора *ToolPower*:

Low(? o) ⇒ hasState(? o, StateLow)

Indicator(? i) ∧ Low(? i) ⇒ Bad(? i)

ToolPower(? tp) ∧ Bad(? tp) ⇒ hasRecommendation(? tp, TPBad)

Таким образом, в результате работы полученной СППР ЛПР получит следующий результат:

*ToolPower*The indicator value is middle0.462

EmployeePower The indicator value is high 0.305

EmployeePower The indicator value is middle0.053

*ToolPower*TPBad Bad recommendation 0.462

EmployeePower EPTPBad Bad recommendation 0.462

*ToolPower*EPTPBad Bad recommendation 0.462

EmployeePower EPGood Good recommendation 0.305

EmployeePower EPBad Bad recommendation 0.053

7. ЗАКЛЮЧЕНИЕ

В статье рассмотрен подход к построению СППР на основе предложенной модели онтологии. Онтология для поддержки принятия

решений позволяет описывать особенности проблемной области и процесса принятия решений. Экспертные правила описываются с применением языка SWRL и его расширения SQWRL. Онтология для НЛВ позволяет выполнять фаззификацию, агрегирование и активизацию.

Литература

1. Gruber T.R. The Role Of Common Ontology In Achieving Sharable, Reusable Knowledge Bases // Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning. 1991. pp. 601-602.
2. Description Logic Programs: Combining Logic Programs With Description Logic / B.N. Grosof, I. Horrocks, R. Volz, S. Decker // Proceedings of the 12th international conference on World Wide Web. 2003. pp. 48-57.
3. The Description Logic Handbook: Theory, Implementation And Applications. / F. Baader, D. Calvanese, D. McGuinness, P. Patel-Schneider, D. Nardi (eds.). – Cambridge university press, 2003.
4. O'Connor M., Das A. SQWRL: a Query Language For OWL // Proceedings of the 6th International Conference on OWL: Experiences and Directions. 2009. Vol. 529. pp. 208-215.
5. Hassanpour S., O'Connor M. J., Das A. K. Visualizing Logical Dependencies In SWRL Rule Bases // Proceedings of the 2010 international conference on Semantic web rules. 2010. pp. 259-272.
6. Модуль поддержки принятия решений информационной среды технологической поддержки производства / А.А. Романов, А.А. Филиппов, Н.Г. Ярушкина, В.А. Маклаев // Автоматизация процессов управления. 2020. №. 2. С.62-72.
7. Romanov A., Yarushkina N., Filippov A. Application Of Time Series Analysis And Forecasting Methods For Enterprise Decision-Management // Proceedings of the International Conference on Artificial Intelligence and Soft Computing. 2020. pp. 326-337.

УСТОЙЧИВОСТЬ ЭМПИРИЧЕСКИХ ЗАВИСИМОСТЕЙ И ПРОБЛЕМА ОБЪЯСНЕНИЯ РЕЗУЛЬТАТОВ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ

*Забезжайло М.И., д.ф-м.н., г.н.с
ФИЦ ИУ РАН
e-mail: m.zabezhailo@yandex.ru*

1. ВВЕДЕНИЕ

Интеллектуальный анализ данных (ИАД) оказался результативным средством решения целого ряда «нагруженных» смысловыми особенностями прикладных задач. Технически (в части математических моделей и алгоритмики) – это, в первую очередь, анализ *семантически* «насыщенных» задач переложенными на компьютер *синтаксическими* средствами. Как обеспечить доверие к порождаемым в процессе ИАД результатам, когда прямая проверка их «приемлемости» не представляется возможной? Например, как «верифицировать» результат классификации новой (не входящей в обучающую выборку) ситуации средствами искусственных нейронных сетей? Говоря формально, здесь приходится иметь дело с неклассической теорией истины (когда проверить достоверность полученного результата в рамках классического подхода А.Тарского [1] не представляется возможным*).

Как следствие, актуальна проблема оценки достаточности оснований для принятия результатов ИАД: синтаксической корректности (отсутствия ошибок в вычислениях) здесь оказывается недостаточно. Действительно,

- ответа на вопрос *КАК?* («вдоль какого *трека*» вычислений) было получено решение оказывается мало в рассматриваемых здесь обстоятельствах. В еще большей степени необходим и ответ на вопрос *ПОЧЕМУ?* именно этот объект следует принять как результат выполненного ИАД;

* Поучительный пример ситуации подобного типа – идентификация Бозона П.Хиггса в адронном коллайдере по виду трека, формируемого проходящей частицей в регистрирующем устройстве.

- появление *артефактов переобучения* (процедурно корректных – сформированных без ошибок в вычислениях, - однако, обладающих неприемлемо низкой обобщающей способностью эмпирических зависимостей [2,3]) требует привлечения семантических соображений для идентификации подобных артефактов и удаления их из финальных результатов ИАД (например, таким способом, как это предложено в [4]);
- особую роль (в связи, в частности, с проблемой ответственности за принимаемые по результатам ИАД решения) приобретает проблема неформальной интерпретируемости получаемых в процессе ИАД результатов.

Своего рода «вишенкой на торте» здесь оказывается отсутствие возможностей при анализе открытых предметных областей формировать дедуктивными средствами доказательства семантической корректности результатов ИАД, т.к. в общем случае не представляется возможным обосновать утверждения класса теорем о полноте и непротиворечивости.

Результативным ответом на вопрос *ЧТО ДЕЛАТЬ?* в описываемой ситуации может, в частности, быть выявление в анализируемых данных эмпирических зависимостей причинно-следственного типа, чтобы с их помощью отслеживать и объяснять возникновение изучаемых целевых эффектов.

Однако, следует учитывать, что в общем случае при таком подходе нам доступны возможности породить лишь *приближенные* описания подобных причинно-следственных зависимостей. По факту это – ИАД-интерполяции исходных данных с точностью до:

- обучающей выборки прецедентов (в том числе - и ее последующих конкретных расширений новыми данными), а также
- выбираемого языка представления знаний (ЯПЗ) об анализируемых прецедентах и целевом эффекте.

Критически значимая роль при уточнении формируемых приближений выпадает на эффекты устойчивости (наследуемости) порождаемых эмпирических зависимостей причинно-следственного типа как при заданных расширениях обучающей выборки, так и при варьировании используемых языков представления знаний. Неформальная интерпретируемость как наследуемости, так и выявляемых случаев ненаследуемости порождаемых эмпирических зависимостей (ЭЗ) – важная составная часть *объяснимости* формируемых результатов ИАД (как значимый фактор оценки *доверия* – достаточности основания для их принятия).

2. ЭФФЕКТЫ УСТОЙЧИВОСТИ (НАСЛЕДУЕМОСТИ) ЭМПИРИЧЕСКИХ ЗАВИСИМОСТЕЙ, ПОРОЖДАЕМЫХ В ПРОЦЕССЕ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ

В целом объяснимость формируемых в машинном обучении и интеллектуальном анализе данных заключений – критически значимый компонент оценивания достаточности оснований для принятия получаемых результатов. И это легко понять, ведь в конечном итоге ответственность за последствия принятых в подобных ситуациях решений ложится не на компьютерную систему, а на человека (так называемое ЛПР - Лицо, Принимающее Решение). Как следствие, готовность ЛПР принять на себя эту ответственность в значительной мере зависит от того, в какой мере основанное на предлагаемых системой ИАД результатах решение приемлемо для ЛПР - содержательно объяснимо и неформально интерпретируемо в терминах и понятиях анализируемой предметной области.

Дополнительные особенности обсуждаемой проблематике придает проблема устойчивости (наследуемости) ранее уже сформированных объяснений при получении новой информации об исследуемой проблеме. Обратимся, например, к области медицины - формированию *персонализированного* варианта рекомендаций по терапии, в основанном на выделении в процессе ИАД имеющихся текущих сведений о пациенте тех и только тех факторов, влиянием которых в данном случае и обусловлено возникновение анализируемого заболевания. В этой ситуации представляется вполне естественным формировать искомую персонализированную терапию на основе целенаправленного противодействия идентифицированным факторам влияния. Как обеспечить наследуемость таких рекомендаций при расширении текущих данных о пациенте новыми сведениями?

В чем, собственно, здесь проблема? Известно, что характерная для машинного обучения и ИАД интерполяционно-экстраполяционная техника анализа данных в общем случае не гарантирует наследуемости построенной интерполяции уже имеющейся обучающей выборки эмпирическими зависимостями произвольного типа в ситуации, когда эта выборка пополняется теми или иными конкретными новыми данными. В целом ряде случаев (как, например, в только что рассмотренной выше задаче медицинского ИАД), необходимо обеспечить устойчивость (наследуемость) хотя бы части формируемой интерполяции исходной обучающей выборки при ее расширении новыми сведениями. С процедурной точки зрения – это проблема

выделения таких типов интерполяционных зависимостей, которые давали бы возможность идентифицировать в уже сформированных интерполяциях наследуемые фрагменты (множества устойчивых - наследуемых - эмпирических зависимостей, содержащихся в неявной форме в описании данных анализируемой обучающей выборки).

Несложно указать некоторые семантические основания для обсуждаемой наследуемости: например, это могут быть анализ причинности (т.е. идентификация факторов влияния, «вынуждающих» появление анализируемого эффекта) и последующее восстановление причинно-следственных связей, скрытых в анализируемых данных. Объяснения в таких ситуациях можно строить с использованием выявленных в процессе ИАД причинно-следственных зависимостей. Очевидным примером здесь может служить обсуждаемая выше проблематика медицинской диагностики на постоянно расширяемых - пополняемых описаниями новых пациентов - обучающих выборках. Именно наследуемость причинной обусловленности возникновения изучаемого заболевания дает надежды на успех в выделении соответствующих факторов влияния и причинно-следственных эмпирических зависимостей, которые устойчивы относительно расширений текущей обучающей выборки описаний уже изученных прецедентов (данных о пациентах) новой информацией.

Разумеется, успешная реализация этих надежд существенным образом зависит от выразимости этих факторов причинно-следственного влияния в используемом языке представления знаний (ЯПЗ) об изучаемых эффектах. Именно проблема выбора адекватных (и в первую* очередь - в части предлагаемых ими дескриптивных возможностей) средств представления данных и знаний об анализируемой предметной области оказывается критичной для результативного выявления причинно-следственных зависимостей, скрытых в накапливаемых исходных данных. Полезным примером здесь могут служить, в частности, языки представления знаний о химических структурах в задаче анализа связи СТРУКТУРА => СВОЙСТВА физиологически активных химических соединений: это могут быть языки графов химических структур – плоские симметризованные описания, пространственные представления, пространственные графы с углами и расстояниями между вершинами, ... ; покрытия графа химической структуры подграфами из заданного множества; спектры – излучения или поглощения, а также многое другое. Приближенный характер таких средств представления знаний

* Не менее значимы также и те характеристики ЯПЗ, которые определяют объемы необходимых при его использовании вычислений.

оказывается полезным, в частности, при балансировке взаимосвязи *точности описания* и *вычислительной сложности* обработки анализируемых объектов – формализованных для использования в компьютерном анализе представлений структур физиологически активных химических соединений.

Комбинируя два рассматриваемых аспекта – наследуемость порождаемых средствами ИАД эмпирических зависимостей при заданном расширении обучающей выборки описаниями новых прецедентов и возможности варьирования используемого ЯПЗ, можно предложить эвристику для анализа устойчивости формируемых причинно-следственных зависимостей:

- выберем несколько вариантов языка представления знаний о прецедентах, собираемых в обучающую выборку и ее расширения, и
- будем искать эмпирические зависимости, которые связывают одни и те же прецеденты при интерполяции этой выборки.

При этом будем дополнительно выявлять такие «устойчивые» при варьировании (по используемому множеству возможностей) языка представления знаний зависимости, которые также наследуемы при заданном конкретном расширении текущей обучающей выборки описаниями конкретного набора новых прецедентов. Таким образом, в центре внимания окажутся

- «инструментальные» средства (математические модели, методы и алгоритмы) выявления устойчивых эмпирических зависимостей причинно-следственного типа, а также
- идентифицированные с их помощью в конкретной ситуации наследуемые множества зависимостей рассматриваемого вида.

3. НЕСКОЛЬКО ИЛЛЮСТРАЦИОННЫХ ПРИМЕРОВ

Обсудим на нескольких демонстрационных примерах как взаимосвязаны эффекты устойчивости (наследуемости) эмпирических зависимостей причинно-следственного типа одновременно в двух «измерениях» - при варьировании средств представления знаний и при пополнении заданной обучающей выборки конкретными новыми данными.

В качестве примера множества (иерархии) взаимосвязанных ЯПЗ для описания прецедентов в текущей обучающей выборке - Базе Фактов – могут, в частности, быть использованы следующие варианты параметризации анализируемых данных:

- каждый прецедент представлен как
- множество признаков (теоретико-множественная параметризация),

- множество признаков с их весами (параметризация средствами так называемых нумерованных множеств - мультимножеств),
- множество признаков вместе с отношениями на этих признаках («простая» графовая параметризация),
- множество признаков с отношениями на них, а также - метками на вершинах и ребрах порождаемых этим способом графов.

Цель предлагаемого рассмотрения - предъявить такие средства выявления причинно-следственных зависимостей, изначально скрытых в анализируемых эмпирических данных, которые «однородным» образом порождали бы искомые эмпирические зависимости для всех задействованных языков представления данных и позволяли бы выделять из множества таких зависимостей те, которые наследуются (устойчивы) при пополнении текущей обучающей выборки описаниями конкретного набора новых прецедентов.

Задействованная для достижения этой цели процедурная техника ИАД будет использовать конструкции ДСМ-метода автоматизированной поддержки исследований [5] как понятийного инструментария для формирования Характеристических Функций [6,7], в свою очередь описывающих формальными средствами специальный класс эмпирических зависимостей причинно-следственного типа.

Рассмотрим несколько простых модельных ситуаций с наследуемостью зависимостей при переходе от ЯПЗ теоретико-множественной параметризации (будем называть его языком M - множеств) к языку мультимножеств – MM . (Дальнейшее движение к графовым представлениям – см. выше – может быть организовано подобным же образом).

Пусть задано множество образующих $U = \{a_1, a_2, \dots, a_n\}$ а также - две пары не имеющих общих элементов подмножеств Ω^+, Ω^- и $\Delta\Omega^+, \Delta\Omega^-$ множества $2^U \setminus \emptyset$ всех непустых подмножеств множества U . При этом $\Omega^+ = \{O_1^+, O_2^+, \dots, O_p^+\}$ – множество примеров, а $\Omega^- = \{O_1^-, O_2^-, \dots, O_q^-\}$ – множество контрпримеров исходной Базы Фактов $\Omega = \Omega^+ \cup \Omega^-$, а множества $\Delta\Omega^+ = \{O_1^+, O_2^+, \dots, O_r^+\}$ и $\Delta\Omega^- = \{O_1^-, O_2^-, \dots, O_s^-\}$ представляют собою соответственно множество примеров и контрпримеров в заданном расширении $\Delta\Omega = \Delta\Omega^+ \cup \Delta\Omega^-$ исходной Базы Фактов Ω . (Ради упрощения изложения будет достаточно рассмотреть примеры как прецеденты наличия целевого свойства, а контрпримеры – прецеденты его отсутствия). Для перехода из языка M в язык MM можно кодировать кратные вхождения образующих в расширенное множество U_{MM} наборами новых образующих так, что

каждому элементу a_i из исходного множества U , преобразуемому в языке MM в $a_i(m)$, вместе с $a_i(m)$ в MM будет сопоставлен набор дополнительных образующих $a_i(1), a_i(2), \dots, a_i(m-1)$. Таким образом, теоретико-множественные операции объединения и пересечения в языке MM можно будет описать расширением обычных теоретико-множественных операций взятием максимума и минимума по соответствующим значениям m – «весов» соответствующих образующих $a_i(m)$ в задействованных операндах.

Несложно убедиться, что на описанной в языке M Базе Фактов Ω выводимость средствами (см. [5]) *Простого Сходства с Запретом на КонтрПримеры (ЗКП)* некоторой зависимости V влечет также выводимость «взвешенного» аналога V и в языке MM , причем - при любом распределении весов m на образующих из U_{MM} . Действительно, непустота сходства V наследуется из M в MM , а выполнение условия *ЗКП* в M означает, что V не содержится ни в одном из контрпримеров из Ω . Таким образом, V не содержится даже в MM -описаниях контрпримеров из Ω с минимальным весом $m=1$. Т.е. – условие *ЗКП* справедливо и для аналога зависимости V в MM -описании Базы Фактов Ω .

Однако, в ситуации, когда в M условие *ЗКП* не выполняется, при переходе к MM -описанию возможны оба исхода – и выводимость, и невыводимость MM -аналога зависимости V . Так в ситуации, когда «вес» m^- образующих в контрпримере O_j^- , «проваливающим» в M условие *ЗКП*, окажется меньше соответствующего «веса» m^+ этих образующих в сходстве V , в MM будет выводиться V , не взирая на невыводимость его аналога в M . При обратном соотношении «весов» невыводимость V будет наследоваться с M на MM .

Похожая ситуация, когда переносимость порождаемых в процессе ИАД заключений определяется контекстными особенностями анализируемой Базы Фактов, имеет место и при расширении исходной выборки прецедентов Ω новой информацией $\Delta\Omega$. Также, как было показано выше, можно привести примеры, когда

- зависимость V наследуется из Ω в расширение $\Omega \cup \Delta\Omega$, и при этом MM -аналог V также наследуется из Ω в $\Omega \cup \Delta\Omega$;

и, наоборот,

- зависимость V наследуется из Ω в расширение $\Omega \cup \Delta\Omega$, а MM -аналог V не наследуется из Ω в $\Omega \cup \Delta\Omega$. (Такая ситуация может возникнуть, если в расширении $\Delta\Omega$ найдется контрпример O_0^- , «проваливающий» условие *ЗКП* для соответствующего V , например, за счет аналогичных представленным выше

соотношений между «весами» образующих собственно в сходстве V и в контрпримере O_0).

Итак, для оценки устойчивости причинно-следственных эмпирических зависимостей, необходимых для формирования объяснений получаемых на конкретной Базе Фактов результатов ИАД, в общем случае может потребоваться исчерпывающий экспериментальный обзор всех возможных вариантов порождения эмпирических зависимостей на исходной Базе Фактов и ее расширении, представляемом различными ЯПЗ. Естественно, что возможности разрешить хотя бы некоторые возникающие здесь переборные задачи «быстрыми» алгоритмами существенным образом упрощают поиск надежных объяснений, одновременно позволяя выбрать и адекватные* целям и задача ИАД языки представления знаний об анализируемых эффектах. Примеры эффективно разрешимых задач при поиске устойчивых эмпирических зависимостей использованием математической техники Характеристических Функций можно найти в работах [6,7].

4. ЗАКЛЮЧЕНИЕ

К.Поппер, рассматривая вопрос о доверии к эмпирическим теориям [8], обратил внимание на асимметричность эффектов двух типов: 1) подтверждение теории теми или иными новыми данными и 2) опровержение теории конкретными новыми фактами. Он показал, что эффект опровержения (т.е. явной *фальсифицируемости* теории) некоторыми конкретными данными имеет приоритетное значение. Подтверждение представляет интерес лишь как демонстрация сохранения состояния нефальсифицируемости (потенциально фальсифицируемой) теории вновь полученными сведениями. Расширение исходной обучающей выборки в процессе ИАД может играть особую роль в условиях, когда фальсификация потенциально возможна, однако, в текущем рассматриваемом вовлечении в интеллектуальный анализ новых данных не происходит утраты наследуемости хотя бы части из ранее уже сформированных эмпирических зависимостей. Именно такие - устойчивые к конкретному расширению - эмпирические зависимости дают в рассматриваемой конкретной ситуации более точное описание причинно-следственных связей, изначально скрытых в исходных данных.

* Например, обеспечивающие разумный баланс между детальностью описаний и объемами вычислений, которые необходимы для поиска соответствующих решений.

Ненаследуемость найденной эмпирической зависимости, связывающей конкретный набор прецедентов в двух «измерениях» - относительно выбора ЯПЗ и относительно конкретного расширения исходной обучающей выборки (т.е. текущего состояния постоянно накапливаемой Базы Фактов), может рассматриваться как значимый аргумент в пользу необходимости провести дополнительный анализ достаточности оснований для принятия формируемых в процессе ИАД заключений. В самом общем виде – это сигнал о необходимости провести оценку альтернатив:

- порожден артефакт (нет возможности неформальным образом интерпретировать порожденный результат),
или же
- выявлен дефект в используемом языке представления знаний (например, конкретный вариант ЯПЗ оказался слишком «груб» - недостаточно «выразителен», чтобы адекватно представить\описать некоторый существенный для данной ситуации содержательный эффект).

Предлагаемым путем (в том числе – на основе содержательной объяснимости и интерпретируемости заключений ИАД) можно оценивать доверие к порождаемым результатам, формируемое на базе причинно-следственных связей, которые изначально скрыты в анализируемых данных, однако, реконструируемы из них средствами интеллектуального анализа данных. Основой для формирования доверия к получаемым результатам ИАД в данном случае оказываются эмпирические описания причин – приближенные представления о факторах влияния, вызывающих целевой эффект. Своего рода «драйвером» предлагаемого уточнения представления знаний о причинности здесь оказывается именно приближенный характер описаний выявляемых эмпирических причин. В каждом конкретном случае они представлены с точностью до «содержания» текущего состояния исходной Базы Фактов («представительности» и «репрезентативности» текущей Базы Фактов по отношению к скрытым в ней взаимосвязям факторов влияния, вызывающим целевой эффект), а также – до выразительных возможностей описания искомым конкретным факторам влияния и зависимостей между ними дескриптивными средствами соответствующего ЯПЗ. Значимым результатом предлагаемого здесь анализа устойчивости* формируемых эмпирических зависимостей оказываются также и выявляемые в процессе ИАД дополнительные основания для выбора адекватного

* См. например, учет особой роли ситуаций с невыполнением условия *Запрета* на *КонтрПримеры*, представленных выше (в Разделе 3).

ЯПЗ, релевантного задачам и требованиям детальности изучения конкретного целевого эффекта.

Наконец, предлагаема эвристика может быть дополнительно расширена еще одним «измерением» – возможностями варьировать используемые при поиске эмпирических зависимостей типы формализации отношения причинности. Так, в частности (см. примеры из Раздела 3), здесь могут быть использованы так называемые *прямые*, и *обратные* зависимости в ДСМ-методе [5], *тернарное* отношение причинности [9] и др.

Литература

1. Tarski A. The Semantic Conception of Truth and the Foundations of Semantics. // Philosophy and Phenomenological Research, 1944, v. 4, n. 3, pp. 341–375. - <https://sites.ualberta.ca/~francisp/Phil426/TarskiTruth1944.pdf>
2. Воронцов К.В. Комбинаторная теория надёжности обучения по прецедентам. – Дисс. на соиск. учёной степени д.ф.м.н. по специальности 05.13.17 Теор. основы информатики. – М: ВЦ РАН, 2010. - 273 С. – URL: <https://www.dissercat.com/content/kombinatornaya-teoriya-nadezhnosti-obucheniya-po-pretседentam>
3. Виноградов Д.В. Вероятностно-комбинаторный формальный метод обучения, основанный на теории решеток. – Дисс. на соиск. уч. ст. д.ф.м.н. по спец. 05.13.17 Теор. основы информ. – М.:ФИЦ ИУ РАН, 2018. – 131 с. – URL: http://www.frccsc.ru/diss-council/00207305/diss/list/vinogradov_dv
4. Забейхайло М.И., Трунин Ю.Ю. О значимости эмпирического противоречия в оценке надежности результатов интеллектуального анализа данных // НТИ, сер.2 – 2021, № 6, стр. 14-20.
5. Финн В.К. Индуктивные методы Д.С.Милля в системах искусственного интеллекта // Искусственный интеллект и принятие решений. - 2010. Ч. I. № 3. С. 3–21; Ч. II. № 4. С. 14–40.
6. Забейхайло М.И. О емкости семейств характеристических функций, обеспечивающих корректное решение диагностических задач// Иск. Инт. и принятие решений. – 2021, №2. – С. 44-53. DOI: 10.14357/20718594210205
7. Zabezhailo M.I. Models of Explanation in Intelligent Data Analysis // IMSC-2021: Integrated Models and Soft Computing in Artificial Intelligence Russian Advances in Fuzzy Systems and Soft Computing: Selected Contributions to the 10th International Conference on “Integrated Models and Soft Computing in Artificial Intelligence (IMSC-2021)”, Kolomna, Russia, May 17-20, 2021 (CEUR Workshop Proc., V.2965 – P.59-63.
8. Popper K.R. The Logic of Scientific Discovery. - N-Y.: Routledge Classics. - 1959.- xxvii+513 P.
9. Финн В.К., Шестерникова О.П. О новом варианте обобщенного ДСМ-метода автоматизированной поддержки научных исследований // Искусственный интеллект и принятие решений. 2016, № 1, с. 57 – 63

ОБ АРГУМЕНТАЦИОННОМ ПОДХОДЕ К АНАЛИЗУ МНЕНИЙ

Михеенкова М.А., д.т.н.

e-mail: m.mikheyenkova@yandex.ru

Финн В.К., д.т.н., профессор

e-mail: ira.finn@gmail.com

ФИЦ ИУ РАН, РГГУ

1. ВВЕДЕНИЕ

Субъективная составляющая социального действия нашла отражение в развитии фундаментальной социологической концепции различных типов социального поведения, в число которых включаются два типа рационального поведения – целерациональное и ценностно-рациональное [1]. Первое может быть упрощённо представлено как выбор наилучших альтернатив действия для максимального приближения к цели с минимизацией возможных издержек. На этой основе базируются модели практической рациональности [2], теории экономической полезности, частично – теории оптимального управления. Использование понятия «интеллектуального агента» в качестве одного из ключевых для Искусственного интеллекта ([3], [4]), совершенствование теории принятия решений, в частности, для описания деятельности таких агентов, привело к конвергенции целерационального и ценностно-рационального поведения. В универсальном смысле рациональное поведение может быть сведено к поиску оптимального соотношения между целью действия и имеющимися знаниями, объективными возможностями и избранными средствами.

Аналогии между социальными сообществами и коллективами искусственных агентов плодотворно используются специалистами как в социальных науках, так и в области искусственного интеллекта [5]. Одной из существенных способностей интеллектуального агента, обеспечивающих эффективную реализацию его действий, является способность принятия наилучшего (с точки зрения тех или иных критериев) решения [6]. Иными словами, предполагается, что в ряду аналогов когнитивных составляющих человеческой активности – интенциональных особенностей, рассуждения, целеполагания – у

интеллектуального агента наличествует и рациональность. Социально обусловленные особенности когнитивных процессов, в частности – рассуждения и принятия решений, – составляют предмет исследований когнитивной социологии. Несмотря на распространённое предубеждение относительно возможности использования методов точных наук при изучении социокультурных явлений [7], ряд исследователей склоняется к интеграции таких методов с принятыми в этой области [8]. ИИ имеет серьёзные традиции предложения идей для когнитивной науки [9], обеспечивая её формальными инструментами исследования различных когнитивных механизмов, в том числе – человеческой рациональности.

Идейным основанием искусственного интеллекта как научной и прикладной области является точная эпистемология (ТЭ) – метатеоретическая дисциплина, формальные средства которой применимы прежде всего к слабо формализованным наукам (наукам о человеке и обществе, наукам о жизни). ТЭ представляет собой исследование взаимодействия познающего субъекта и соответствующего объекта познания [10]. Результатом является формализация и автоматизация (в компьютерных системах) процесса познания как такового, поддержанная конструктивными инструментами порождения нового знания и его принятия – формализованными эвристиками и логиками рассуждения [11]. Практическая реализация в компьютерных интеллектуальных системах этих средств точной эпистемологии обеспечивает конструктивную имитацию и усиление некоторых сторон интеллектуальной деятельности человека. Проекция задач когнитивной социологии на идейный каркас точной эпистемологии означает формализацию собственно познавательной деятельности познающего субъекта – социолога – в его взаимодействии с объектом познания – социумом, представленным множеством индивидуумов.

Научным и конструктивно реализованным аппаратом точной эпистемологии является ДСМ-метод автоматизированной поддержки исследований (ДСМ-метод АПИ) [10, 12]. ДСМ-метод представляет собой методологию создания интеллектуальных систем и является средством формализации, имитации и усиления интеллектуальных процессов, воспроизводя и реализуя в интеллектуальных системах (ИС-ДСМ) эвристики исследователей. Таким образом, при решении задач качественного (неколичественного) анализа социологических данных (с распознаванием эмпирических закономерностей) средствами интеллектуального анализа данных [13, 14], ИС-ДСМ выполняет функции познающего субъекта. Одной из таких задач

является задача анализа и прогнозирования мнений с включением в структуру исследования эффектов социальной когнитивности, в частности – мотивации поведения и его рациональности.

2. АРГУМЕНТАЦИЯ КАК ОСНОВА РАЦИОНАЛЬНОСТИ

Развитие ДСМ-метода автоматизированной поддержки исследований позволило сформировать в рамках метода концепцию эмпирической качественной причинности как эффективную реализацию расширенного понятия рациональности средствами ИС. Образцом рациональности в рамках этого расширенного понятия является исследование, принятие результатов которого опирается на формализованную эвристику*, аргументацию и преобразование неясных идей в точно определённые понятия. Использование средств ДСМ-метода для анализа и прогнозирования мнений не только включает эти три аспекта рациональности, но и позволяет их уточнить и дополнить, а также усилить анализ собственно когнитивных особенностей формирования мнений.

Создание инструментов ИИ, ориентированных на имитацию рациональной феноменологии человеческого интеллекта, требует формализации таких способностей естественного интеллекта, являющихся частью познавательного процесса, как отбор посылок, адекватных цели рассуждения, и порождение достаточных оснований для принятия решений. Развитие логических теорий аргументации [16, 17, 18] приносит несомненную пользу в практических приложениях – использовании аргументации в системах принятия решений, при анализе конфликтов, взаимодействии агентов в многоагентных системах, представлении знаний в интеллектуальных системах. Существенный интерес представляет использование аргументации респондентом при формировании мнений в процессе социологического исследования, формализация которой обеспечивает усиление имитации рациональности мнений. Можно представить, что когнитивный рассуждающий субъект действует (выбирает или не выбирает то или иное мнение) на основании некоторых аргументов (представляющих собой, например, внутренние убеждения или мотивы).

Одним из основных источников получения эмпирического материала в социологии является опрос. Традиционными инструментами обработки результатов опроса являются статистические методы, которые рассматривают обезличенного

* Так, средством рациональности полагал эвристику П. Бернайс [15].

субъекта и не позволяют решать вопросы индивидуальной рациональности и, соответственно, определять степень рациональности индивидов и их групп. Эти вопросы до некоторой степени решаются совершенствованием опросных методологий [19].

Для анализа и прогнозирования мнений (личного восприятия индивидуумом различных сторон рассматриваемого социального явления) на основе результатов социологического опроса и анализ рациональности этих мнений в ДСМ-методе были предложены логические средства для представления мнений, что позволило уточнить формальную структуру закрытого m -значного ($m \geq 3$, m – число возможных вариантов ответа) социологического опроса [20, 21]. Между оценкой для ответа «нет», представленной истинностным значением «ложь», 0, и оценкой ответа «да» – истинностным значением «истина», 1, может располагаться ряд других социологически интерпретируемых вариантов ответа.

В структуре эмпирического опроса аргументированное отношение участников к теме исследования Т может формироваться на основе индивидуальной оценки респондентом утверждений (параметров опроса) из $P = \{p_1, \dots, p_n\}$, раскрывающих её содержание.

В базисном варианте оценка утверждений из P является непосредственной реакцией на ситуацию опроса. Заметим, что полнота представления темы элементами P решается исследователем на уровне формирования содержательной модели. Поскольку такая модель формируется специалистом, носителем экспертного знания, она может рассматриваться в качестве элемента объективного знания.

Рациональность отношения респондента к теме опроса Т может быть усилена, если и само мнение формируется с использованием некоторой аргументации, в частности, аргументационной семантики четырёхзначной логики A_4 [22, 23]. В этом случае мы имеем дело с многоуровневой аргументацией. Это означает возможность учёта влияния когнитивной деятельности на социальное поведение.

Рассмотрим опрос, формализуемый средствами четырёхзначной логики с аргументационной семантикой JA_4 [21]. Элементы P являются пропозициональными переменными, функция оценки $v[p_i] = v \in \{1, -1, 0, \tau\}$, $i = 1, \dots, n$.

Пусть A – множество доводов (аргументов и контраргументов), аргументационная база относительно принятия или непринятия некоторых утверждений. P – множество всех пропозициональных переменных (например, для $P = \{p_1, \dots, p_n\}$ темы Т). Для каждой $p_i \in P$ определим две функции g^+ и g^- , задающие множество аргументов «за» и множество аргументов «против», соответственно:

$$g^+: P \rightarrow 2^A, g^+(p_i) \subseteq A, i = 1, \dots, n.$$

$$g^-: P \rightarrow 2^A, g^-(p_i) \subseteq A, i = 1, \dots, n.$$

Пару функций g^+ , g^- будем называть нормальной, если для всех $p_i \in P$ ($i = 1, \dots, n$) $g^+(p_i) \cap g^-(p_i) = \emptyset$.

Согласие j -го респондента с утверждением p_i (выбор ответа «да») означает, что у него есть аргументы «за» принятие утверждения p_i и нет аргументов «против».

Атомарная оценка для множества истинностных значений $V = \{1, -1, 0, \tau\}$ определяется следующим образом ($i = 1, \dots, n$):

$$v[p_i] = 1 \Leftrightarrow g^+(p_i) \neq \emptyset \text{ \& } g^-(p_i) = \emptyset;$$

$$v[p_i] = -1 \Leftrightarrow g^+(p_i) = \emptyset \text{ \& } g^-(p_i) \neq \emptyset;$$

$$v[p_i] = 0 \Leftrightarrow g^+(p_i) \neq \emptyset \text{ \& } g^-(p_i) \neq \emptyset;$$

$$v[p_i] = \tau \Leftrightarrow g^+(p_i) = \emptyset \text{ \& } g^-(p_i) = \emptyset.$$

Таким образом, в четырёхзначном аргументационном опросе ответы интерпретируются следующим образом: 1 – «да», -1 – «нет» (есть только аргументы «против»), 0 – «и да, и нет», «затрудняюсь ответить» (есть аргументы как «за», так и «против»), τ – «не знаю» (нет никаких аргументов).

Элементы P являются пропозициональными переменными, ответом на вопрос «Какова оценка v утверждения p_i ?» ($i = 1, \dots, n$) является утверждение $J_v p_i$. J_v – введённые Д.А. Бочваром логические связки, которым соответствуют характеристические функции, распознающие истинностные значения v [24]. $J_v p = \begin{cases} t, & \text{если } v[p] = v \\ f, & \text{если } v[p] \neq v \end{cases}$, $v[p]$ – функция оценки, $v \in \{1, -1, 0, \tau\}$, t и f – (внешние) истинностные значения двузначной логики «истина» и «ложь», соответственно. Аргументированное восприятие j -м респондентом темы T – *мнение* – представляется максимальной конъюнкцией логики $J_{A_4} \varphi_j \equiv J_{v_1^{(j)}} p_1 \& \dots \& J_{v_n^{(j)}} p_n$ (\equiv – графическое равенство), где $v_i^{(j)}$ – соответствующие оценки утверждений p_i ($i = 1, \dots, n$), $v_i^{(j)} \in \{1, -1, 0, \tau\}$, $j = 1, \dots, 4^n$. Эта конъюнкция определяется по аналогии с максимальной конъюнкцией двузначной логики, т.е. для каждой p_k ($k=1, \dots, n$) $J_{v_k^{(j)}} p_k$ входит в C_j , причём без повторений, а $J_{v_k^{(j)}} p_k$ и $J_{v_i^{(j)}} p_k$, где $v_k^{(j)} \neq v_i^{(j)}$, одновременно в C_j не входят. Множество всех возможных ответов по теме T с каркасом P обозначим K , $|K|=4^n$, $K = \{\varphi_j \mid \varphi_j \equiv J_{v_1^{(j)}} p_1 \& \dots \& J_{v_n^{(j)}} p_n, v^{(j)}[p_i] = v_i^{(j)}, v_i^{(j)} \in \{1, -1, 0, \tau\}, i = 1, \dots, n, j = 1, \dots, 4^n\}$. В реальном опросе реализуется множество ответов $K' \subseteq K$. Если при расширении множества опрашиваемых K' не изменяется,

можно говорить об экспериментально полученной репрезентативной (в логическом, а не статистическом смысле) выборке респондентов.

Можно оценить, насколько рациональным является выбор j -м респондентом аргументационных функций g_j^σ ($\sigma \in \{+, -\}$) для элементов $P = \{p_1, \dots, p_n\}$ и, соответственно, формирования собственного мнения. Множество P формируется исследователем на основании содержательной экспертной модели и может содержать семантические связи между утверждениями p_1, \dots, p_n . Представим эти связи в виде непротиворечивого множества формул $\Sigma = \{\psi_1, \dots, \psi_s\}$ соответствующей логики JA_4 , выражающих логические зависимости между p_1, \dots, p_n , причём конъюнкция формул $\bar{\psi} = \psi_1 \& \dots \& \psi_s$ не является тавтологией. Представление множества Σ логическими средствами является элементом формализованной эвристики, однако вопрос полноты описания семантических связей между p_1, \dots, p_n в Σ , так же, как и полноты представления ими темы T , не может быть решён в рамках формального аппарата.

Непротиворечивость множества Σ может быть установлена методом аналитических таблиц для логики JA_4 [21]. Здесь используются одноместные логические связки I_1, I_{-1}, I_0, I_τ и двуместные связки $\&, \vee, \rightarrow$. Соответственно, $v[I_1 p_i] = t \Leftrightarrow g^+(p_i) \neq \emptyset$ & $g^-(p_i) = \emptyset$, и т.п. Контрарными парами являются $t\phi, f\phi$ (для помеченных формул) и $I_\nu p, I_\mu p$, где $\nu \neq \mu$ (для непомеченных формул), $\nu, \mu \in \{1, -1, 0, \tau\}$. Формулируются соответствующие α -правила (правила конъюнктивного типа), β -правила (правила дизъюнктивного типа) и χ -правила (вследствие применения которых дерево становится традическим).

Аналитической таблицей \mathcal{T}_Σ для множества формул $\Sigma = \{\psi_1, \dots, \psi_s\}$ является аналитическая таблица такая, что её началом является последовательность формул ψ_1, \dots, ψ_s . Легко показать, что \mathcal{T}_Σ равносильна по замкнутости аналитической таблице $\mathcal{T}_{\bar{\psi}}$ с корнем $\bar{\psi}$. Для непротиворечивого множества Σ аналитическая таблица \mathcal{T}_Σ ($\mathcal{T}_{\bar{\psi}}$) не замкнута. Задание множества Σ отвечает принципу рационализации представления темы с учётом объективного знания и позволяет оценить рациональность восприятия темы респондентом.

3. ЛОГИЧЕСКАЯ ИНТЕРПРЕТАЦИЯ АРГУМЕНТАЦИИ В ОПРОСЕ

Мнение рационально мыслящего субъекта не должно входить в противоречие с представленными в Σ зависимостями между утверждениями p_1, \dots, p_n . Пусть $consis(\Sigma)$ – метапредикат

непротиворечивости множества формул (Σ). Рассмотрим полученное в результате реального опроса множество ответов $K' = \{C_1, \dots, C_k\}$, $K' \subseteq K$, $k \leq 4^n$, C_j – мнение j -го респондента, $j = 1, \dots, k$. Если аналитическая таблица $\mathcal{T}_{\Sigma \cup \{C_j\}}$ не замкнута, это значит, что мнение C_j не противоречит Σ – имеет место $\text{consis}(\Sigma \cup \{C_j\})$, – а самого j -го респондента можно считать рациональным. Можно оценить также степень непротиворечивости всего опроса: $\delta(K^*, K') = \frac{|K^*|}{|K'|}$ (см. [20]). Здесь K^* – множество мнений, не противоречащих Σ , $K^* = \{X \mid \text{consis}(\Sigma \cup \{X\}) \& (X \in K')\}$, $K^* \subseteq K'$.

При создании формальных инструментов для представления и анализа когнитивных процессов не следует забывать об адекватном отображении синтаксическими средствами семантических особенностей. Соответственно, от выбора логической семантики зависят правила построения аналитической таблицы для выявления мнений, находящихся в противоречии с элементами множества Σ .

Зададим в рамках аргументационного подхода оценки различных комбинаций аргументов, отличные от приведённых выше, сохраняя возможность их содержательной интерпретации. Пусть множество истинностных значений есть $V = \{0, \frac{1}{3}, \frac{2}{3}, 1\}$. Определим атомарную оценку следующим образом ($i = 1, \dots, n$):

$$v[p_i] = 1 \Leftrightarrow g+(p_i) \neq \emptyset \& g^-(p_i) = \emptyset;$$

$$v[p_i] = 0 \Leftrightarrow g+(p_i) = \emptyset \& g^-(p_i) \neq \emptyset;$$

$$v[p_i] = \frac{2}{3} \Leftrightarrow g+(p_i) \neq \emptyset \& g^-(p_i) \neq \emptyset;$$

$$v[p_i] = \frac{1}{3} \Leftrightarrow g+(p_i) = \emptyset \& g^-(p_i) = \emptyset.$$

Здесь 1 интерпретируется как ответ «да», 0 – «нет», ответу «затрудняюсь ответить» соответствует $\frac{2}{3}$, «не знаю» – $\frac{1}{3}$.

В [25] рассмотрена возможная семантика отличного от 0 («ложь») и 1 («истина») истинностного значения $\frac{1}{2}$ для так называемых «логик значения»: сильная бессмысленность (математическая) в логике Бочвара B_3 , слабая бессмысленность (лингвистическая) в логике Эббингхауза E_3 , неопределённость (неизвестно, истина или ложь) в предложенном В.К. Финном варианте логики Лукасевича L'_3 [26]. Такая семантика может быть распространена на истинностные значения $\frac{1}{3}$ и $\frac{2}{3}$, возникшие в результате использования аргументационного подхода, для логик B_4 , E_4 и L'_4 , соответственно. Логические связки – конъюнкция, дизъюнкция, импликация – определяются для каждой логики соответствующими истинностными

таблицами. Отрицание \sim определяется одинаково для всех трёх логик: для 0 и 1 – как в двузначной логике, $\sim \frac{1}{3} = \frac{1}{3}$, $\sim \frac{2}{3} = \frac{2}{3}$, определение формулы стандартно. Полное описание соответствующих логик находится за рамками нашего изложения. Рассмотрим лишь особенности вычисления определённой выше рациональности мнений в каждой из этих логик.

3.1. ЛОГИКА V_4

В этой логике сохраняется интерпретация оценок высказываний, отличных от 0 и 1, как (сильно) бессмысленных [24], т.е. имеющих дефекты смысла: если атомарное высказывание p бессмысленно, то сложное высказывание $\varphi(p)$, содержащее вхождение p , также бессмысленно. При этом оценка $\frac{1}{3}$ означает максимальную сильную бессмысленность, $\frac{2}{3}$ – сильную бессмысленность меньшей степени. В этом случае сложное высказывание $\varphi(p_1, \dots, p_n)$ бессмысленно с оценкой $\frac{1}{3}$, если хотя бы для одного p_i ($i = 1, \dots, n$) $v[p_i] = \frac{1}{3}$. Если хотя бы для одного p_i ($i = 1, \dots, n$) $v[p_i] = \frac{2}{3}$ и при этом ни для одного p_j ($j = 1, \dots, n, j \neq i$) $v[p_j] \neq \frac{1}{3}$, $\varphi(p_1, \dots, p_n)$ бессмысленно с оценкой $\frac{2}{3}$.

Логические связки \cap , \cup , \rightarrow – конъюнкция, дизъюнкция и импликация – определяются соответствующими истинностными таблицами, отрицание \sim определено выше, определение формулы стандартно, логическим связкам J_v соответствуют характеристические функции, распознающие истинностные значения $v \in \{0, \frac{1}{3}, \frac{2}{3}, 1\}$. Как и в

V_3 , логические связки \sim , \cap и \cup являются внутренними связками с областью значений $\{0, \frac{1}{3}, \frac{2}{3}, 1\}$; импликация \rightarrow является внешней связкой с областью значений $\{0, 1\}$. Внутренние формулы (включающие переменные, которые не находятся в сфере действия внешней связки) используются для представления фактов, внешние – для представления и доказательства утверждений о фактах. Для внешних формул существуют J -ДНФ [26] – дизъюнкция конъюнкций атомов $J_v p$, соответственно, может быть применён метод аналитических таблиц, где используются так называемые помеченные формулы $J_v \varphi$, φ – непомеченная формула. Контрными парами для помеченных формул являются $J_v \varphi$, $J_{\neq v} \varphi$, где $v \neq \mu$, $v[(J_v \varphi \cap J_{\neq v} \varphi)] = 0$ для

любой формулы φ и ν , $\mu \in \{0, \frac{1}{3}, \frac{2}{3}, 1\}$. Формулируются соответствующие правила, которые мы не будем здесь приводить, а перейдём сразу к вычислению рациональности (непротиворечивости) мнения на простом примере.

Ответ j -го респондента по теме T , характеризующейся утверждениями $P = \{p_1, \dots, p_n\}$, представляется максимальной конъюнкцией логики B_4 $C_j \equiv J_{\nu_1^{(j)}} p_1 \cap \dots \cap J_{\nu_n^{(j)}} p_n$. ($\nu^{(j)}[p_i] = \nu_i^{(j)}$, $\nu_i^{(j)} \in \{0, \frac{1}{3}, \frac{2}{3}, 1\}$, $i = 1, \dots, n$).

Пусть некоторый опрос по теме T характеризуется утверждениями $P = \{p, q, r, s\}$, при этом $\Sigma = \{(p \rightarrow (q \cup (r \cap s)))\}$. Пусть в результате опроса $C \in K'$ – один из полученных ответов, $C \equiv J_1 p \cap J_1 q \cap J_{\frac{1}{3}} r \cap J_{\frac{2}{3}} s$. Чтобы проверить, не противоречит ли $C \Sigma$, т.е. имеет ли место $consis(\Sigma \cup \{C\})$, построим аналитическую таблицу $\mathcal{T}_{\Sigma \cup \{C\}}$.

Для этого воспользуемся соответствующими α -правилом $\frac{J_1(\varphi \cap \psi)}{J_1 \varphi, J_1 \psi}$, ξ -правилом $\frac{J_1(\varphi \rightarrow \psi)}{J_0 \varphi \mid J_{\frac{1}{3}} \varphi \mid J_{\frac{2}{3}} \varphi \mid J_1 \psi}$ и χ -правилами $\frac{J_1(\varphi \cup \psi)}{J_1 \varphi, J_0 \psi \mid J_1 \varphi, J_1 \psi \mid J_0 \varphi, J_1 \psi}$ и $\frac{J_0(\varphi \cap \psi)}{J_0 \varphi, J_1 \psi \mid J_0 \varphi, J_0 \psi \mid J_1 \varphi, J_0 \psi}$ логики B_4 . Здесь φ, ψ – формулы логики B_4 .

$$\begin{array}{c}
 J_1(p \rightarrow (q \cup (r \cap s))) \\
 J_1 p \\
 J_1 q \\
 J_{\frac{1}{3}} r \\
 J_{\frac{2}{3}} s
 \end{array}$$

| | | | | | | | |
|---------|---------------------|---------------------|--------------------------|-----------------|-----------------|--|--|
| $J_0 p$ | $J_{\frac{1}{3}} p$ | $J_{\frac{2}{3}} p$ | $J_1(q \cup (r \cap s))$ | | | | |
| * | * | * | | | | | |
| | | | $J_1 q$ | $J_1 q$ | $J_0 q$ | | |
| | | | $J_0(r \cap s)$ | $J_1(r \cap s)$ | $J_1(r \cap s)$ | | |
| | | | | $J_1 r$ | s | | |
| | | | $J_0 r$ | $J_1 r$ | $J_1 r$ | | |
| | | | $J_1 s$ | $J_0 s$ | $J_1 s$ | | |
| | | | * | * | * | | |
| | | | | | | | |
| | | | | | | | |

Рис.1. Пример построения аналитической таблицы для B_4

Таблица замкнута, нет открытых ветвей, следовательно, $\neg\text{consis}(\Sigma \cup \{C\}) - C$ противоречит Σ , $C \notin K^*$.

3.2. ЛОГИКА E_4

Проверим, является ли противоречивым приведённое выше мнение в логике E_4 , являющейся расширением логики V_4 . Бессмысленность здесь, в отличие от V_3 и V_4 , является слабой – высказывания рассматриваются как могущие терять смысл. При этом $\frac{1}{3}$ – слабая бессмысленность, $\frac{2}{3}$ – меньшая степень слабой бессмысленности. Как и в E_3 , сложное высказывание $\varphi(p_1, \dots, p_n)$ бессмысленно тогда и только тогда, когда все атомарные высказывания p_1, \dots, p_n , входящие в φ , являются бессмысленными, однако общая оценка – $\frac{1}{3}$ или $\frac{2}{3}$ – определяется в зависимости от входящих в $\varphi(p_1, \dots, p_n)$ логических связок \cap, \vee, \rightarrow – конъюнкции, дизъюнкции и импликации, определяемых соответствующими истинностными таблицами, и определённого выше отрицания \sim . Определение формулы стандартно.

Рассмотрим описанный выше опрос по теме T , характеризующийся утверждениями $P = \{p, q, r, s\}$, при этом $\Sigma = \{(p \rightarrow (q \vee (r \cap s)))\}$. Рассмотрим тот же ответ $C \in K'$, $C \equiv J_1 p \cap J_1 q \cap J_{\frac{1}{3}} r \cap J_{\frac{2}{3}} s$ и построим аналитическую таблицу $\mathcal{T}_{\Sigma \cup \{C\}}$.

$$\begin{array}{c}
 J_1(p \rightarrow (q \vee (r \cap s))) \\
 J_1 p \\
 J_1 q \\
 J_{\frac{1}{3}} r \\
 J_{\frac{2}{3}} s
 \end{array}$$

| | | | |
|---------|---------------------|---------------------|--------------------------|
| $J_0 p$ | $J_{\frac{1}{3}} p$ | $J_{\frac{2}{3}} p$ | $J_1(q \vee (r \cap s))$ |
| * | * | * | $J_1 q$ |
| | | | $J_1(r \cap s)$ |
| | | | $J_1 r$ |
| | | | $J_1 s$ |
| | | | * |

Рис.2. Пример построения аналитической таблицы для E_4

При построении таблицы использовались соответствующие α -

правило $\frac{J_1(\varphi \wedge \psi)}{J_1\varphi, J_1\psi}$, ξ -правило $\frac{J_1(\varphi \rightarrow \psi)}{J_0\varphi \mid \frac{J_1\varphi}{3} \mid \frac{J_2\varphi \mid J_1\psi}{3}}$, которое совпадает с соответствующим правилом логики B_4 , и β -правило $\frac{J_1(\varphi \vee \psi)}{J_1\varphi \mid J_1\psi}$ логики E_4 .

Здесь φ, ψ – формулы логики E_4 .

Таблица не замкнута, есть открытая ветвь, т.е. имеет место $consis(\Sigma \cup \{C\})$ – C не противоречит Σ , $C \in K^*$. Следовательно, вычисленная по формуле $\delta(K^*, K') = \frac{|K^*|}{|K'|}$ степень непротиворечивости опроса будет отличной от степени непротиворечивости, вычисленной средствами B_4 , поскольку различными будут множества K^* .

3.3. ЛОГИКА L'_4

В отличие от B_4 и E_4 – логик бессмысленностного типа [25], – логика L'_4 относится к логикам неопределённостного типа, где оценки $\frac{1}{3}$ и $\frac{2}{3}$ появляются не вследствие смысловых дефектов, а как результат неполноты информации. Иными словами, $\frac{1}{3}$ здесь – неопределённость, $\frac{2}{3}$ – меньшая степень неопределённости: высказывание p либо истинно, либо ложно, но его оценка неизвестна (с разной степенью недостаточности информации). Логические связи $\&$, \vee , \rightarrow – конъюнкция, дизъюнкция и импликация – определяются соответствующими истинностными таблицами. При этом учитывается, что L'_4 представляет собой расширение логики E_4 такое, что оно сохраняет на множестве истинностных значений $\{0, 1\}$ логические связи двузначной логики.

Для описанного выше опроса по теме T с $P = \{p, q, r, s\}$ Σ представляется в виде $\Sigma = \{(p \rightarrow (q \vee (r \& s)))\}$, ответ $C \equiv J_1p \& J_1q \& \frac{J_1r}{3} \& \frac{J_2s}{3}$, $C \in K'$. Воспользовавшись соответствующими α -

правилом $\frac{J_1(\varphi \& \psi)}{J_1\varphi, J_1\psi}$, β -правилом $\frac{J_1(\varphi \vee \psi)}{J_1\varphi \mid J_1\psi}$ и ξ -правилом $\frac{J_1(\varphi \rightarrow \psi)}{J_0\varphi \mid \frac{J_1\varphi}{3} \mid \frac{J_2\varphi \mid J_1\psi}{3}}$

логики L'_4 (φ, ψ – формулы логики L'_4) для построения аналитической таблицы $\mathcal{T}_{\Sigma \cup \{C\}}$, убедимся, что, как и для E_4 , таблица имеет открытые ветви, а значит имеет место $consis(\Sigma \cup \{C\})$ и $C \in K^*$.

В приведённых примерах представлен один и тот же опрос по теме T с одним и тем же множеством $P = \{p_1, \dots, p_n\}$, характеризующим содержание темы. Представленные в Σ связи между утверждениями p_1, \dots, p_n семантически эквивалентны, однако их формальное представление связано с определением логических связей в соответствующей логике. Отсюда видно, насколько существенным

является выбор средств формализации для адекватного отражения семантики предметной области, в нашем случае – от оценок опроса и их содержательной и аргументационной интерпретации.

4. ЗАКЛЮЧЕНИЕ

Представленная аргументационная схема вписывается в предложенную в [23] схему многоуровневых аргументационных систем, когда аргументы некоторых высказываний могут, в свою очередь, иметь аргументы, что создает условия для отображения рефлексии. Описанное выше аргументированное отношение к теме опроса T посредством множества $P = \{p_1, \dots, p_n\}$ может быть детализировано следующим образом: тема $\{T\}$ является одноэлементным множеством аргументируемых высказываний P^a , элементы $\{p_1, \dots, p_n\}$ являются аргументирующими (базисными) высказываниями P' . Если же оценка элементов $\{p_1, \dots, p_n\}$ формируется на основании аргументационных функций, аргументационная база A оказывается нижним уровнем аргументации (аргументирующими высказываниями $P' = A$), а элементы множества $\{p_1, \dots, p_n\}$ переходят в ранг аргументируемых и аргументирующих высказываний, $P^b = \{p_1, \dots, p_n\}$. Таким образом, мы получаем усиленный инструмент для анализа рациональной когнитивной составляющей такого вида социального поведения, как мнение (в том числе, общественное).

Предложенный аргументационный подход дополняет расширенное понятие рациональности, явившееся результатом развития концепции эмпирической качественной причинности в ДСМ-методе автоматизированной поддержки исследований. Это достигается логическим представлением мнений, достижением стабилизированного множества ответов, формированием представления данных на основании постулатов поведения и ситуационизма (см. [12]) – включением в структуру исследования источников мотивации, выбором оценок на основании аргументации и анализом непротиворечивости мнений с учетом объективного знания (Σ).

Помимо этого, полученные результаты подтверждают необходимость тщательной разработки языка представления данных и выбора адекватных формальных средств при решении исследовательских задач в плохо формализованных областях. Здесь незаменимым является междисциплинарное взаимодействие представителей точных наук и специалистов предметной области – в нашем случае, социальных наук [27]. Для обеспечения плодотворной

исследовательской деятельности необходима предварительная содержательная работа эксперта, взаимная адаптация представлений о предмете и последующая эмпирическая верификация формальных инструментов применительно к конкретным задачам.

Литература

1. Вебер М. Основные социологические понятия / В кн.: Вебер М. Избранное: протестантская этика и дух капитализма – М.: РОССПЭН, 2006. С. 453–482.
2. Девятко И.Ф. Социологические теории деятельности и практической рациональности. – М.: АВАНТИ ПЛЮС, 2003.
3. Рассел С., Норвиг П. Искусственный интеллект. Современный подход. Изд.2-е. – М.: Вильямс, 2018.
4. Wooldridge M.J. A Brief History of Artificial Intelligence. – New York: Flatiron Books, 2021.
5. Wooldridge M.J. An Introduction to Multiagent Systems (2nd ed.). – Chichester, United Kingdom: John Wiley&Sons, 2009.
6. Тарасов В.Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. – М.: УРСС, 2002.
7. Zerubavel E. Social mindscape. An Invitation to cognitive sociology. – L.: Harvard University Press, 1997.
8. Ignatow G. Ontology and Method in Cognitive Sociology // Sociological Forum. 2014. Vol. 29. No. 4, pp. 990–994. DOI: 10.1111/socf.12131
9. Boden M. A. AI. Its nature and future. – Oxford: Oxford University Press, 2016.
10. Финн В.К. Эпистемологические основания искусственного интеллекта / В кн.: Финн В.К. Интеллект, информационное общество, гуманитарное знание и образование. – М.: ЛЕНАНД, 2021. С. 27–98.
11. McCarthy, J. Epistemological problems of artificial intelligence / In: Webber, B.L., Nilsson, N.J. (eds.) Readings in Artificial Intelligence. – Morgan Kaufmann, 1981. Pp. 459–465. <https://doi.org/10.1016/B978-0-934613-03-3.50035-0>
12. Автоматическое порождение гипотез в интеллектуальных системах. Под общ. ред. В.К. Финна. – М.: Книжный дом «ЛИБРОКОМ», 2020 (изд. 2-е стереотип.).
13. Михеенкова М.А., Климова С.Г. Интеллектуальный анализ данных в социологических исследованиях // НТИ, сер. 2. 2018. № 12. С. 12–24.
14. Mikheyenkova M.A. Cognitive Knowledge Discovery in Social Sciences // Communications in Computer and Information Science. 2020. Vol 1282, pp. 48–63. DOI 10.1007/978-3-030-60447-9_4
15. Бернайс П. О рациональности / В кн.: Эволюционная эпистемология и логика социальных наук. Карл Поппер и его критики. – М.: Эдиториал УРСС, 2000. С. 154–162.
16. Bench-Capon T.J.M., Dunne Paul E. Argumentation in Artificial Intelligence // Artificial Intelligence. 2007. Vol.171, pp .619–641.

17. Rahwan I., and Simari G.R. (Eds.) *Argumentation in artificial intelligence.* – Dordrecht, Heidelberg, London, New York: Springer, 2009.
18. Вагин В.Н. Знания и убеждения в интеллектуальном анализе данных. – М.: Физматлит, 2019.
19. Schwarz N. Cognitive aspects of survey methodology // *Applied Cognitive Psychology.* 2007. Vol. 21. Issue 2, pp. 277–287.
20. Finn V.K., Mikheyenkova M.A. Plausible Reasoning for the Problems of Cognitive Sociology // *Logic and Logical Philosophy.* 2011. Vol. 20, pp. 113–139.
21. Финн В.К., Михеенкова М.А. О логических средствах концептуализации анализа мнений / В сб.: *Многозначные логики и их применения*, т. 2: *Логики в системах искусственного интеллекта* (под ред. В.К. Финна) – М.: URSS, 2020. С. 152-199.
22. Финн В.К. Стандартные и нестандартные логики аргументации // Там же. С. 59–91.
23. Финн В.К. Двенадцать тезисов об аргументационных системах // *НТИ*, сер. 2. 2012. № 11. С. 20–32.
24. Бочвар Д.А. Об одном трёхзначном исчислении и его применении к анализу парадоксов классического расширенного функционального исчисления / В сб.: *Многозначные логики и их применения*, т. 1: *Логические исчисления, алгебры и функциональные свойства* (под ред. В.К. Финна) – М.: URSS, 2020. С. 25-46.
25. Финн В.К., Аншаков О.М., Григолия Р.Ш., Забежайло М.И. Многозначные логики как фрагменты формализованной семантики / Там же. С. 323–356.
26. Финн В.К. Аксиоматизация некоторых трёхзначных исчислений высказываний и их алгебр / Там же. С. 64–104.
27. Климов И.А., Климова С.Г., Михеенкова М.А. Формирование зоны общего языка в междисциплинарном взаимодействии: рефлексии по поводу опыта сотрудничества представителей точных и социальных наук // *Социологический журнал.* 2019. Том 25. № 3. С. 117–132. DOI: 10.19181/socjour.2019.25.3.6679

СЕКЦИЯ 5

**КОГНИТИВНЫЕ МОДЕЛИ
В ИСКУССТВЕННОМ
ИНТЕЛЛЕКТЕ**

УДК 004.891.3

**ИНТЕГРАЦИЯ СРЕДСТВ КОГНИТИВНОЙ
ГРАФИКИ И НЕЧЕТКОЙ ЛОГИКИ ДЛЯ
ПРЕДСТАВЛЕНИЯ И ДИАГНОСТИРОВАНИЯ
СЛОЖНЫХ ПАТОЛОГИЙ ЗРЕНИЯ***

*Еремеев А.П., д.т.н., профессор,
НИУ «МЭИ»*

e-mail: eremeev@appmat.ru

*Цапенко И.В., к.б.н., главный специалист
НМИЦ глазных болезней им. Г. Гельмгольца*

e-mail: sunvision@mail.ru

1. ВВЕДЕНИЕ

Анализ и диагностика проблемных ситуаций в сложных системах и объектах различного типа (технических, транспортных, организационных, биологических и других), особенно в режиме реального времени, а также при наличии различного вида так называемых НЕ-факторов (или «шума») в поступающей из различных источников информации требует как формирования соответствующей базы данных, так и наличие средств для оперативного и образного представления информации с применением средств когнитивной графики [1,2].

* Работа выполнена при финансовой поддержке РФФИ, проекты №20-07-00498 А, №20-57-00015 Бел-а.

Специалистами НИУ «МЭИ» (кафедры Прикладной математики и искусственного интеллекта и Управляющих и интеллектуальных технологий) и НМИЦ глазных болезней им. Г. Гельмгольца (отдел Клинической физиологии зрения им. С.В. Кравкова) выполняются исследования и разработки методов и программных средства для создания прототипа экспертной системы поддержки принятия решений для помощи врачам при анализе и диагностике сложных патологий зрения на примере заболеваний сетчатки. При разработке прототипа используется интеграция различных методов и подходов системного анализа и искусственного интеллекта: амплитудно-частотный анализ, вейвлет преобразования, нейросетевые технологии и мягкие вычисления, байесовские сети доверия и теория свидетельств Демпстера-Шефера, образное представление патологии на основе когнитивной графики (когнитивных образов) и т.д. [3,4].

В данной работе рассматриваются возможности интеграции методов и средств когнитивной графики (когнитивных образов) и нечеткой логики для образного представления сложных патологий зрения (патологий сетчатки глаза) для помощи офтальмологам в анализе и диагностике глазных болезней с применением подхода на основе электроретинографии (электроретинограмм, ЭРГ), заключающейся в исследовании биопотенциалов сетчатки глаза при определенном световом стимулировании [3-7].

2. ОБРАЗНОЕ ПРЕДСТАВЛЕНИЕ ПАТОЛОГИЙ СЕТЧАТКИ НА ОСНОВЕ ДАННЫХ ЭРГ

Для получения информации о функциональном состоянии различных клеточных элементов сетчатки офтальмологами используются электроретинографические тесты – ЭРГ. ЭРГ представляет собой графическое выражение биоэлектрической активности клеточных элементов сетчатки в ответ на световое раздражение (рис. 1). Каждый из компонентов ЭРГ генерируется различными структурами сетчатки. На рис. 1: a_1 и a_2 – амплитуда a -волны; b_1 и b_2 – амплитуда b -волны; D – длительность b -волны; L – латентный период; T_b – время кульминации. По оси ординат указаны амплитуда волн ЭРГ, по оси абсцисс – длительность волн ЭРГ.

В ЭРГ можно выделить 3 основных компонента: начальная a -волна, b -волна и поздняя c -волна. Негативная a -волна отражает функцию фоторецепторов как начальную часть позднего рецепторного потенциала. Эта волна имеет двойное происхождение соответственно двум видам фоторецепторов: более ранняя a_1 -волна связана с активностью фотопической системы сетчатки, a_2 -волна — со

скотопической системой. Далее a -волна переходит в позитивную b -волну, отражающую электрическую активность биполяров и клеток Мюллера с возможным вкладом горизонтальных и амакриновых клеток. Волна b (оп-эффект) в стандартных условиях регистрации ЭРГ отражает биоэлектрическую активность в зависимости от условий адаптации и функции фотопической и скотопической системы сетчатки, которые представлены в позитивном компоненте волнами b_1 и b_2 . Волны c и d представляют затухающие процессы и менее информативны для диагностики.

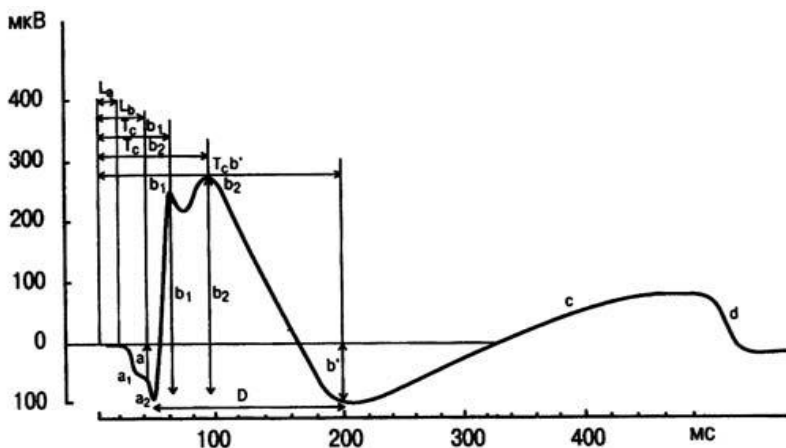


Рис. 1. Схематическое изображение максимальной ЭРГ

ЭРГ отражает электрическую активность большинства клеточных элементов сетчатки и зависит от количества здоровых функционирующих нейронов, а оценка различных компонентов ЭРГ позволяет проводить раннюю и дифференциальную диагностику заболеваний сетчатки.

К сожалению, интерпретировать графики ЭРГ способны только эксперты – специалисты в области физиологии зрения, для практикующих врачей-офтальмологов эта информация сложна в восприятии. Представление патологий сетчатки в виде когнитивных образов (КО) позволяет решить эту проблему [6].

Формально модель когнитивного образа (КО) определяется тройкой: $K = (X, R, F)$, где $X = \{ C_i \mid i=1, 2, \dots, n \}$ – непустое множество концептов (объектов предметной области), $R = \{ R_j \mid j=1, 2, \dots, m \}$ – семейство отношений на множестве X , F – множество функций интерпретации.

Каждый концепт C_i определяется следующим образом: $C_i = (N_i, T_i, P_i, Ch_i, A_i)$, где N_i – имя концепта C_i , $P_i = \{ C_i^k \mid k=1, 2, \dots, x \}$ – множество предков концепта C_i , $Ch_i = \{ C_i^s \mid s=1, 2, \dots, y \}$ – множество потомков концепта C_i , $A_i = \{ A_i^u \mid u=1, 2, \dots, q \}$ – список атрибутов концепта C_i .

Каждый атрибут A_i^u концепта C_i определяется следующим образом: $A_i^u = (Na_i^u, Ta_i^u, Va_i^u)$, где Na_i^u – имя атрибута A_i^u , Ta_i^u – тип атрибута A_i^u , Va_i^u – значение атрибута A_i^u . T_i – тип концепта C_i определяется следующим образом: $T_i = (Nq_i, Aq_i, Mq_i, Pq_i)$, где Nq_i – имя типа концепта C_i ; $Aq_i = \{ Aq_i^h \mid h=1, 2, \dots, r \}$ – список атрибутов типа концепта C_i ; $Mq_i = \{ Mq_i^d \mid d=1, 2, \dots, f \}$ – список действий (методов) типа концепта C_i ; Pq_i – имя родительского типа концепта C_i .

Тогда алфавитом в модели КО сетчатки является набор (множество) геометрических примитивов (точка, сплайн, эллипс, прямоугольник) - $Aq_i = \{ Aq_i^h \mid h=1, 2, \dots, r \}$. Формулой – один элемент слоя, состоящий из последовательности символов алфавита.

Аксиомы описывают построения элементов (элементарных КО) каждого вида:

1) образ «Палочка» состоит из двух эллипсов и находится в первом слое;

2) образ «Колбочка» состоит из трех эллипсов и находится в первом слое;

3) образ «Биполяр» состоит из прямоугольника и эллипса, занимающего 0.3 части от всего элемента, и находится во втором слое;

4) образ «Ганглиозная клетка» состоит из прямоугольника и эллипса, занимающего 0.8 части от всего элемента, и находится в третьем слое;

5) образ «Горизонтальная клетка» состоит из эллипса и сплайна и находится во втором слое;

6) образ «Амакриновая клетка» состоит из эллипса и находится во втором слое;

7) образ «Клетка Мюллера» состоит из прямоугольника и множества эллипсов двух видов и находится сбоку от всех других слоев.

Правила вывода определяют связи между элементами различных слоев:

1) образы «Палочка» и «Колбочка» могут быть связаны с элементами «Биполяр» и «Горизонтальная клетка»;

2) образы «Биполяр» и «Амакриновая клетка» могут быть связаны с элементом «Ганглиозная клетка»;

3) образ «Ганглиозная клетка» имеет связь, называемую «Аксон».

Множество концептов задаются следующим образом:

$C_1 = (\text{Палочка}, 0, \text{Биполяр(палочковый)}, A_1);$

$C_2 = (\text{Колбочка}, 0, \text{Биполяр(колбочковый)}, A_2);$

$C_3 = (\text{Биполяр(палочковый)}, \text{Палочка}, \text{Ганглиозная клетка}, A_3);$

$C_4 = (\text{Биполяр(колбочковый)}, \text{Колбочка}, \text{Ганглиозная клетка}, A_4);$

$C_5 = (\text{Ганглиозная клетка}, \text{Биполяр(палочковый)}, \text{Аксон}, A_5);$

$C_6 = (\text{Ганглиозная клетка}, \text{Биполяр(колбочковый)}, \text{Аксон}, A_6);$

$C_7 = (\text{Аксон}, \text{Ганглиозная клетка}, 0, A_7);$

$C_8 = (\text{Горизонтальная клетка}, \text{Палочка}, \text{Биполяр(палочковый)},$

$A_8);$

$C_9 = (\text{Горизонтальная клетка}, \text{Колбочка}, \text{Биполяр(колбочковый)},$

$A_9);$

$C_{10} = (\text{Амакриновая клетка}, \text{Биполяр}, \text{Горизонтальная клетка},$

$A_{10}).$

Для каждого концепта определяются список из трех атрибутов

$A_i = \{ A_i^u \mid u=1,2,3 \};$

$A_i^1 = (\text{Цвет}, 3 \text{ числа}, 0-255);$

$A_i^2 = (\text{Ширина}, \text{число}, W_i);$

$A_i^3 = (\text{Высота}, \text{число}, H_i).$

Аналогично вводится множество наборов показателей $P = \{ P_i^h \mid h=1, 2, \dots, r \};$

$P_i^h = (\text{Вид исследования}, \text{показатель}, \text{значение});$

$P_i^1 = (\text{Максимальная ЭРГ}, A_a, 70-220) -$ показывает изменение в фоторецепторах(палочки и колбочки);

$P_i^2 = (\text{Максимальная ЭРГ}, T_a, 10-30) -$ показывает изменение в связях между фоторецепторами и их биполярами;

$P_i^3 = (\text{Максимальная ЭРГ}, A_b, 125-380) -$ показывает изменение в палочках и их биполярных клетках, также в клетке Мюллера;

$P_i^4 = (\text{Максимальная ЭРГ}, T_b, 18-56) -$ показывает изменение в связях между палочками и их биполярами;

$P_i^5 = (\text{Скотопическая ЭРГ}, A_b, 50-150) -$ показывает изменение в палочковых биполярных клетках;

$P_i^6 = (\text{Скотопическая ЭРГ}, T_b, 30-90) -$ показывает изменение в связях между палочками и их биполярами;

$P_i^7 = (\text{Фотопическая ЭРГ}, A_a, 14-44) -$ показывает изменение в колбочках;

$P_i^8 = (\text{Фотопическая ЭРГ}, T_a, 7-21) -$ показывает изменение в связях между колбочками и их биполярами;

P_i^9 = (Фотопическая ЭРГ, Ab , 50-150) показывает изменение в колбочковых биполярах и клетке Мюллера;

P_i^{10} = (Фотопическая ЭРГ, Tb , 14-42) – показывает изменение в связях между колбочками и их биполярами;

P_i^{11} = (ФЭРГ 30 Гц, *Апик-тупик*, 30-100) – показывает изменение в колбочковых биполярах без Мюллера;

P_i^{12} = (Осциляторные потенциалы, $O1$, 40-70) – показывает изменение в Амакринах и обратных функциях связи от ганглиозных клеток к Амакринам;

P_i^{13} = (Осциляторные потенциалы, $O2$, 40-70) (основной показатель) -показывает изменение в Амакринах и обратных функциях связи от ганглиозных клеток к Амакринам;

P_i^{14} = (Осциляторные потенциалы, $O3$, 40-70) - показывает изменение в Амакринах и обратных функциях связи от ганглиозных клеток к Амакринам;

P_i^{15} = (Осциляторные потенциалы, $O4$, 40-70) - показывает изменение в Амакринах и обратных функциях связи от ганглиозных клеток к Амакринам;

P_i^{16} = (Паттерн ЭРГ *stade state*, *A pik-to-pik*, 50-120) – показывает изменение в ганглиозных клетках и аксоне;

P_i^{17} = (Паттерн ЭРГ, $P50$, 45-90) – показывает изменение в фоторецепторах и колбочковых биполярах;

P_i^{18} = (Паттерн ЭРГ, $N95$, 30-110) – показывает изменение в ганглиозных клетках и аксонах колбочек (больше в макуле).

Вычисления показателей производится на основе рекомендаций Международного общества клинических электрофизиологов зрения ISCEV [8]:

- амплитуда Aa вычисляется от 0 до \max негативного;
- амплитуда Ab вычисляется от a до \max позитивного;
- время кульминации Ta вычисляется от оси до \max негативного;
- время кульминации Tb вычисляется от оси до \max позитивного;
- амплитуда *A pik-to-pik* вычисляется от \max негативного до \max позитивного;
- амплитуда $P50$ вычисляется от первого отрицательного отклонения до \max $P50$;
- амплитуда $N95$ вычисляется от $P50$ (или изолинии) до $N95$;
- амплитуды $O1$, $O2$, $O3$, $O4$ вычисляются от \max негативного до \max позитивного.

База знаний содержит набор продукционных правил. Левая и правая части правил – подсети семантической сети (универсума). Левая часть – набор показателей, правая – набор концептов с атрибутами.

Примеры продукционных правил.

Если P^1 (Максимальная ЭРГ, Аа, 60), то $A_1^1(100.100.70)$ (цвет тусклый);

Если P^1 (Максимальная ЭРГ, Аа, 100), то $A_1^1(150.150.79)$ (цвет умеренный);

Если P^1 (Максимальная ЭРГ, Аа, 150), то $A_1^1(200.220.100)$ (цвет яркий);

...

Если P^{13} (Фотопическая ЭРГ, Аа, 15), то $A_2^2(10)$ (ширина маленькая);

Если P^{13} (Фотопическая ЭРГ, Аа, 30), то $A_1^1(20)$ (ширина средняя);

Если P^{13} (Фотопическая ЭРГ, Аа, 45), то $A_1^1(30)$ (ширина большая);

Если P^{14} (Фотопическая ЭРГ, Аb, 50), то $A_3^1(30.200.70)$ (цвет тусклый);

Если P^{14} (Фотопическая ЭРГ, Аb,100), то $A_3^1(100.100.150)$ (цвет умеренный);

Если P^{14} (Фотопическая ЭРГ,Аb,150), то $A_3^1(200.220.130)$ (цвет яркий);

Если выполняется левая (условная) часть правила, то правая часть (заключение) правила используется для построения КО сетчатки.

Тогда для оценки состояния КО каждого вида, фрагмента сетчатки, может быть использован комбинированный метод визуализации, заключающийся в изменении размера изображения и его цвета в зависимости от выбранных показателей.

Связь различных типов ЭРГ и КО представлена в виде онтологии на рис. 2.

Примеры КО сетчатки в нормальном состоянии и некоторых патологиях представлены на рис. 3. Нетрудно заметить, что при некоторых КО (например, нижние КО для случаев преклинической и пролиферативной диабетической ретинопатии) недостаточно наглядны по причине наличия неполноты и неопределенности в исходной информации.

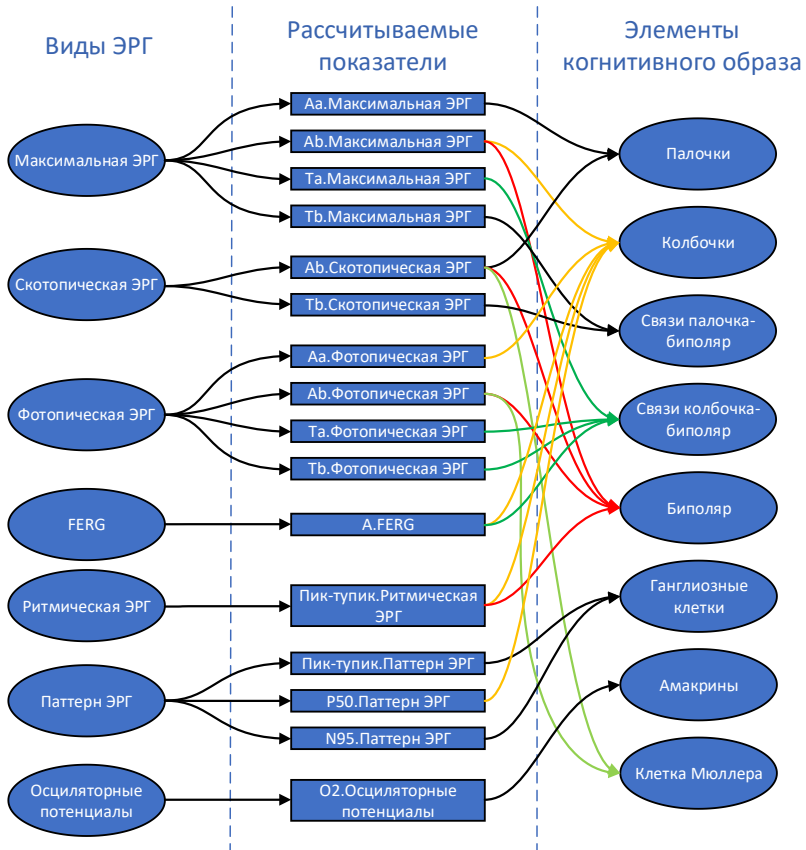


Рис. 2. Взаимосвязь видов ЭРГ и элементов КО

Для повышения наглядности и информативности КО было предложено использовать метод построения КО с применением нечеткой логики (нечетких продукционных правил). Нечеткие продукционные правила имели традиционный вид IF $(\beta_i \text{ IS } \alpha_i)$ AND... AND $(\beta_n \text{ IS } \alpha_n)$ THEN $(\beta_{n+1} \text{ IS } \alpha_{n+1}, k)$, где β_i – соответствующие параметры (показатели), α_i – лингвистические значения, $i=1, \dots, n+1$, k – коэффициент правдоподобия (уверенности) значения α_{n+1} , $k \in [0, 1]$ (по умолчанию $k=1$).

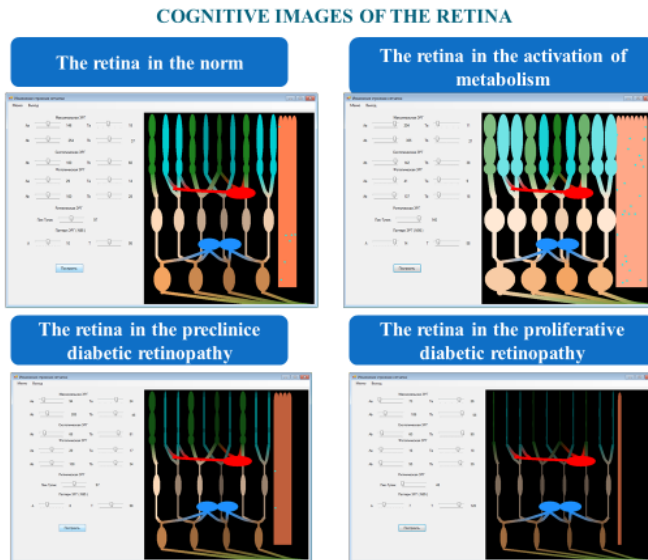


Рис. 3. Примеры КО сетчатки

Для нечеткого вывода был применен алгоритм Мамдани. Приведем примеры лингвистических переменных для входных и выходных данных при построении КО.

Имя переменной: “размер”, терм-множество: {малый, средний, большой}. Носитель (универсум): интервал от минимального до максимального значения размера соответствующего КО.

Имя переменной: “цвет”, терм-множество: {яркий, тусклый}. Носитель (универсум): интервал от минимального до максимального значения цвета соответствующего КО.

Имя переменной: “здоровье”, терм-множество: плохое, удовлетворительное (нормальное), хорошее}. Носитель (универсум): интервал целых чисел от 20 до 100 (в %).

Универсум и функции принадлежности определяются с помощью экспертов-физиологов.

Приведем примеры нечетких продукционных правил для выходной переменной “здоровье” (в нотациях библиотеки FuzzyLogicLibrary):

- 1) if (Size is small) or (Color is dim) then Health is bad;
- 2) if (Size is big) or (Color is bright) then Health is good;
- 3) if (Size is middle) or (Color is dim) then Health is bad;

4) if (Size is big) or (Color is dim) then Health is normal.

Построенные с помощью нечеткой логики КО патологий сетчатки оказались более информативными и наглядными медиков (рис. 4, 5).

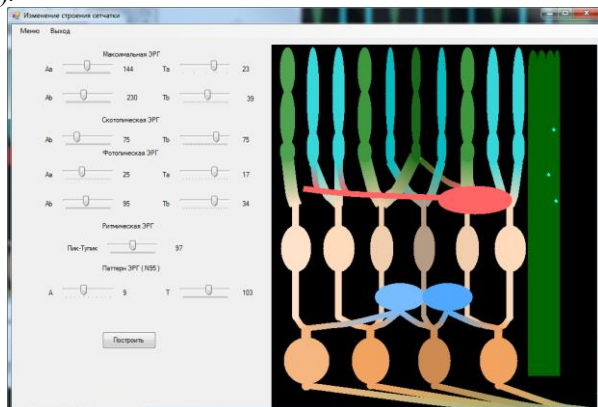


Рис. 4. Пример КО сетчатки при преклинической диабетической ретинопатии

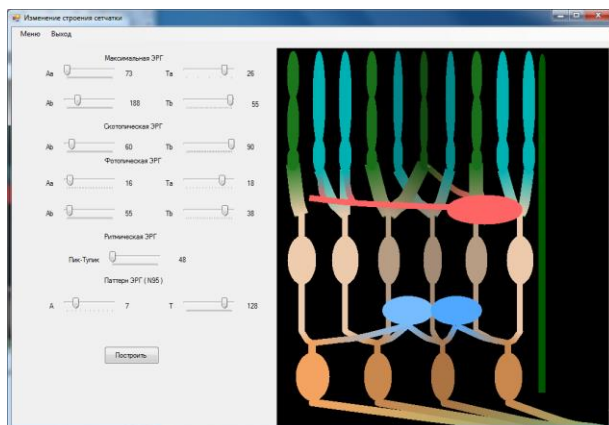


Рис. 5. Пример КО сетчатки при пролиферативной диабетической ретинопатии

Нетрудно заметить, что применение нечеткой логики позволило существенно повысить качество КО (см. рис. 3, нижние КО). В настоящее время проводятся исследования и разработки по

применению других алгоритмов вывода, в частности алгоритма Сугено.

3. ЗАКЛЮЧЕНИЕ

Полученные результаты показали, что отображение изменений функционального состояния сетчатки глаза на основе КО с применением нечеткой логики существенно лучше, чем без применения нечеткой логики. В дальнейшем планируется использование также других алгоритмов вывода и получение сравнительных характеристик генерируемых КО. Для реализации КО выбран объектно-ориентированный подход, язык реализации – C#, платформа Visual Studio Ultimate. Данная среда удобна также для отображения динамики развития патологии сетчатки с помощью КО.

Литература

1. Зенкин А.А. Когнитивная компьютерная графика /Под ред. Д.А. Поспелова. – М.: Наука, 1991.
2. Башлыков А.А. Основы конструирования интеллектуальных систем поддержки принятия решений в атомной энергетике: учебник /А.А. Башлыков, А.П. Еремеев. – М.: ИНФРА-М, 2018.
3. Alexander Eremeev, Sergey Ivliev. Using Convolutional Neural Networks for the Analysis of Nonstationary Signals on the Problem Diagnostics Vision Pathologies // Proceedings of the 16th Russian Conference on Artificial Intelligence RCAI 2018, Moscow, Russia, September 24-27, 2018. – Pp. 164-175.
4. Колосов О.С., Короленкова В.А., Пронин А.Д., Зуева М.В., Цапенко И.В. Построение амплитудно-частотных характеристик сетчатки глаза и формализация их параметров для использования в системах диагностики // Мехатроника, автоматизация, управление. 2018. 19 (7). – С. 451–457.
5. Еремеев А.П., Ивлиев С.А. Методы и программные средства прототипа интеллектуальной системы поддержки принятия решений для анализа и диагностики сложных патологий зрения // Вестник МЭИ. 2020. № 5. – С. 140-147. DOI: 10.24160/1993-6982-2020-5-140-147.
6. Aleksandr P. Eremeev, Irina V. Tsapenko. THE USE OF COGNITIVE GRAPHICS IN THE DIAGNOSIS OF COMPLEX VISION PATHOLOGIES // International Journal “Information Theories and Applications”, Vol. 26, Number 1, 2019. – Pp. 83-99. ISSN 1310-0513 (printed). ISSN 1313-0463 (online).
7. Шамшинова А. М. Электроретинография в офтальмологии. – М.: Медика, 2009.
8. Типы ЭРГ, рекомендуемые ISCEV <https://zreni.ru/print:page,1,3066-metodicheskie-osnovy-sovremennoy-elektroretinografii.html>

СЕМИОТИЧЕСКИЕ КОГНИТИВНЫЕ КАРТЫ

*Кулинич А.А., к.т.н., с.н.с.
ИПУ РАН им В.А. Трапезникова
e-mail: alexkul@rambler.ru*

1. ВВЕДЕНИЕ

Многочисленные исследования [1,2,3,4] когнитивных карт и методов их анализа заложили теоретический фундамент для создания прикладных коммерческих систем моделирования когнитивных карт. Интерес к методологии когнитивного моделирования связан, по-видимому, с простотой построения модели ситуации и ее наглядностью. Когнитивная карта представляется в виде ориентированного знакового графа. При этом возможны самые разные интерпретации дуг орграфа, значений в вершинах, методов вычисления прогнозов развития ситуаций, решения обратных задач, а также, методов исследования ситуаций с помощью когнитивных карт. В работах [5,6] предложена классификация когнитивных карт, методов их анализа, а также описаны российские системы когнитивного моделирования. В работе [7] рассмотрены вопросы применения методологии когнитивного моделирования в процессах управления. В работе [8] рассмотрены вопросы применения когнитивных карт при моделировании игровых постановок в управлении и мультиагентных системах. В работе [9] дан обзор применения методологии когнитивного моделирования в областях социально-политических наук, в сфере образования, медицине, бизнесе и др.

Поддержка принятия решений основана на получении прогноза изменения значений всех факторов при изменении значения любого другого фактора, которая решается с помощью системы конечно-разностных уравнений вида: $X(t+1)=WX(t)$ где $X(t)$ – вектор начального приращения значений факторов, $t=0$; $X(t+1)$ – вектор изменений значений факторов в моменты времени $t=1, \dots, n$, W - матрица смежности орграфа.

Задача принятия решений заключается в анализе динамики изменения состояния системы, представленной векторами состояния в последовательные моменты времени $X(t)$ и их интерпретации.

Интерпретация заключается в представлении вектора состояния в другой системе $X(t) \rightarrow \mathcal{G}$, где \mathcal{G} - интерпретирующая система. В

настоящее время интерпретирующая система это субъективная понятийная система лица, принимающего решение – его знания о предметной области и его способности: рассуждения, обобщение, воображение, способные представить вектор состояния в виде понятий предметной области, в которой модель построена.

Таким образом, когнитивные карты как инструмент анализа ситуаций представляют собой качественные (грубые) субъективные модели, ориентированные на быструю генерацию множества решений и их правдоподобное обоснование и интерпретацию.

Однако обоснование и интерпретация результатов анализа также субъективна и считается аргументированной только для аналитика или эксперта, который построил когнитивную карту. Ответственные лица, принимающие решения, имеющие иные понятийные представления о предметной области, могут не принять такие решения или считать их недостаточно обоснованными.

Для поддержки интерпретации прогнозов развития ситуаций, полученных на когнитивных картах, предлагается расширить существующую парадигму представления знаний в когнитивных картах как множества концептов и причинно-следственных отношений на этом множестве, на ее представление в виде знаковой системы, т.е. семиотическое представление когнитивных карт. Семиотические когнитивные карты кроме решения традиционных задач когнитивной методологии (прогноз развития ситуации, решения обратной задачи и др.) ориентированы на поддержку интерпретации получаемых прогнозов и решений.

2. СЕМИОТИЧЕСКИЕ, ЗНАКОВЫЕ СИСТЕМЫ

Исследованием знаков и знаковых систем занимались известные математики и логики. Практический интерес представляют работы немецкого логика Г.Фреге [10]. Его модель знака (треугольник Фреге) включает: имя, смысл и значение знака. Имя знака – это символ, (изображение), обозначающее в сознании человека объект реального мира. Под смыслом знака Фреге понимает описание объекта реального мира, отличающего его от других объектов, и под значением знака Фреге понимает реальный объект, который определяется смыслом знака. Представления знаний в виде знака связывает ментальные процессы, в которых принимает участие имя и смысл знака, определяемые субъектом с объектами реального мира (значением), обозначаемые именем и определяемые его смыслом. Это означает, что между реальным миром и знаками существуют отношения

моделирования, позволяющее с помощью знаков осуществлять процесс мышления со знаками, а не реальными объектами.

Сложные объекты, состоящие из множества знаков, связанных разными отношениями образуют семиотические системы. Классики семиотики [11, 12] определили три аспекта семиотических систем – синтаксический, семантический и прагматический.

В прикладной семиотике [13] модель семиотической системы известна как квадрат Поспелова. Квадрат Поспелова включает: метазнак, определяющий имя семиотической системы, синтаксис, определяющий правила построения знаковой системы; семантику, определяющую смысл, основные свойства семиотической системы; прагматику, определяющую основные действия, которые возможно выполнить в рамках этой семиотической системы.

3. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ СЕМИОТИЧЕСКИХ КОГНИТИВНЫХ КАРТ

Математическая модель семиотической когнитивной карты основана на субъективной качественной модели представления знаний эксперта о динамических системах. Элементами этой модели является модель знака Г. Фреге [10], связывающая реальный мир (денотат) с ментальными представлениями о мире – знаниями эксперта.

Математическая модель семиотической когнитивной карты включает три, связанные параметрами модели: синтаксическую, семантическую и прагматическую [14]. Синтаксическая модель ответственна за представление отношений между знаками, описывающими реальность. Семантическая модель определяет отношения между знаками и тем, что они обозначают в реальном мире, а прагматическая модель отвечает за отношения между знаками и теми, кто ими пользуется (ЛППР).

Считается, что определена предметная область $DF = \langle D, F \rangle$, где D – множество понятий и F – множество параметров предметной области.

Синтаксическая модель. Синтаксическая модель – это когнитивная карта в ее традиционном понимании. Определено имя моделируемой ситуации (объекта) – $d \in D$ в предметной области DF . Для ситуации с этим именем определено множество имен параметров $F^* = \{f_i\}$, $F^* \subset F$, $i = 1, \dots, N$. Для каждого параметра задано множество значений, $Z = \{Z_i\}$ в виде упорядоченного множества лингвистических значений, т.е. $Z_i = \{z_{i1}, \dots, z_{iq}\}$, $z_{iq+1} \succ z_{iq}$, $q = 0 \dots n-1$. Вектор значений всех параметров ситуации в момент времени t , $Z(t) = (z_{1e}, \dots, z_{nq})$, называется

ее состоянием. Это состояние в начальный момент времени определяется именем d , $Z(0) \leftrightarrow d$.

В синтаксической модели определено причинно-следственное отношение W , заданное на множествах значений параметров. Динамика изменения состояния представлено как отображение:

$$W: Z(t) \rightarrow Z(t+1), \quad (1)$$

где $Z(t) \in Z_i \times Z_i$.

То есть, синтаксическая модель определена кортежем:

$$\langle F, Z, W, Z(t) \rangle, \quad (2)$$

где F - множество параметров, Z - множество множеств значений параметров, W - причинно-следственное отношение на множестве значений параметров, $Z(t)$ - вектор значений всех параметров.

Семантическая модель. Семантическая модель связана параметрами с синтаксической моделью и является надстройкой над синтаксической моделью. В семантической модели представлены возможные состояния синтаксической модели (2) в виде частично упорядоченного множества имен классов состояний.

Модель основана на представлении пространства возможных состояний динамической системы (1) $SS = \times_i Z_i$ как признакового семантического пространства (*Semantic Space*).

В признаковом семантическом пространстве состояния ситуации представляются как понятия (состояния-денотаты), которые определяются именами и векторами значений признаков, определяющие их содержание (смысл). В семантических пространствах ситуации с близкими значениями параметров образуют классы состояний, а между классами определяются отношения (класс-подкласс или род-вид), т.е. определяется понятийная структура.

В работе [15] был предложен метод структуризации пространства состояний SS динамической системы (1) на вложенные области возможных состояний $SS(d^H) \subset SS$, имеющие искусственные имена d^H , $SS(d^H) \leftrightarrow d^H$, и определяющие класс состояний системы (1).

На рисунках 1-6 дана иллюстрация предложенного метода структуризации пространства состояний, на примере системы с двумя параметрами (признаками).

На рисунке 1 показан пример семантического пространства, для объекта v , имеющего два признака $F = \{f_1, f_2\}$, значения которых определяются из множеств значений (Z_1, Z_2), начальное состояние объекта $Z(0) = (z_{1p}, z_{2p})$, показано как точка с координатами (z_{1p}, z_{2p}) , в пространстве $SS = Z_1 \times Z_2$.

Рассмотрим окрестность точки семантического пространства, представляющей начальное состояние $Z(0) = (z_1 \pm \varepsilon_1, z_2 \pm \varepsilon_2)$, где $z_i \pm \varepsilon_i \in Z_i$, $i=1,2$. Окрестность $z_i \pm \varepsilon_i$ значений i -го признака состояния $Z(0)$ определяет класс состояний и называется интервалом толерантности класса по этому признаку $\Delta_i = [z_{ik} + \varepsilon_{ik}, z_{ik} - \varepsilon_{ik}]$.

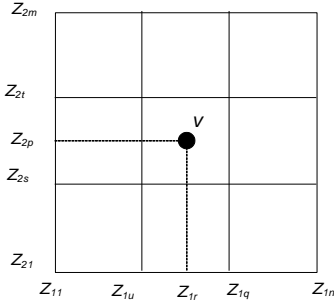


Рис.1. Семантическое пространство

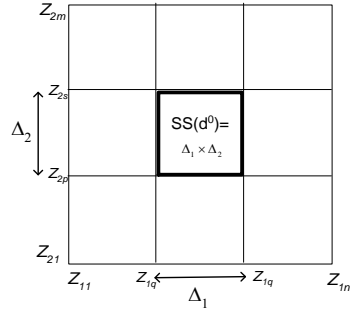


Рис.2. Базовое понятие

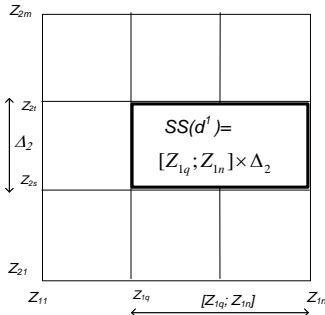


Рис.3. Положительное обобщение

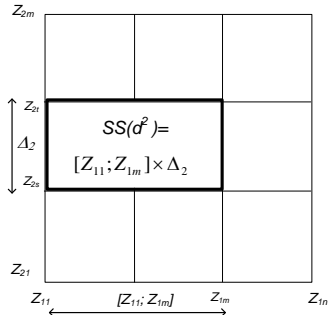


Рис.4. Отрицательное обобщение

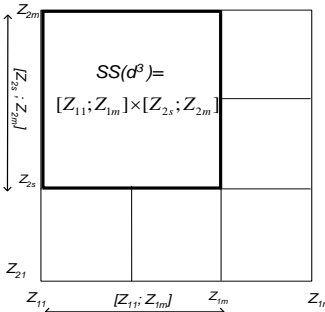


Рис.5. Обобщение по двум признакам

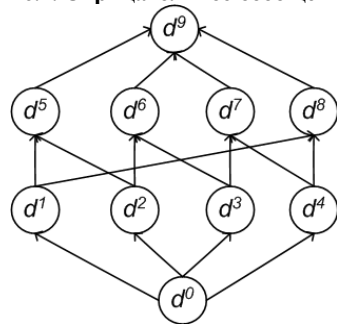


Рис.6. Концептуальный каркас

Область семантического пространства $SS(d^0) \subseteq SS$, полученная прямым произведением интервалов толерантности всех признаков состояния $Z(0)$ называется содержанием базового понятия: $SS(d^0) = \Delta_1 \times \Delta_2$, $SS(d^0) \subseteq S$.

На рисунке 2 дано графическое представление базового понятия состояния $Z(0)$. Имя базового понятия совпадает с именем начального состояния, т.е. $d = d^0$, $d^0 \leftrightarrow SS(d^0)$.

На рисунке 3 показано качественное положительное обобщение базового понятия d^0 по признаку 1. Получаем новый класс состояний динамической системы (1) с новым именем d^1 . Это новое понятие определяется в семантическом пространстве как область $SS(d^1) \subset SS$:

$$SS(d^1) = \Delta_2 \times [Z_{1q}; Z_{1n}], SS(d^1) \subset SS, SS(d^0) \subset SS(d^1).$$

На рисунке 4 показано содержание качественного отрицательного обобщенного понятия базового понятия d^0 по признаку 1. Здесь новое понятие с именем d^2 и областью $SS(d^2) = [Z_{11}; Z_{1m}] \times \Delta_2$, $SS(d^2) \subset SS$, $SS(d^0) \subset SS(d^2)$ определяет новый класс состояний системы (1).

Пример качественного обобщения базового понятия по двум признакам представлен на рисунке 5. Новое понятие с именем d^6 имеет область класса состояния $SS(d^6) = [Z_{11}; Z_{1m}] \times [Z_{2s}; Z_{2m}]$, $SS(d^6) \subset SS$, $SS(d^0) \subset SS(d^6)$, $SS(d^2) \subset SS(d^6)$.

Таким образом могут быть получены 3^N обобщений базового понятия d^0 по разным признакам в разных сочетаниях.

Было показано, что, полученные имена d^H по вложению областей состояний $SS(d^H)$ образуют частично упорядоченное множество имен $\{d^H\}$ классов состояний $CF = (\{d^H\}, \leq)$, которое называется качественным концептуальным каркасом (*Conceptual Framework*) (Рис.6) и определяет качественную онтологию предметной области, для которой была построена синтаксическая модель (2).

Таким образом, семантика синтаксической модели (2) определена качественным концептуальным каркасом, в котором упорядочены классы состояний динамической системы (1):

$$CF = (\{d^H\}, \leq), \quad (3)$$

где $d^H \leftrightarrow SS(d^H)$ – имена классов состояний, $Z(t) \in SS(d^H)$ – состояние синтаксической модели.

Прагматическая модель. В семиотике прагматика отвечает за отношения между знаками и теми, кто ими пользуется. В прагматической модели рассматривается оценка полезности состояния системы для лица, принимающего решение. Оценка основана на

определении коэффициентов предпочтений эксперта α_{ij} относительно значений параметров в векторе состояний ($Z(t)$).

Оценка состояния $O(Z(t))$ представляется линейной сверткой [22]:

$$O(Z(t)) = \sum_j \alpha_{ij} x_j(t), j=1, \dots, n. \quad (4)$$

где $x_j(t) \in [0, 1]$ – отображение φ лингвистических значений параметров $z_j(t)$ на отрезок числовой оси $[0, 1]$, т.е. $\varphi: z_j(t) \rightarrow x_j(t) \in [0, 1]$, $z_j(t) \in Z(t)$, $O(Z(t)) \in [0, 1]$.

Все три модели связаны между собой общими параметрами, и поэтому изменения состояния любой из них приведет к изменению состояний других моделей.

4. РЕШЕНИЕ ОБРАТНОЙ ЗАДАЧИ В СЕМИОТИЧЕСКОЙ КОГНИТИВНОЙ КАРТЕ

В синтаксической модели параметры системы заданы на линейно упорядоченных множествах лингвистических значений, т.е. в качественных шкалах. Методы решение обратных задач для динамических систем параметры, которых определены качественно или нечетко (нечеткие множества, лингвистические переменные и т.д.) рассматривались в работах [16, 17]. Эти методы используются для решения обратной задачи в синтаксической модели сложной системы.

Предложен метод [18], основанный на решении обратной задачи в семиотических моделях, путем последовательной передачи результатов решения из прагматической модели в синтаксическую, а затем в семантическую. В результате решения в прагматической, синтаксической и семантической моделях в семиотической системе устанавливается состояние соответствующее сформулированной цели.

Пусть начальный вектор параметров $Z(t)$, прагматическая оценка которого O . Задача заключается в том, чтобы изменить существующий вектор состояния $Z(t)$ на целевой вектор параметров U_{goal} , имеющий лучшую прагматическую оценку O^* . Эта задача решается как обратная задача с учетом существующих причинно-следственных отношений W , заданных при описании синтаксической модели системы.

$$U = W^{-1} \circ U_{goal}$$

где \circ процедура обратного вывода.

Решение обратной задачи - это множество решений $U = \{U_i\}$, где $U_i = (u_{1e}, \dots, u_{nq})$ – вектор состояния, $u_{ij} \in Z_i$. Если представить эти решения в виде точек с координатами элементов векторов в структурированном семантическом пространстве, то эти решения

(точки в семантическом пространстве) попадают в разные области $SS(d^H)$, характеризующие классы состояний с разными именами d^H .

Таким образом, формальное решение обратной задачи U , дает нам множество имен классов решений $\{d^{H*}\}$, которое представляется в семантической модели как частично упорядоченное множество имен классов состояний $CF^* = (\{d^{H*}\}, \leq)$, $CF^* \subset CF$. Решения обратной задачи в семантической модели называется концептуальным каркасом решений.

5. ОПРЕДЕЛЕНИЕ И ИНТЕРПРЕТАЦИЯ ИМЕН КЛАССОВ РЕШЕНИЙ

При формальном решении обратной задачи в семантической модели имена классов решений определены формально в виде математических символов d^H . Однако, в поддержке принятия решений необходимы общепринятые имена, обозначающие реальные ситуации или объекты (денотаты).

В работе [18] были рассмотрены методы определения имен классов решений, основанных на исследованиях психологов процессов классификации и категоризации. Первый метод определения составного имени нового класса основан на добавлении к понятию базового класса оценки отличительного признака или признаков [19].

В синтаксической модели значения параметров f_i определены в виде вектора их значений – $Z(t) = (z_{1j}, \dots, z_{nk})$, где $z_{ij} \in Z_i$, Z_i – множество значений i -го признака. В семантической модели определена область $SS(d^0) = \times \Delta_i$, где $\Delta_i = z_{ij} \pm \varepsilon_i$, в которой определено имя базового понятия

d_i^0 . Решение обратной задачи - это вектор: $U = (u_{1j}, \dots, u_{nk})$, $u_{ij} \in Z_i$.

Результат решения обратной задачи в семантической модели (3) будем представлять в виде вектора $A = (a_1, \dots, a_n)$, $a_i \in \{-1, 0, 1\}$, где $a_i = -1$, если $u_{ij} < z_{ij} - \varepsilon_i$, $a_i = 1$, если $u_{ij} > z_{ij} + \varepsilon_i$, $a_i = 0$, если $u_{ij} \in z_{ij} \pm \varepsilon_i$.

В векторе решения $A = (a_1, \dots, a_n)$, $a_i \in \{1, 0, -1\}$, a_i – качественно характеризует оценку значения i -го параметра (f_i) в решении обратной задачи: т.е. (+1) – параметр имеет большое значение, (-1) – параметр имеет малое значение. Этот вектор можно представить в виде вектора с лингвистическими оценками $Z^* = (z_1^*, \dots, z_n^*)$, где $z_i^* =$ «Большой», если $a_i = 1$, $z_i^* =$ «Малый», если $a_i = -1$. Тогда, составное искусственное имя класса решений будет иметь следующий вид:

$$d^H = d^0 \ \& \ z_i^* \quad i (a_i \neq 0)$$

Например, если имя базового класса $d^0 =$ «Шар», то возможные составные имена новых классов по признаку «Размер» могут быть такими: $d^1 =$ «Шар большой» или $d^2 =$ «Шар малый».

Метод определения составного искусственного имени дает решение – имя класса решений на ограниченном естественном языке как перечисление значений отличительных признаков класса. Общепринятые имена классов решений будем искать в сети Интернет.

6. ОПРЕДЕЛЕНИЕ ИМЕН КЛАССОВ РЕШЕНИЙ В ИНТЕРНЕТ

Определение общепринятого имени класса решения основано на психологической теории прототипов (категорий) [20]. В этой теории считается, что имя класса (категории) определяется именем наиболее характерного представителя класса. В работе [21] описан подход поиска имен классов решений, основанный на их поиске в сети Интернет. Т.е., в Интернет находятся имена, которые часто употребляются совместно с составным именем класса решений d^{H} .

Этот подход заключается в следующем.

Создание корпуса текста предметной области. Для этого автоматически с помощью программных модулей извлечения информации из сети Интернет (например, модуль googlesearch) получается текстовая информация с множества сайтов, включающих ключевое слово, совпадающее с именем базового понятия d^0 .

Выделение в корпусе текста родовидовых отношений («класс-подкласс»). Выделение таких отношений необходимо для определения возможных имен классов состояний в концептуальном каркасе решений. Для выделения родовидовых отношений из корпуса текста используется метод лексико-синтаксических шаблонов [22]. По результатам работы лексико-синтаксических шаблонов строится словарь гиперонимов-гипонимов, включающий предложения из корпуса текста, в которых обнаружены родовидовые отношения [21].

Словарь определим следующим кортежем: $\langle \text{Hiper}, \text{Hipo}, \text{Contex} \rangle$, где Hiper – гиперонимы, Hipo – гипонимы, Contex – предложения, в которых обнаружены гиперонимы. Словарь гиперонимов-гипонимов и контекста строится из корпуса текста автоматически.

Определение векторов совместной встречаемости слов в корпусе текста. Здесь определяются имена в корпусе текста, которые часто встречаются совместно с составным именем класса решений d^{H} . Поиск таких имен основан на дистрибутивном анализе текста, при котором синтаксические связи в предложениях корпуса текста игнорируются. Корпус текста W_r для дистрибутивного анализа нормализуется, т.е. в корпусе текста остается множество предложений $W_r = \{St_i\}$, содержащие ряд существительных единственного числа

$(wr_{ij}), St_i=(wr_{ij})$. Множество всех неповторяющихся слов в предложениях образует словарь предметной области, $Vc=\{w_i\}$.

Для определения векторов слов используется программная библиотека word2vec. Для получения векторов совместной встречаемости слов нейронная сеть word2vec [23] обучается на нормализованном корпусе текста Wr . После обучения модели word2vec, векторная модель совместной встречаемости слов в корпусе текста может быть представлена как функция $w2v(\bullet)=R$, где \bullet - аргументом функции может быть любое слово из словаря Vc , а вектор $R=(w_i/r_{i1}, \dots, w_n/r_{in})$ характеризует совместную встречаемость r_{ij} с другими словами (w_i, \dots, w_n) из словаря предметной области Vc .

В векторную модель word2vec в качестве аргумента функции $w2v(\bullet)$ может быть подставлено сочетание слов $w_i \& \dots \& w_q, w_i, \dots, w_q \in Vc$, из словаря предметной области. Если подставить в качестве аргумента составное имя решения d^{th} , $w2v(d^{th})=R_s$ - то вектор $R_s=(w_i/r_{i1}, \dots, w_n/r_{in})$ будет включать имена $w_i \in Vc$, которые совместно употребляются с именем d^{th} и упорядочены согласно характеристикам частоты их совместной встречаемости r_{ij} .

Определение альтернативных имен классов решений.

Вектор совместной встречаемости составного имени решения и слов предметной области из словаря R_s может содержать слова, которые не определяют имя класса решений. Это могут имена (слова), определяющие имена экземпляров класса решения. Отметим, что гиперонимы в словаре гиперонимов-гипонимов определяют имя класса решений. Для получения альтернатив решений берем пересечение слов имен из вектора R_s и словаря гиперонимов-гипонимов. Тогда, альтернативы имен классов решений – это кортеж: $\langle W_s \cap \text{Hyper}, \text{Context} \rangle$, где $W_s=(w_i, \dots, w_n)$ слова вектора R_s , Hyper – гиперонимы словаря, Context – предложения, в которых обнаружены гиперонимы.

Таким образом, имена классов решений в виде множества слов-гиперонимов предметной области, упорядоченного по убыванию их совместной встречаемости с составным именем решений и предложений, поясняющих эти имена классов решений.

Метод поиска имен классов решений в сети Интернет был опробован. Был построен программный макет семиотической когнитивной карты на базе системы когнитивного моделирования «Канва» [24]. Для поиска имен классов решений в сети Интернет разработан программный макет на языке Python3. При решении обратной задачи, например, было получено составное имя «Недовольные олигархи». Интерпретация этого имени в предметной области позволило получить два альтернативных имени класса

решений, «Вредные олигархи» и «Недоверчивые олигархи». Для классов решений с такими именами легко найти их экземпляры (имена олигархов), определить их поведение, влияние на экономику страны.

7. ЗАКЛЮЧЕНИЕ

В работе предложены семиотические когнитивные карты, ориентированные на поддержку интерпретации результатов когнитивного моделирования. Предложена математическая модель семиотической когнитивной карты, состоящая из трех связанных параметрами моделей: синтаксической, семантической и прагматической. Предложен метод интерпретации решений обратной задачи, который основан на поиске имен классов решений в сети Интернет. Для этого предложены методы извлечения и обработки неструктурированной информации Интернет. Это методы лексико-синтаксических шаблонов, для выделения родовидовых отношений в корпусе текста и дистрибутивного анализа для определения часто употребляемых имен классов решений. Эксперименты показали работоспособность предложенных методов и подходов. Применение предложенной модели семиотической когнитивной карты позволит расширить возможности систем поддержки принятия решений в динамических плохо определенных ситуациях.

Литература

1. Axelrod R. The Structure of Decision: Cognitive Maps of Political Elites. – Princeton. University Press, 1976.
2. Kosko B. Fuzzy Cognitive Maps.//International Journal of Man-Machine Studies, (1986) 24, 65-75.
3. Carvalho J.P., Jose A.B. Tome. Rule based fuzzy cognitive maps. –Fuzzy causal relations. Computational Intelligence Modeling. Control and Automation, Ed. By M. Mohammadian, 1999.
4. Федулов А.С., Борисов В.В Модели системной динамики на основе нечетких реляционных когнитивных карт // Системы управления, связи и безопасности. 2016. № 1. С. 66-80.
5. Кулинич А.А. Компьютерные системы моделирования когнитивных карт: подходы и методы./Проблемы управления, 2010, № 3 С.2-16.
6. Кулинич А.А. Компьютерные системы анализа ситуаций и поддержки принятия решений на основе когнитивных карт: подходы и методы. / Проблемы управления, 2011, № 4 С.31-45.
7. Авдеева З.К., Коврига С.В., Макаренко Д.И., Максимов В.И. Когнитивный подход в управлении // Проблемы управления.- 2007. - №3. - С. 2-8.
8. Новиков Д.А. «Когнитивные игры»: линейная импульсная модель // Проблемы управления. - № 3. - 2008. - С. 14-22.

9. Papageorgiou E.I. A Review study on Fuzzy Cognitive Maps and their applications during the last decade // Proc. of IEEE International Conference of Fuzzy Systems (FUZZ IEEE). - 2011, Taipei, Taiwan. - P. 828 – 835.
10. Frege G. On sense and nominatum. Readings in philosophical analysis. (1949).
11. Пирс Ч.С. Что такое знак? // Вестник Томского Гос. Университета. Философия. Социология. Политология. - 2009. - № 3(7). - С. 88-95.
12. Моррис Ч.У. Основания теории знаков. - Семиотика: антология. Изд. 2-е, перераб. и доп. М.–Екатеринбург. - 2001. - С. 45–97.
13. Поспелов Д.А., Осипов Г.С. Прикладная семиотика//Новости ИИ. - 1999. - №1.
14. Кулинич А.А. Семиотический подход в моделировании и принятии решений в плохо определенных сложных ситуациях//Материалы 11-й международной конференции «MLSD'2018». ИПУ РАН. 2018. - Т.2. - С. 298-301.
15. Кулинич А.А. Концептуальные каркасы онтологий слабо структурированных предметных областей // Искусственный интеллект и принятие решений. - 2014. - № 4. - С. 31-41.
16. Pappis C.P., Sugeno M. Fuzzy relational equations and the Inverse Problem// Fuzzy Sets and Systems. - 1985. - № 15. - p. 79-90.
17. Pedrycz W. Algorithms for solving fuzzy relational equations in a probabilistic setting// Fuzzy Sets and Systems. - 1990. - № 38. - p. 313-327.
18. Кулинич А.А. Решение обратной задачи в семиотических системах в условиях неопределенности//IX-Международная научно-практическая конференция «Интегрированные модели и мягкие вычисления в искусственном интеллекте». Сборник научных трудов. (Коломна, 19-22 мая 2019 г.). - Т2. - С. 572-582.
19. Выготский Л. С. Мышление и речь // Собр. соч.: Т. 2. М.: Педагогика, 1982. С. 5-361.
20. Rosch E. Principles of categorization. In: Rosch E., Lloyd B.D. Cognition and categorization. N.Y.: Lawrence Erlbaum association. 1978.
21. Kulnich A. Semiotic Models in Monitoring and Decision Support Systems. In: Kovalev S.M. (eds) Artificial Intelligence. RCAI 2021. Lecture Notes in Computer Science. Springer, Cham. 2021. - V. 12948. - pp. 303-316.
22. Большакова Е.И., Васильева Н.Э., Морозов С.С. Лексико-синтаксические шаблоны для автоматического анализа научно-технических текстов//10-я Национальная конференция по искусственному интеллекту (КИИ-2006). Труды конференции. М.: Физматлит. - 2006. - Т. 2. - С.506-524.
23. Mikolov, et al. Distributed representations of words and phrases and their compositionality. In Advances in Neural Information Processing Systems: 27th Annual Conference on Neural Information Processing Systems. Proc. Dec. 5-8, 2013, Lake Tahoe, Nevada, United States. - 2013. - p. 3111–3119.
24. Кулинич А.А. Когнитивная система поддержки принятия решений «Канва»// Программные продукты и системы. №3, 2002 г. с. 25-28.

КОГНИТИВНЫЕ ОСНОВЫ КЛАССИФИКАЦИИ ЭЛЛИПСИСОВ

*Найденова К.А. к.т.н., ст. н. с.,
e-mail: ksennaidd@gmail.com,
Мартирова Т.А., мл. н. с.
e-mail: martta462@yandex.ru
Военно медицинская академия*

1. ВВЕДЕНИЕ

Лингвистами давно осознается ограниченность подхода к разрешению эллипсисов в рамках синтактико-сематического анализа предложений [1]. В [2] сравниваются ответы на основные вопросы в разрешении эллипсисов, которые ставятся в рамках генеративной лингвистики и когнитивного подхода:

- Является ли эллипсис синтаксической единицей?
- Определяется ли смысл эллипсиса только его антецедентом?
- Равнозначна ли реконструкция пропущенной части предложения пониманию эллипсиса?

Адепты генеративной лингвистики на эти вопросы отвечают утвердительно. «Когнитивисты» склоняются к отрицательным ответам [3].

В когнитивном подходе утверждается, что смысл пропущенной части тесно связан со смыслом всего предложения. Процесс понимания эллипсиса не означает, что мы сначала должны восстановить его, а потом обратиться к пониманию всего предложения. Понимание предложения влечет и понимание эллипсиса. Смысл эллипсиса определяется как явно выраженной частью предложения, так и знаниями предметной области, включая контекст, лингвистические, прагматические, энциклопедические знания и т.д.

Генеративная лингвистика не интересуется реальными ментальными процессами человека при разрешении эллипсиса. Когнитивные лингвисты предполагают, что конструирование смысла есть концептуальный процесс [4, 5].

В работе [6] явление эллипсиса связывается со структурной неполнотой синтаксической конструкции. Устранение того или иного структурного элемента возможно на основе контекста, который

позволяет восстановить пропущенный элемент. Когнитивная проблема рассматривается автором как проблема взаимодействия языка и синтаксиса в процессе порождения предложений. Интерпретация предложений происходит в системе наших знаний. Но для описания некоторого фрагмента интерпретируемого знания может быть образовано множество предложений, эквивалентных в том смысле, что им соответствует одна и та же интерпретация.

Работы, посвященные когнитивным процессам, происходящим при мысленной генерации предложений, охватывают многие проблемы [6]: исследуется акцентуация различных аспектов события и её отражение в синтаксисе предложений, рассматривается явление топикализации обстоятельств времени, места и образа действия, что означает перемещения обстоятельств в начало предложения. Это явление связывается с когнитивной выделенностью объектов.

В целом предполагается, что при конструировании предложения реализуется принцип гештальтности [7]: в мышлении устанавливаются различные связи между отдельными фрагментами объективной действительности и на основе этих связей фрагменты соотносятся друг с другом в сфере предложения.

В статье [8] автор рассматривает когнитивно-функциональные основания отношений порядка между компонентами сочинительных конструкций в русских простых предложениях и предикативных частях сложных предложений. Автор предлагает считать, что эти отношения подчиняются когнитивным правилам линеаризации при построении предложений. Рассматриваются два типа правил: когнитивные и текстовые.

При осмыслении явлений внешнего мира человек строит когнитивные модели [7], выделены 4 типа моделей – образно-схематическая, метонимическая, метафорическая и пропозициональная. Пропозициональная модель отражает структуру концепта. Из простых пропозиций образуются сложные при использовании различных механизмов объединения. Этот тезис лежит в основе работ Давыдовой [9, 10] (по когнитивному моделированию сложносочиненного предложения). Сложносочиненное предложение представляется как объединение пропозиций с помощью связок. В когнитивной модели базовые операции объединения пропозиций связаны с логическими операциями. Различные смысловые оттенки связей реализуются через выбор союзов. Рассмотрены все возможные типы отношений соединения: присоединение, сопоставление и противопоставление, которые могут включать причинные уступительные и некоторые другие отношения.

В качестве когнитивных механизмов порождения сложносочиненных предложений рассматривается: фокусирование внимания на главном событии (фигуре), выделение фигуры и фона и выбор концептуальной точки зрения.

2. КОГНИТИВНЫЙ ПОДХОД К КЛАССИФИКАЦИИ ЭЛЛИПСИСОВ

Настоящая работа посвящена рассмотрению когнитивных оснований эллипсиса и формулировке правил генерации некоторого типа эллиптических предложений. Правила генерации эллиптических предложений были сформулированы на основе анализа большого корпуса текстов планиметрических задач, однако они применимы и для аналогичных конструкций в художественной и научной литературе.

Часто явление эллипсиса связывают с желанием избежать избыточности в речи или тексте. Однако понятие избыточности трудно формализуемо. Избыточность текста складывается из совокупности разных типов избыточности (таких, как семантическая, лексическая, грамматическая, графическая и пр.). В зависимости от типа текста его общая избыточность может складываться за счёт разных составляющих. Автор работы [11] связывает исследование избыточности с изучением функциональной нагрузки различных элементов и структур текста. Ею проведено оригинальное исследование, когда из текста производилось удаление различных элементов и затем наблюдались реакции носителей языка при восстановлении пропусков.

Общепринятой типологии эллиптических конструкций на данный момент не существует [12]. В конкретных примерах между эллипсисом и сходными явлениями неполноты, имплицитности, зевгмы может не быть четкой границы, подчеркивают Л.Ю. Максимов и В.В. Бабайцева [13].

В результате в одном и том же предложении эллипсис относят к разным типам. Так, эллипсис в предложении «Один ведерком черпает, другой – шапкой, третий – горстями» относят к зевгме [11], к глагольному эллипсису [12], анафорическому или сочинительному эллипсису [14] и к грамматическому эллипсису [15].

Основание эллипсиса мы связываем с мыслительными процессами образования эллиптических конструкций. Важным является изучение тех когнитивных процедур, которые прямо не наблюдаются, но позволяют выявлять связь между когнитивными процессами, то есть процессами манипулирования знаниями, и

синтаксисом различных языков. Начало этому направлению работ было положено Джекендоффом Р. [16, 17, 18].

С нашей точки зрения классификация эллипсисов может быть произведена на основе знаний, которые участвуют в процессе их образования:

- однозначно восстанавливаемый эллипсис в силу того, что опускаемая часть уже есть в тексте в том же или ближайшем предложении;
- однозначное восстановление на основе общих знаний, традиций, принятых соглашений, подразумеваемом контексте;
- многозначное восстановление – по контексту может быть несколько возможностей восстановления;
- невозможность восстановления.

Приведем некоторые примеры из дневников драматурга Е. Шварца [19].

Восстановление однозначное.

Добриков уже не живет у нас, но я прорезаю окна в домах по его способу. По его же вырезаю я из бумаги сани с полозьями.

Я становился все более одесситом, как недавно майкопцем – в Майкопе и рязанским мальчиком – в Рязани.

Варю нельзя было попросить сыграть Четырнадцатую сонату Бетховена, а Наташу – Седьмую.

Иногда Милочка здоровалась со мной приветливо, иной раз невнимательно, как бы думая о другом, то дружески, а вдруг – как с малознакомым.

Я словно заново научился ходить и смотреть, а главное – говорить.

Мы купили и учебники, и тетради, и деревянный пенал, верхняя крышка которого отодвигалась с пистком, и, чтобы носить все это в училище, – ранец.

Восстановление многозначное синтаксически разными фрагментами, но семантически эквивалентными и зависимыми от контекста

В домах живут. (Люди – общезначимо, у Шварца, семьи военных, театральные деятели)

Мы отправились с мамой в магазин Мареева покупать учебники. В магазине было *полно*. Каждый приказчик знал, какие учебники нужны данному классу. Мы купили и учебники, и тетради, и деревянный пенал, верхняя крышка которого отодвигалась с пистком, и, чтобы носить все это в училище, – ранец. Можно предположить, что

слово «полно» относится либо к толпе покупателей, либо к изобилию учебников.

Восстановление однозначное на основе традиционных знаний.

Был вчера в Художественном. Достал билеты в Художественный.

Я живу на Суворовском

Невозможно восстановить – возможно из-за противоречия логического или незнания контекста.

Такой пример дает Умберто Эко в [20]: Charles makes love with his wife twice a week, so does John.

Далее мы будем рассматривать только два типа эллипсисов, которые однозначно разрешимы на основе анализа предложения, содержащего эллипсис и, возможно, соседних предложений. Мы будем называть их «глагольный эллипсис» (the VPE) и именной эллипсис, структура которых основана на именных, глагольных и предложных группах (Noun Phrases, Verb Phrases and Prepositional Phrases). В литературе эти типы эллипсисов известны: первый как эллипсис глагольной группы (исключение глагола, как изолированного, так и в составе глагольных групп и целых клауз), второй как эллипсис составляющих именной группы с «сохранением представителя».

Когнитивный подход к выделенным типам эллипсисов (которые восстановимы однозначно, так как в тексте выражены пропущенные фрагменты) заключается в понимании когнитивных процессов, связанных с генерацией этих эллиптических конструкций.

Когнитивные аспекты эллипсисов:

- Полное предложение (клауза) описывает некоторую когнитивную воображаемую ситуацию, в частном случае геометрическую ситуацию, выраженную в тексте геометрической задачи [21];
- Полное предложение (клауза) мысленно трансформируется в неполное эллиптическое предложение;
- Эта трансформация основана на когнитивных операциях, мысленно выполняемых **по определенным алгоритмам** в процессе генерации эллиптической части предложения.

Как результат этих когнитивных оснований появились следующие правила генерации эллиптических предложений [21].

- **Правило 1.** Если в предложении введено обозначение некоторого объекта, то в дальнейшей части изложения может употребляться только введенное обозначение без именованного объекта;

- **Правило 2.** Если в предложении введено обозначение некоторой геометрической фигуры, то в дальнейшем изложении это обозначение может употребляться без именованя фигуры;
- **Правило 3.** Если имеется в виду одно и то же действие над (с) несколькими объектами, то после описания этого действия над первым объектом в предложении, далее это действие над (с) другими объектами может быть описано без копирования именованя этого действия (пропускается глагол);
- **Правило 4.** Объект в предложении может быть выражен в составе именной группы (Noun Phrase), тогда при упоминании в предложении того же самого объекта возможен пропуск не только его наименования, но и общих повторяющихся фрагментов именной группы, характеризующих объект;
- **Правило 5.** Глагол может входить в предложение в составе глагольной группы (Verb Phrase), тогда после полного описания действия над (с) первым объектом возможно в последующих описаниях того же действия над (с) другими объектами можно опускать не только глагол, но и повторяющиеся аргументы глагола (или их составляющие).

Так как аргументы глагола могут включать именные группы, то пропуск объекта и составляющих его именной группы может быть реализован при элиминировании глагола и его аргументов. Это явление определяется в [15] как эллипсис существительных при глагольных актантах с «сохранением представителя».

Рассмотрим некоторые примеры.

Задача 1. *Дана трапеция $ABCD$ с основанием AD . Биссектрисы внешних углов при вершинах A и B пересекаются в точке P , а при вершинах C и D – в точке Q .*

Во втором предложении, часть именной группы «биссектрисы внешних углов» и глагол «пересекаются» опущены.

Задача 2. *В прямоугольном треугольнике ABC , высота CK проведена из вершины прямого угла C , а в треугольнике ACK – биссектриса CE .*

В этом предложении опущены глагол и часть глагольной группы «проведена из вершины прямого угла C »

Задача 3. *Обозначим основания перпендикуляров, опущенных из точки A на заданные прямые через M и N , а основания перпендикуляров, опущенных из точки B – через K и L .*

Действие «опустить перпендикуляр» имеет два аргумента: «из точки» и «на заданные прямые». Когда перпендикуляр опускается из

второй точки В, глагол и аргумент «на заданные прямые», будучи теми же самыми, опускаются.

Задача 4. *Другим источником были донесения заводских представителей в войсковых частях. Третьим – официальные сообщения командования авиационных частей и соединений. (А.И.Шахурин. Крылья победы.1984. стр.214).*

Во втором предложении пропущено главное слово «источником» в именной группе «третьим источником», которая является косвенным дополнением глагола «были» и сам глагол, но именная группа подлежащего полностью изменилась – «официальные сообщения командования авиационных частей и соединений», эта именная группа полностью заменяет именную группу «донесения заводских представителей в войсковых частях».

Задача 5. *Внутри квадрата A_1, A_2, A_3, A_4 взята точка P . Из вершины A_1 проведена прямая, перпендикулярная к прямой A_2P , из вершины A_2 – к прямой A_3P , из вершины A_3 – к прямой A_4P и из вершины A_4 – к прямой A_1P . Докажите, что все 4 перпендикуляра пересекаются в одной точке.*

Во втором предложении 3 раза опущены глагол и составляющая его прямого дополнения «проведена прямая, перпендикулярная к прямой».

Задача 6. *В результате первой операции он должен был добиться уничтожения врага к западу от реки и выйти к ней, в результате второй – создать плацдармы, а затем провести вторжение в глубь Германии. (У. Черчилль. Вторая мировая война Том 6. Часть 2).*

В этом предложении пропущена составляющая глагола «должен был».

На умытом лице он чувствовал свежесть, на сильном теле – непривычную после похода чистоту, во всех отдохнувших членах – спокойствие и силу (Лев Толстой, Казаки, стр 391).

В этом предложении опущен глагол и подлежащее «он».

Но когнитивные правила выполнимы только с соблюдением некоторых синтаксических правил, назовем их синтаксическими аспектами эллипсисов:

- каждое слово (словосочетание) может иметь подчиняющее (стержневое – *pivotal, main, governing*) слово и притом только одно; (обратное неверно); (без этого правила эллипсис был бы в принципе невозможен);
- подчиняющее слово (например, вершина именной или глагольной группы) или словосочетание может иметь несколько аналогичных

по грамматической форме и по значению управляемых слов (subordinate) слов или словосочетаний;

- слова, присутствующие в эллиптических фрагментах предложений, сохраняют грамматические функции и формы, свойственные им в соответствующих полных предложениях [22];
- реконструкция глагольных эллипсисов не зависит от того сколько раз пропущен глагол, и от того, сколько у глагола аналогичных подчиненных словосочетаний;
- пропущенные слова или составляющие уже были употреблены в ближайшем или в том же самом предложении [22];
- в эллиптической части предложения опускаются управляющие (pivotal) слова (вершины именной или глагольной группы) отдельно или с некоторыми составляющими, но зависимые от них слова остаются [22].

Модели управления для каждой лексемы и способы их морфосинтаксического оформления даются в [23].

3. ОСНОВНОЙ ПРИНЦИП РАЗБОРА ПРЕДЛОЖЕНИЙ С ЭЛЛИПСИСАМИ

Разбор предложений с приведенными типами эллипсиса основан на выделении именных, глагольных и предложных групп и определении для каждого слова в эллиптической части его «подчинителя» (стержневого или управляющего слова) в antecedente, то есть в соответствующей полной части предложения без эллипсиса. В глагольном эллипсисе существует только один глагол, с которым в эллиптической части предложения нужно связать **все** потенциальные его аргументы, которые отвечают на определенные вопросы, выполняют те же самые роли в предложении и имеют соответствующие грамматические формы. В случае, если есть несколько вариантов, например слово в именительном падеже и в винительном падеже имеют одну и ту же форму, нужно учесть, что слово в именительном падеже будет требовать появления нового глагола в предложении, если такового нет, то слово подчиняется уже употребленному глаголу как прямое дополнение. Приведем пример реставрации эллипсиса в предложении: «В данный треугольник вписан квадрат **так, что** две его вершины принадлежат основанию треугольника, **а** две другие – боковым сторонам». Разрешение эллипсиса будем основывать на копировании полной глагольной группы и замене аргументов глагола (или их составляющих) на соответствующие составляющие, сходные по роли и по грамматическим признакам, имеющиеся в эллиптической части .

Сравниваем полную глагольную группу со структурой неполной группы: (две другие \emptyset) – (боковым сторонам). Именная группа «боковым сторонам» может заменить косвенное дополнение «основанию треугольника». Однако в именной группе «две другие» пропущено слово, которому подчиняется прилагательное «другие».

Сравним именные группы подлежащего в полном предложении и в эллиптической части: Две его (Что?) вершины и Две другие (Что?) \emptyset . Определяем, что составляющая «две другие» может заменить составляющую «две», так как обе они подчиняются составляющей «его вершины». Окончательно имеем: две (его вершины) принадлежат основанию треугольника, а две другие (его вершины) принадлежат боковым сторонам.

Детально алгоритм еще не разработан, но метод связывания главных слов и их составляющих в полном фрагменте предложения с их зависимыми составляющими в эллиптической части применяется в решателе геометрических задач по их текстам на ЕЯ [24].

Объектами эллипсисов являются не слова и не члены предложения, а составляющие [14, стр 187]. Замена составляющих именных и глагольных групп основана на следующих отношениях эквивалентности между ними:

- Слабая синтаксическая эквивалентность (СлабаяСЭ), когда фрагменты выполняют определенную роль в предложении (например, косвенное или прямое дополнение при глаголе); СлабаяСЭ согласуется с грамматическим тождеством составляющих, то есть тождеством «семантически наполненных грамматических категорий лексем» из [14, стр 166].
- Сильная синтаксическая эквивалентность (СильнаяСЭ), когда фрагменты имеют слабую синтаксическую эквивалентность и совпадают их синтаксические структуры;
- Семантическая эквивалентность, когда эквивалентны когнитивные структуры фрагментов; в геометрических текстах эквивалентны когнитивные модели геометрических ситуаций [21] – геометрические объекты и отношения между ними.

В предложении «здесь каждая точка должна принадлежать некоторой линии, одна пара точек принадлежит дуге сектора, другие точки – радиусам сектора» составляющие «дуге сектора» и «радиусам» находятся в СлабойСЭ и в семантической (геометрической) эквивалентности, так как «дуга сектора» и «радиус» являются линиями. Эквивалентность геометрических отношений может проверяться в когнитивной модели или/и при помощи геометрической онтологии [21, 24].

Задача 7. Задачи на максимум и минимум могут быть решены элементарными средствами, без использования производной. (Одни из

них) **сводятся** (к нахождению наибольшего или наименьшего значения квадратного трехчлена), (другие) – (к исследованию выражения, содержащего тригонометрические функции). Сравниваем копию полного фрагмента всего предложения и эллиптическую часть предложения:

(Одни из них) **сводятся** (к нахождению наибольшего или наименьшего значения квадратного трехчлена), (другие \emptyset) \emptyset (к исследованию выражения, содержащего тригонометрические функции).

Здесь мы имеем СлабуюСЭ между предложными группами косвенных дополнений глагола и СлабуюСЭ между именными группами подлежащего в полном и восстановленном предложении.

4. ЗАКЛЮЧЕНИЕ

Работа посвящена когнитивным основаниям классификации эллипсисов и когнитивным мыслительным операциям формирования предложений с эллипсисами определенного типа (глагольный и именной эллипсис). Уделяется внимание возможному алгоритмическому подходу к разрешению этих типов однозначно восстанавливаемых эллипсисов. В работе вводятся три типа отношений эквивалентности между составляющими предложений в полной и эллиптической части: слабая и сильная синтаксические эквивалентности и семантическая эквивалентность. Формулируемые положения, правила мысленной генерации эллипсисов и алгоритм разрешения эллипсисов иллюстрируются большим количеством примеров. Показывается важность когнитивного направления к разрешению эллипсисов в свете изучения взаимодействия когниции и синтаксиса в человеческом мышлении.

Литература

1. Jurafsky D. A Cognitive Model of Sentence Interpretation: the Construction Grammar Approach. // 1993. TR-93-077. International Computer Science Institute.
2. Zhao, G. A Cognitive Approach to Ellipsis // Theory and Practice in Language Studies. 2016. Vol. 6. No. 2. P. 372-377.
3. Culicover & Jackendoff. Simpler syntax. – Oxford: Oxford University Press, 2005.
4. Evans, V. & Green, M. Cognitive linguistics: An introduction. – Edinburgh: Edinburgh University Press, 2006.
5. Langacker, R.W. Subjectification. Cognitive Linguistics // [Электронный ресурс]. 1990 – URL: <http://dx.doi.org/10.1515/cogl.1990.1.1.5> (дата обращения: 01.09.2021)
6. Фурс Л.А. Когнитивные модели синтаксиса // Вопросы когнитивной лингвистики. 2007. No. 4(013). С. 81-85. 10.

7. Лакофф Дж. Лингвистические гештальты. Новое в зарубежной литературе. – Москва: Прогресс, 1981. Вып.10.
8. Сигал К.Я. Прескрипторные правила линеаризации в когниции и тексте. // Вопросы лингвистики. № 3, 2005.
9. Давыдова Е.Н. Когнитивная модель сложносочиненного предложения (на материале русского языка) // Автореферат на соискание ученой степени к. ф. н. 2006. Тамбов.
10. Давыдова Е.И. Сложносочиненное предложение в когнитивном аспекте. Предложение и слово. // Межвузовский сборник науч. пр. – Саратов: Изд-во Саратовского университета. 2006. С. 358-362.
11. Грудева Е.В. Избыточность текста, редукция, эллипсис (на материале русского языка). // Автореферат на соискание ученой степени к. филологических наук. – СПб. 2008. С. 1-41.
12. Путинцева А., Ковригина Л., Шилин И. Автоматическая классификация эллиптических конструкций в русской спонтанной речи. ceur-ws.org/Vol-2233/paper_16.pdf
13. Бабайцева В.В., Максимов Л.Ю. Современный русский язык. Учеб. для пед. институтов по спец № 2101 «Рус. яз. и лит.» в 3 ч. Ч. 3: Синтаксис. Пунктуация. 2-е изд. – М.: Просвещение, 1987.
14. Падучева Е. В. О семантике синтаксиса. – М.: URSS, 2019.
15. Кобзарева Т. Ю, Елифанов М.Е., Лахути Д.Г. Восстановление грамматических эллипсисов при синтаксическом анализе // Труды 14-ой национальной конференции по искусственному интеллекту с международным участием. Том 1. – Казань: Изд-во РИЦ «ШКОЛА», 2014. С. 108-116.
16. Jackendoff R. Semantic And Cognition. MIT Press, 1983. Cambridge
17. Jackendoff R. Information Is In The Mind Of The Beholder // Linguistic and Philosoph. Vol. 8. No. 1. 1985. P. 23-33.
18. Jackendoff R. Grammar As Evidence For Conceptual Structure // Hall, Bresnan and Miller (eds). Linguistic theory and psychological reality. – Cambridge: MIT Press 1978. P. 201-228.
19. Шварц Е. Московская телефонная книга. – М.: Изд-во АСТ, 2016.
20. Эко У. Роль читателя. Исследование по семиотике текста. – М.: Изд-во АСТ, 2015.
21. Naidenova X.: Cognitive Elements In Forming And Understanding Sentences // Proceedings of the International Workshop on Digital Technologies for Teaching and Learning (DTTL-2021), O.F. Nevzorova, N.V. Loukashevitch, E.K. Lipachev (eds.), <http://ceur-ws.org/vol-2910>, 2021. P. 65-73.
22. Гвоздев А Н Современный русский литературный язык. Часть II Синтаксис. – М.: Учпедгиз, 1958.
23. Mel'cuk I.: Lexical Functions: A Tool For The Description Of Lexical Relations In The Lexicon // L. Wanner (eds.), Lexical functioning, lexicography and NLP. – Amsterdam /Philadelphia: Benjamins, 1996. P. 37-102.
24. Naidenova, X., Kurbatov, S., Ganapol'skii, V.: An analysis of plane task text ellipticity and the possibility of ellipses Reconstructing Based On Cognitive Modelling Geometric Objects And Actions. In: A. Elizarov, N. Loukachevich (eds.), Proceedings of Computational Models in Language and Speech Workshop, CMLS 2018, co-located with the 15th TEL International Conference on Computational and Cognitive Linguistics, TEL-2018, vol. 2, pp. 70-85, the Tatarstan Academy of Sciences, Kazan, Russia. 2018.

КАК АВТОНОМНЫЙ КОМПЬЮТЕРНЫЙ АГЕНТ МОЖЕТ САМОСТОЯТЕЛЬНО ОТКРЫВАТЬ ЗАКОНЫ ПРИРОДЫ*

*Редько В.Г., д.ф.-м.н., гл.н.с.
НИИ системных исследований РАН
e-mail: vgrepko@gmail.com*

1. ВВЕДЕНИЕ

В настоящей работе обсуждаются возможности построения моделей автономных агентов, самостоятельно открывающих закономерности природы.

Сначала мы рассмотрим предпосылки к построению и изучению таких моделей. Остановимся на ряде работ, близких к исследованию когнитивных агентов, познающих внешний мир, и постараемся выделить наиболее важные познавательные свойства, которые можно использовать при построении моделей. Особое внимание будет уделяться вопросам, близким к научному познанию. В частности, будем отталкиваться от известных монографий: В.Ф. Турчин «Феномен науки» [1], И. Кант «Критика чистого разума» [2]. Также будем использовать подход книги В.Г. Редько «Моделирование когнитивной эволюции» [3], в которой проанализированы перспективы исследования эволюции познавательных способностей биологических организмов, той эволюции, в результате которой произошло наше мышление, используемое в научном познании. Причем подчеркнем один из ключевых моментов в развитии научного познания – возникновение математических доказательств в древней Греции. Отметим яркие процессы изучения математики лицеистами академии Платона.

Далее постараемся выделить те когнитивные способности живых организмов, которые целесообразно использовать при построении моделей автономных агентов, познающих закономерности природы.

* Настоящая работа выполнена в рамках государственного задания по проведению фундаментальных научных исследований по теме «Исследование нейроморфных систем обработки больших данных и технологии их изготовления» № FNEF-2022-0003

Затем кратко опишем уже разработанные достаточно простые модели когнитивных агентов, познающих внешний мир и использующих свои познавательные способности: модели агентов, предсказывающих будущие ситуации, агентов, строящих свои собственные модели и цепочки моделей.

После этого построим и проанализируем иллюстративную модель агента, описывающего мысленную работу лицеиста в академии Платона.

2. ПРЕДПОСЫЛКИ ПОСТОРОВАНИЯ МОДЕЛЕЙ АГЕНТОВ, ПОЗНАЮЩИХ ПРИРОДУ

Почему важно строить модели агентов, познающих природу? Рассмотрим работы Имануила Канта и Конрада Лоренца [2, 4, 5]. Кант уделял центральное внимание априорным формам мышления. Он считал, что «чистый разум» априорен, и в силу этого он делал естественный вывод, что наш рассудок в познавательном процессе предписывает свои законы природе [4]:

«... хотя вначале это звучит странно, но тем не менее верно, если я скажу: *рассудок не черпает свои законы (a priori) из природы, а предписывает их ей*».

Возникает вопрос: можем ли мы познавать законы природы?

Конрад Лоренц в статье «Кантовская концепция a priori в свете современной биологии» [5] подчеркнул, что априорные формы познания произошли в процессе биологической эволюции, в результате взаимодействия организмов с внешней средой. Образно говоря, наш рассудок не чисто априорен, а имеет также определенные эмпирические корни.

Рассуждая упрощенно, мы, конечно, можем попытаться представить когнитивную эволюцию следующим образом. В процессе биологической эволюции животные приобретали способности познавать внешний мир. Эти познавательные способности помогали им приспосабливаться к окружающей среде, и приспособленности тех животных, которые приобретали «хорошие» познавательные способности, увеличивались. Животные с повышенными приспособленностями распространялись в популяциях, следовательно, «хорошие» познавательные способности фиксировались эволюцией. Шаг за шагом, эволюционно возникали все более сложные и более эффективные когнитивные способности, что, в конечном итоге, привело к появлению человеческого мышления, человеческой логики.

Но этих простых рассуждений явно недостаточно. Проблема применимости нашего мышления к познанию природы –

фундаментальная гносеологическая проблема, и она должна быть проанализирована настолько глубоко, насколько это возможно. И наиболее надежно попытаться построить теорию эволюционного происхождения мышления с помощью моделей когнитивных агентов, моделируя ключевые способности животных разного эволюционного уровня.

Интересно, что кантовская доктрина априорного подобна платоновской теории идей, согласно которой объекты математики (точки, линии, прямоугольные треугольники, ...) реально существуют, но не как материальные предметы, а как образы, или идеи.

Платоном была основана платоновская Академия в 380-х годах до н. э. близ Афин в местности, названной в честь мифического героя Академа. В академии разрабатывался широкий круг дисциплин: философия, математика, астрономия, естествознание и другие.

Особая роль математики подчёркнута в девизе академии: «Ἀγεωμέτρητος μηδεὶς εἰσίτω» = «Не геометр да не войдёт!». В академии читались лекции, делались доклады, проводились дискуссии, решались математические проблемы. Платон полагал, что именно решение математических задач способствует активному мышлению, тренирует ум человека.

В своих знаменитых «Началах» Евклид (в примерно 300 г. до н. э.), собрал и систематизировал все важнейшие труды по математике, существовавшие в конце IV в. до н. э., и изложил их в том же духе, как это было принято в школе Платона.

Хотя математика существовала и до древней Греции (египетская и вавилонская математика), но именно греки сделали мощный прорыв – ввели математические доказательства. Как подчёркивает Турчин [1]:

«Ни в египетских, ни в вавилонских текстах мы не находим ничего, что хотя бы отдаленно было похоже на математическое доказательство. Понятие о доказательстве ввели греки, и это является их величайшей заслугой».

Теперь охарактеризуем направления, которые можно использовать при построении моделей когнитивных агентов, при моделировании когнитивной эволюции.

В книге «Феномен науки» Турчин вводит концептуальную теорию метасистемных переходов. Кратко суть теории метасистемных переходов сводится к следующему: переход от нижних уровней системной иерархии к верхним происходит путем метасистемных переходов. Каждый метасистемный переход можно рассматривать как объединение ряда подсистем нижнего уровня и появление дополнительного механизма управления объединенными

подсистемами. В результате метасистемного перехода формируется система нового уровня, которая может быть включена как подсистема в следующий метасистемный переход.

Турчин характеризует биологическую эволюцию следующими метасистемными переходами:

- управление положением = движение,
- управление движением = раздражимость (простой рефлекс),
- управление раздражимостью = (сложный) рефлекс,
- управление рефлексами = ассоциации (условный рефлекс),
- управление ассоциациями = человеческое мышление,
- управление человеческим мышлением = культура.

Один из важных метасистемных переходов – переход от первобытного мышления к критическому (от мышления первобытных племен к возникновению формального логического мышления). Критическое мышление отличается от первобытного тем, что возникает оценка мыслительного процесса самим мыслящим субъектом [1]:

«Критическое мышление рассматривает каждое объяснение (языковую модель действительности) наряду с другими, конкурирующими объяснениями (моделями), и оно не удовлетворится, пока не будет показано, чем данное объяснение лучше, чем конкурирующее».

Ещё одно важное направление – анализ эволюции познавательных способностей животных, представленный в книге Л.Г. Воронина «Эволюция высшей нервной деятельности» [6]. При этом выделяются следующие уровни: безусловный рефлекс, привыкание (угасание реакции на биологически нейтральный стимул), классический условный рефлекс, инструментальный условный рефлекс, цепи рефлексов, ..., логика. Причём подчёркивается, что с появлением свойств более высокого уровня, свойства нижних уровней не исчезают, а дополняются новыми когнитивными способностями.

В последнее время проявляется большой интерес к моделям, основанным на теории функциональных систем П.К. Анохина [7]. Например, в работах Е.Е. Витяева проведён детальный анализ теории функциональных систем с целью разработки моделей на её основе и разработаны принципы моделирования иерархических систем управления агентов на основе теории функциональных систем [8, 9].

3. СВОЙСТВА, КОТОРЫЕ ЦЕЛЕСООБРАЗНО ЗАЛОЖИТЬ В МОДЕЛИ АГЕНТОВ, ПОЗНАЮЩИХ ПРИРОДУ

В работе [10] были предложены принципы построения модели агента-физика, который мог бы аналогично И. Ньютону самостоятельно изучать природу. Кратко перечислим способности такого агента, самостоятельно познающего природу. У агента должна быть база знаний, должно быть стремление к получению новых знаний и к обобщению знаний, должна быть любознательность, направляющая агента к постановке вопросов о внешнем мире и решению этих вопросов путем постановки экспериментов. Агенты должны учитывать многочисленные связи между уже имеющимися знаниями. Должен быть коллектив агентов, исследующий внешний мир, и должны быть коммуникации между агентами (если основной агент – аналог И. Ньютона, другие агенты могут быть аналогами Г. Галилея, Р. Гука, Р. Декарта, Н. Коперника, Г.В. Лейбница). Должно быть самосознание агентов, эмоциональная самооценка результатов своей деятельности и стремление агента достигнуть наиболее высоких результатов в коллективе агентов. Агенты должны иметь стремление к получению наиболее ясных, четких и компактных знаний, таких как законы Ньютона или аксиомы Евклида. И, конечно, агенты должны освоить возможности логических выводов, позволяющих получить многочисленные следствия законов и аксиом. Понятно, что путь к построению модели познающего агента такого уровня весьма непрост, но, все же, он просматривается.

Рассмотрим и другие свойства, которые целесообразно заложить в свойства познающего агента.

Важное понятие – внутренняя модель агента, с помощью которой он формирует представления о внешнем мире и использует знания, сложившиеся в этой модели при взаимодействии с внешним миром. Такие модели аналогичны научным теориям. Более того, общая научная картина мира – создаваемая всем международным научным сообществом – также может рассматриваться как совокупность моделей. Используя научные модели, мы можем делать предсказания будущих событий во внешнем мире. Простейшие внутренние модели агентов – модели предсказания событий во внешнем мире, будут охарактеризованы в следующем разделе.

Возможны многоуровневые, иерархические внутренние модели агента. Как сказано выше, целесообразно стремление к построению «аксиоматичных» моделей, в которых многочисленные результаты выводятся из небольшого числа «аксиом».

Совсем недавно была предпринята интересная попытка поиска общих (не обязательно биологических) законов эволюции и обучения в многоуровневых системах [11-13]. При этом рассматриваются процессы эволюции каких угодно сложных систем с использованием методов теоретической физики.

В работе [14] анализировались подходы к формализации процессов озарения у животных и человека. При этом подчёркивалось, что при озарении большую роль играет подсознательная память, с помощью которой внезапно находится решение достаточно долго обдумываемой проблемы, после чего решение переносится в рабочую память, в сознание. В определенном смысле рассматриваемые подсознательные и сознательные процессы сходны с процессами в Системе 1 и Системе 2 в понимании Канемана [15].

4. НАЧАЛЬНЫЕ МОДЕЛИ АГЕНТОВ, ПОЗНАЮЩИХ ВНЕШНИЙ МИР

В работе [16] построена и исследована модель, в которой запоминаются причинно-следственные связи между ситуациями внешней среде.

Предполагалось, что в основном ситуации случайные и чередуются во времени случайно. Но возможны и причинные связи между ситуациями. Считалось, что эпизодически появляются пары последовательно чередующихся ситуаций:

$$S_{1i}(t) \rightarrow S_{2j}(t+1) .$$

где t – дискретное время, $t = 1, 2, \dots$. Эти пары представляют собой причинно-следственные связи мира. Некоторые из ситуаций-следствий S_{2j} (вторые элементы пар) существенны для агентов. В этих ситуациях агенты могут приобретать или терять существенную часть своего ресурса $R(t)$. Эти ситуации S_{2j} были благоприятными, если агент приобретает в них ресурс, и неблагоприятными, если в такой ситуации агент теряет ресурс.

Считалось, что имеется два вида агентов: агенты первого вида и агенты второго вида. Агенты первого вида не имеют чувства причинности, агенты второго вида имеют чувство причинности.

Агент с чувством причинности при обучении запоминает в своей базе знаний причинно-следственные связи между повторяющимися многократно парами ситуаций $S_{1i} \rightarrow S_{2j}$ и после обучения заранее готовится к появлению благоприятных и неблагоприятных ситуаций S_{2j} . Путём компьютерного моделирования было показано, что агенты с чувством причинности явно используют

это чувство. Они приспосабливаются к изменениям во внешней среде. Их ресурс растет быстрее, чем у агентов без чувства причинности. Агенты с большим ресурсом чаще рожают потомков, поэтому в результате эволюции агенты с чувством причинности обычно вытесняют из популяции агентов без чувства причинности.

В ряде разработанных нами моделей агенты имели мотивации к выполнению тех или иных действий (см. [3], глава 5). Например, в работе [17] агенты в процессе поискового поведения имели мотивацию к сохранению направления движения, что позволило им адаптивно менять свое поведение при поиске минимума функции нескольких переменных.

На примере моделирования многозвенного целенаправленного поведения новокаледонских ворон исследовалась модель формирования планов цепочки действий, составленных из отдельных звеньев [18].

В ряде моделей поискового поведения рыб в лабиринтах вводилась уверенность агентов в получаемых знаниях, получаемых предсказаниях будущих ситуаций. Полагалось, что модельные агенты должны находить баланс между уверенностью в своих знаниях и поиском новых знаний, если уверенность в достигнутых знаниях высока [19].

Обобщая свойства агентов в отмеченных биологически инспирированных моделях, можно выделить следующие характерные черты исследованных когнитивных агентов (см. [3], глава 7):

- 1) Оценка *знаний* о каждой из возможных ситуаций.
- 2) Движение к ситуации, знания о которой минимальны. Тем самым *агент может самостоятельно формировать новые текущие цели своего поведения*. Например, рыба в лабиринте может решить двигаться в тот коридор, для которого знания минимальны, т.е. в тот коридор, который она давно не посещала.
- 3) Формирование *предсказаний* о результатах действий. Оценка *уверенности в предсказаниях*, стремление к надежным предсказаниям, уверенность в которых высока.
- 4) Формирование *базы знаний*, характеризующей ситуации, действия, прогнозы результатов действий, оценки расстояний между текущими ситуациями и целевой ситуацией.
- 5) Использование базы знаний при планировании целенаправленного поведения.
- 6) *Прямые и обратные мыслительные процессы* (от исходной ситуации к целевой и обратно).
- 7) Формирование *стереотипа* целенаправленного поведения.

5. ИЛЛЮСТРАТИВНАЯ МОДЕЛЬ ЛИЦЕИСТА АКАДЕМИИ ПЛАТОНА

Вернёмся к академии Платона и построим иллюстративную модель учёбы лицеиста академии.

Будем считать, что имеется когнитивный агент, который функционирует так же, как учащийся в академии лицеист. Будем считать, что агент-лицеист делает следующее:

1) Решает задачи. Под задачами будем подразумевать как конкретные геометрические задачи, так и доказательства утверждений.

2) Слушает лекции.

3) Участвует в математических обсуждениях.

4) Если агент достиг достаточно высокого уровня, то он может сделать доклад и даже прочитать лекцию.

Считаем, что агент в процессе учёбы накапливает математические знания, осваивает методы решения задач. При удачном решении задач у агента формируется уверенность в своих математических способностях. Для простоты будем считать, что все задачи имеют одинаковую сложность. Считаем, что задачи рассматриваются агентом последовательно. Вероятность решения отдельной задачи агентом равна $P(t)$, время t дискретно. Знания о методах решения задач равна $K(t)$, уверенность в своих способностях равна $E(t)$.

У агента имеется память о нерешённых им задачах. Он может запомнить несколько таких задач. При достаточно больших знаниях о методах решения задач у агента может возникнуть озарение: он может неожиданно для себя сообразить, как решить определённую из запомненных им ранее задач.

Рассмотрим динамику введенных величин. Считаем, что при $t = 0$ агент поступил в академию. При этом у него величины $P(t)$, $K(t)$, $E(t)$ малы. Агент посещает лекции, участвует в обсуждениях, что увеличивает его знания $K(t)$. Он начинает пробовать решать задачи. Считаем, что каждый такт времени он пробует одну задачу.

При успешном решении задачи вероятность решения следующих задач возрастает:

$$P(t + 1) = P(t) + \Delta(t) [1 - P(t)], \quad (1)$$

также возрастает уверенность в своих мыслительных способностях $E(t)$. Если решение не находится, то $P(t)$ не меняется. Постепенно вероятность $P(t)$ возрастает. Естественно, что величина Δ ,

характеризующая увеличение вероятности решения задач, зависит от знаний $K(t)$ и уверенности $E(t)$:

$$\Delta(t) = \Delta (K(t), E(t)) . \quad (2)$$

Для иллюстрации процесса обучения агента-лицеиста был проведен простой компьютерный расчёт зависимости $P(t)$ от времени.

Предполагалось, что изменения $P(t)$ определяются выражением (1). Динамика $K(t)$, $E(t)$ и $\Delta(t)$ задавалась модельно в соответствии со следующими простыми выражениями (нормировку величин сразу проводим таким образом, чтобы эти переменные менялись в интервале между 0 и 1).

Знания о методах решения задач растут до определенного предела:

$$K(t) = \text{th}(t/T_k) , \quad (3)$$

где T_k – характерное время существенного обучения.

Считаем, что уверенность агента в своих способностях пропорциональна текущей вероятности решения задач:

$$E(t) = P(t) , \quad (4)$$

а величина $\Delta(t)$ пропорциональна произведению $K(t)$ и $E(t)$:

$$\Delta(t) = \alpha K(t)E(t) . \quad (5)$$

Пример расчета зависимости $P(t)$ приведен на рис.1.

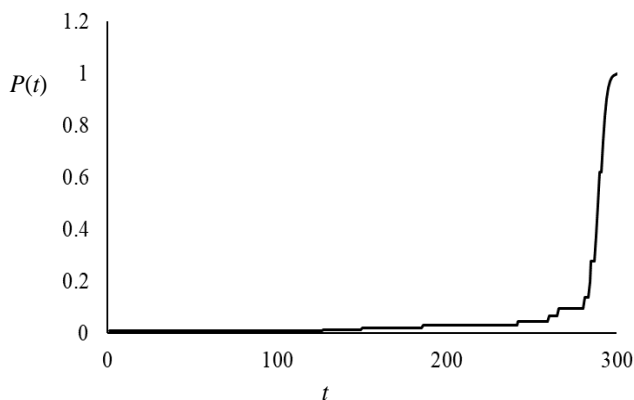


Рис.1. Зависимость вероятности правильного решения задачи $P(t)$ агентом-лицеистом от времени t

При расчёте полагалось, что начальная вероятность решения задачи агентом-лицеистом равна 0.01: $P(0) = 0.01$. Другие параметры составляли: $T_k = 100$, $\alpha = 0.5$.

Видно, что сначала агент-лицеист довольно длительное время обучается и решает задачи довольно редко. Затем вероятность решения задачи растёт и приближается к 1. Явно видны скачки роста $P(t)$ в моменты правильного решения задач.

6. ЗАКЛЮЧЕНИЕ

Итак, проанализированы предпосылки и подходы к построению моделей автономных когнитивных агентов, способных к познанию законов природы. Охарактеризованы начальные модели когнитивных агентов, познающих внешний мир, в том числе модели предсказания закономерных изменений ситуаций и биологически инспирированные модели. Характерные черты исследованных когнитивных агентов в этих биологически инспирированных моделях приведены в конце раздела 4. Дополнительно проанализирована иллюстративная модель агента-лицеиста, моделирующего процессы обучения и решения задач лицеистом в академии Платона.

Литература

1. Турчин В.Ф. Феномен науки. Кибернетический подход к эволюции. – М.: Наука, 1993. (1-е изд.). М.: ЭТС, 2000. (2-е изд.). Turchin V.F. The Phenomenon of Science. A Cybernetic Approach to Human Evolution. – New York: Columbia University Press, 1977. См. также <http://www.refal.ru/turchin/phenomenon>
2. Кант И. Критика чистого разума. Соч. в 6-ти томах. Т. 3. – М.: Мысль, 1964. – С.69–695.
3. Редько В.Г. Моделирование когнитивной эволюции: На пути к теории эволюционного происхождения мышления. Изд. 2, исп. и доп. – М. ЛЕНАНД/URSS, 2018.
4. Кант И. Прологомены ко всякой будущей метафизике, могущей появиться как наука. Соч. в 6-ти томах. Т. 4, часть 1. – М.: Мысль, 1965. – С.67–210.
5. Lorenz K. Kant's doctrine of the a priori in the light of contemporary biology (1941) // Learning, Development and Culture: Essays in Evolutionary Epistemology (Ed. Plotkin H.). – New York: Wiley, 1982. – pp. 121–143. См. также перевод: Лоренц К. Кантовская концепция a priori в свете современной биологии // В книге: Эволюция. Язык. Познание (Под ред. Меркулова И.П.). – М.: Языки русской культуры, 2000. – С. 15–41.
6. Воронин Л.Г. Эволюция высшей нервной деятельности (очерки). – М.: Наука, 1977.
7. Анохин П.К. Системные механизмы высшей нервной деятельности. – М.: Наука, 1979.

8. Витяев Е.Е. Принципы работы мозга, содержащиеся в теории функциональных систем П.К. Анохина и теории эмоций П.В. Симонова // *Нейроинформатика (электрон. журнал)*. – 2008. – Т. 3. – № 1. – С. 25–78. URL: <http://www.niisi.ru/iont/ni/Journal/V3/N1/Vityaev.pdf>
9. Витяев Е.Е. Логика работы мозга // *Подходы к моделированию мышления (Под ред. Редько В.Г.)*. – М.: УРСС, 2014. – С.119–151.
10. Red'ko V.G. Principles of functioning of autonomous agent-physicist // *Biologically Inspired Cognitive Architectures 2012. Proceedings of the Third Annual Meeting of the BICA Society (A. Chella, R.Pirrone, R. Sorbello, K.R. Johanssdottir, Eds)*. Springer: Heidelberg, – New York, Dordrecht, London. – PP.265–266.
11. Vanchurin V., Wolf Yu.I., Katsnelson M.I., Koonin E.V. Toward a theory of evolution as multilevel learning // *Proc. Natl. Acad. Sci. U.S.A.* – 2022. Vol.119. – No.6. – e2120037119. DOI: 10.1073/pnas.2120037119.
12. Vanchurin V., Wolf Yu.I., Koonin E.V., Katsnelson M.I. Thermodynamics of evolution and the origin of life // *Proc. Natl. Acad. Sci. U.S.A.* – Vol.119. – No.6. – e2120042119 DOI: 10.1073/pnas.2120042119.
13. Кунин Е., Демина Н. Жизнь как многоуровневое обучение // *Троицкий вариант – Наука*. 2022. – №4. – С.1,3,5.
14. Редько В.Г., Самсонович А.В. Формализация процессов озарения // *Девятнадцатая Национальная конференция по искусственному интеллекту с международным участием КИИ-2021. Труды конференции / под ред. В.В. Борисова, Б.А. Кобринского*. – Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2021. – С.61–86.
15. Канеман Д. *Думай медленно... решай быстро*. — М.: АСТ, 2013.
16. Редько В.Г. Модель чувства причинности // *Труды НИИСИ РАН*. – 2020. Т.10. – №2. – С.34–38.
17. Непомнящих В.А., Редько В.Г. Метод формирования поискового поведения // *Искусственный интеллект и принятие решений*. – 2011. – №1. – С.49–54.
18. Red'ko V.G., Burtsev M.S. Modeling of mechanism of plan formation by New Caledonian crows // *Procedia Computer Science*. – 2016. – Vol.88. – pp.403–408. DOI: 10.1016/j.procs.2016.07.456. URL: <http://www.sciencedirect.com/science/article/pii/S1877050916317124>
19. Red'ko V.G., Nepomnyashchikh V.A., Osipova E.A. Models of fish exploratory behavior in mazes // *Biologically Inspired Cognitive Architectures*. – 2015. – Vol.13. – pp.9–16. DOI: 10.1016/j.bica.2015.05.002.

МОДЕЛЬ ПОЗНАНИЯ ЗАКОНОМЕРНОСТЕЙ АВТОНОМНЫМ АГЕНТОМ*

*Редько В.Г., д.ф.-м.н., гл.н.с.,
e-mail: vcredko@gmail.com*

*Сохова З.Б., к.т.н.
e-mail: zarema.sokhova@gmail.com
НИИ системных исследований РАН*

1. ВВЕДЕНИЕ

Можно очертить общий подход к исследованию эволюции познавательных способностей биологических организмов, той эволюции, которая привела к способности научного познания природы человеком. Общая схема в этом подходе моделирования постепенного усовершенствования процессов познания может включать в себя исследование с помощью автономных когнитивных агентов следующих процессов: 1) использование чувства причинности, простые предсказания будущих ситуаций, 2) обобщение понятий, 3) обобщение процессов предсказания, 4) введение формальных теорий, используемых в процессах познания.

В настоящей работе проведен определённый начальный этап очерченного общего исследования. Построена и исследована модель познания автономным агентом закономерностей во внешней среде. Эта модель развивает ранее разработанные модели предсказания [1, 2], в которых были изучены сравнительно простые модели причинности, в которых агент предсказывает и использует причинно-следственные связи.

2. ОПИСАНИЕ МОДЕЛИ ПОЗНАНИЯ ЗАКОНОМЕРНОСТЕЙ АВТОНОМНЫМ АГЕНТОМ

В настоящей модели рассматриваем популяцию автономных агентов. Будем предполагать, что агенты могут определять уровень освещённости в окружающей среде. Уровень освещённости меняется

* Настоящая работа выполнена в рамках государственного задания по проведению фундаментальных научных исследований по теме «Исследование нейроморфных систем обработки больших данных и технологии их изготовления» № FNEF-2022-0003

периодически со временем. Например, мы можем рассматривать модельных “рыб”, которые определяют уровень освещённости в течение суток в летнее время и питаются преимущественно на утренней заре.

Уровень освещённости в течение суток аппроксимируем следующим образом. Считаем, что ночью/днем уровень освещённости равен F_0/F_1 ($F_0 < F_1$), на рассвете уровень освещённости линейно возрастает от F_0 до F_1 , на закате уровень освещённости уменьшается от F_1 до F_0 . В следующие сутки динамика освещённости периодически повторяется. Зависимость освещённости от времени $F(t)$ представлена на рис.1. Полагаем, что длительности рассвета, дня, заката и ночи равны t_1, t_2, t_3, t_4 , соответственно. Для определенности в модели полагаем $t_1 = t_3, t_2 = 2t_4, t_4 = 10t_1$. Время считаем дискретным, полагаем, что такт времени равен $0.1t_1$. Каждый такт времени агент может выполнить одно из двух действий: питаться A_1 или отдыхать A_2 . Для удобства моделирования полагаем, что $A_1 = 1, A_2 = 0$. Считаем, что, когда агент питается на рассвете, он каждый такт времени получает большой ресурс (подкрепление), равный r_{max} . В остальные периоды времени при питании агент каждый такт теряет небольшой ресурс, равный r_{mean} . Когда агент отдыхает, он тоже теряет небольшой ресурс, равный r_{min} . Считаем, что $r_{min} \ll r_{mean} < r_{max}$.

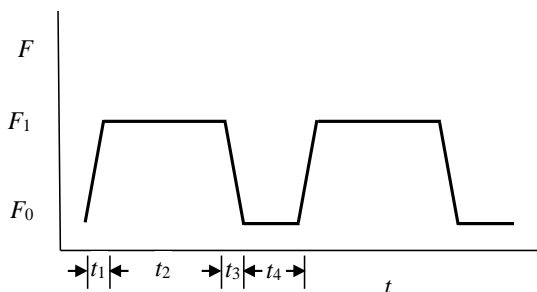


Рис.1. Зависимость освещённости от времени $F(t)$

Естественно, что разумная тактика поведения агента такова: питаться на рассвете, отдыхать в другое время суток. В настоящей модели исследуется, как могла бы сформироваться такая тактика поведения.

Считаем, что агенты имеют систему управления, с помощью которой они выбирают свои действия. Предполагаем, что система управления агента оптимизируется путем обучения с подкреплением [3] и посредством дарвиновской эволюции.

Ситуация $\mathbf{S}(t)$, в которой находится агент, определяется двумя величинами $F(t)$ и $A(t)$, $\mathbf{S}(t) = \{F(t), A(t)\}$, где $A(t)$ – действие агента.

Система управления агента. Система управления агента предназначена для максимизации функцию полезности $U(t)$ [3]:

$$U(t) = \sum_{j=0}^{\infty} \gamma^j r(t+j), \quad t = 1, 2, \dots \quad (1)$$

где $r(t)$ – текущее подкрепление, γ – дисконтный фактор ($0 < \gamma < 1$). Величина $U(t)$ – это субъективная оценка агентом будущей суммарной награды.

Система управления агента состоит из двух нейронных сетей (НС): Модель и Критик. НС Модель предсказывает динамику освещения $F(t)$. НС Критик оценивает функцию полезности $U(t)$ для текущей ситуации $\mathbf{S}(t)$ и предсказываемых ситуаций для двух возможных действий агента в следующий такт времени.

Работа и обучение системы управления. На вход Модели подается m предыдущих значений освещения $F(t-m+1), \dots, F(t)$, на выходе формируется прогноз освещения F в следующий такт времени $F^{pr}(t+1)$. Модель представляет собой двухслойную НС, работа которой описывается формулами:

$$\mathbf{x}^M = \{F(t-m+1), \dots, F(t)\}, \mathbf{y}^M_j = \text{th}(\sum_i w^M_{ij} x^M_i), F^{pr}(t+1) = \sum_j v^M_j y^M_j, \quad (2)$$

где \mathbf{x}^M – входной вектор, \mathbf{y}^M – вектор выходов нейронов скрытого слоя, w^M_{ij} и v^M_j – веса синапсов данной НС.

Критик предназначен для оценки качества ситуаций $V(\mathbf{S})$, а именно, оценки функции полезности $U(t)$ для агента, находящегося в ситуации \mathbf{S} . Критик представляет собой двухслойную НС, работа которой описывается формулами:

$$\mathbf{x}^C = \mathbf{S}(t) = \{F(t), A(t)\}, \mathbf{y}^C_j = \text{th}(\sum_i w^C_{ij} x^C_i), V(t) = V(\mathbf{S}(t)) = \sum_j v^C_j y^C_j, \quad (3)$$

где \mathbf{x}^C – входной вектор, \mathbf{y}^C – вектор выходов нейронов скрытого слоя, w^C_{ij} и v^C_j – веса синапсов НС.

При работе системы управления агента каждый такт времени t выполняются следующие операции:

- 1) Модель предсказывает освещённость в следующий такт времени $F^{pr}(t+1)$.
- 2) Критик оценивает величину V для текущей ситуации $V(t) = V(\mathbf{S}(t))$ и для предсказываемых ситуаций для двух возможных действий $V^{pr}_A(t+1) = V(\mathbf{S}^{pr}_A(t+1))$, где $\mathbf{S}^{pr}_A(t+1) = \{F^{pr}(t+1), A(t+1)\}$, $A(t+1) = 0$ либо $A(t+1) = 1$.

- 3) Применяется ε -жадное правило [3]: с вероятностью $1 - \varepsilon$ выбирается действие, соответствующее максимальному значению $V^{pr}_A(t+1)$, в противном случае выбирается альтернативное действие ($0 < \varepsilon \ll 1$). Выбор действия есть выбор величины $A(t+1)$: отдыхать $A(t+1) = 0$, либо питаться $A(t+1) = 1$.
- 4) Выбранное действие $A(t+1)$ выполняется. Происходит переход к такту времени $t+1$. Подсчитывается подкрепление $r(t+1)$ так, как изложено выше. Наблюдаемое значение $F(t+1)$ сравнивается с предсказанием $F^{pr}(t+1)$. Веса НС Модели подстраиваются так, чтобы минимизировать ошибку предсказания методом обратного распространения ошибки [4]. Скорость обучения Модели равна $\alpha_M > 0$.
- 5) Критик подсчитывает $V(t+1) = V(\mathbf{S}(t+1))$; $\mathbf{S}(t+1) = \{F(t+1), A(t+1)\}$. Рассчитывается ошибка временной разности [3]:

$$\delta(t) = r(t) + \gamma V(t+1) - V(t). \quad (4)$$
- 6) Веса НС Критика подстраиваются так, чтобы минимизировать величину $\delta(t)$, это обучение осуществляется градиентным методом, аналогично методу обратного распространения ошибки. Скорость обучения Критика равна $\alpha_C > 0$.

Схема эволюции. Рассматривается эволюционирующая популяция, состоящая из n агентов. Агенты имеют систему управления, которая была описана выше. Каждый агент имеет ресурс $R(t)$, который изменяется в соответствии с подкреплениями: $R(t+1) = R(t) + r(t+1)$, где подкрепление $r(t+1)$ определяется так, как изложено выше. Эволюция происходит в течение ряда поколений, $n_g = 1, 2, \dots$. Каждое поколение n_g функционирует T_g тактов времени (T_g – продолжительность жизни одного поколения). В начале каждого поколения ресурс R каждого агента равен нулю.

Каждый агент имеет два набора весов синапсов нейронных сетей: \mathbf{G} и \mathbf{W} . Набор \mathbf{G} представляет собой начальные веса синапсов НС, получаемые агентом в момент его рождения от агента-родителя. Этот набор \mathbf{G} есть генотип агента, который не меняется в течение его жизни. Набор \mathbf{W} – текущие веса синапсов НС, которые подстраиваются в течение жизни агента путем обучения, описанного выше. В момент рождения агента $\mathbf{W} = \mathbf{G}$. Потомки агента наследуют генотип \mathbf{G} (с небольшими мутациями). Так как наследуется именно генотип \mathbf{G} , а не изменяемые в течение жизни веса \mathbf{W} , то эволюция носит дарвиновский характер.

Процесс размножения происходит следующим образом. В конце каждого поколения определяется агент, имеющий максимальный ресурс $R_{max}(n_g)$ (лучший агент поколения n_g). Этот лучший агент

порождает n потомков, которые составляют новое (n_g+1) -е поколение. Генотипы потомков \mathbf{G} отличаются от генотипа родителя небольшими мутациями. Более конкретно, предполагается, что в начале нового (n_g+1) -го поколения для каждого агента его генотип формируется следующим образом $G_i(n_g+1) = G_{best,i}(n_g) + rand_i$, где $G_{best,i}(n_g)$ – компоненты генотипа лучшего агента предыдущего n_g -го поколения, $rand_i$ – равномерно распределенные случайные величины в интервале $[-0.5P_{mut}, 0.5P_{mut}]$, P_{mut} – интенсивность мутаций, i – индекс веса синапса.

3. РЕЗУЛЬТАТЫ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ

Было проведено компьютерное моделирование процессов обучения для двух вариантов модели: 1) нейросетевая модель, изложенная выше, 2) упрощённая модель, в которой использовался хорошо известный в обучении с подкреплением метод SARSA [3] для обучения агента; эта модель будет изложена ниже.

Для обоих случаев полагалось, что, когда агент питается на рассвете, он получает ресурс $r_{max} = 1.0$, когда он делает попытку питания в другие моменты времени, он теряет ресурс $r_{mean} = 0.1$, при отдыхе агент теряет совсем малый ресурс $r_{min} = 0.001$. Длительности рассвета, дня, заката и ночи равны $t_1 = 10$, $t_2 = 200$, $t_3 = 10$, $t_4 = 100$ тактов времени, соответственно. Продолжительность одного такта равна 0.01. Освещённость ночью равна 1, освещённость днем равна 11.

Изложим сначала результаты для нейросетевой модели.

Параметры нейросетевого метода были следующие: количество агентов $N_A = 10$; параметр ϵ -жадного правила $\epsilon = 0.01$ или 0.1; дисконтный фактор $\gamma = 0.5$ или 0.9; скорость обучения Критика $\alpha_C = 0.01$ или 0.1; скорость обучения Модели $\alpha_M = 0.01$; количество нейронов входного слоя Модели $N_{mi} = 320$; число нейронов скрытого слоя Модели $N_{mj} = 20$; число нейронов входного слоя Критика $N_{ci} = 2$; число нейронов скрытого слоя Критика $N_{cj} = 10$; число итераций обучения для Модели $N_{LTM} = 100$; число итераций обучения для Критика $N_{LTC} = 1$; число поколений $n_g = 1000$; продолжительность жизни поколения $T_g = 1000$ тактов времени; интенсивность мутаций $P_{mut} = 0.1$.

Были проанализированы следующие варианты:

- случай L (чистое обучение); в этом случае рассматривалась популяция агентов, которая обучалась путем минимизации ошибки временной разности (4);

- случай E (чистая эволюция), т.е. эволюционирующая популяция агентов без обучения;
- случай LE (обучение + эволюция), т.е. полная модель, изложенная выше.

Было проведено сравнение ресурса, приобретаемого агентами за 1000 временных тактов для этих трех способов адаптации. Для всех случаев бралось $T_g = 1000$ (T_g – продолжительность поколения) и регистрировалось максимальное значение ресурса в популяции $R_{max}(n_g)$ в конце каждого поколения.

Графики $R_{max}(n_g)$, представленные на рис.2 (а, б), показывают, что обучение, объединенное с эволюцией (случай LE), обеспечивает более эффективный рост R_{max} , чем обучение или эволюция отдельно (случаи L и E). Наблюдается эффект, аналогичный известному эффекту Болдуина [5, 6]. Отметим, что в случае чистого обучения L, ресурс агентов быстро возрастал (этот быстрый рост на рис.2 незаметен, так как рассматривалось большое число поколений).

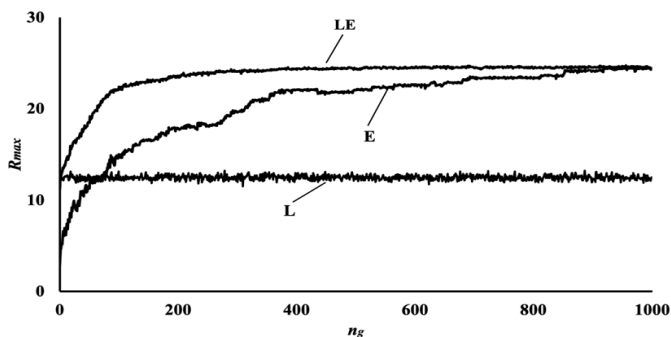
Также было исследовано влияние параметра ε на динамику ресурса популяций агентов. Видно, что при $\varepsilon = 0.01$ ресурс агентов популяции получается выше для случая LE (при $\varepsilon = 0.01$ достигается ресурс ≈ 28 усл. ед., при $\varepsilon = 0.1$ максимальный ресурс равен ≈ 24 усл. ед.). Такой результат вполне очевиден. При малом ε агенты, имеющие хороший генотип и научившиеся питаться на рассвете, выполняют меньше случайных действий, которые уменьшают ресурс при выполнении действия «питание» не на рассвете. Параметр ε практически не оказал влияния на линию E.

Отдельно отметим, что значение ресурса в чистом обучении (линия L) при $\varepsilon = 0.1$ получается выше, чем при $\varepsilon = 0.01$. Причина в том, что при $\varepsilon = 0.1$ агенты чаще случайно попадают на правильные действия «питание», и получают большую награду $g_{max} = 1.0$.

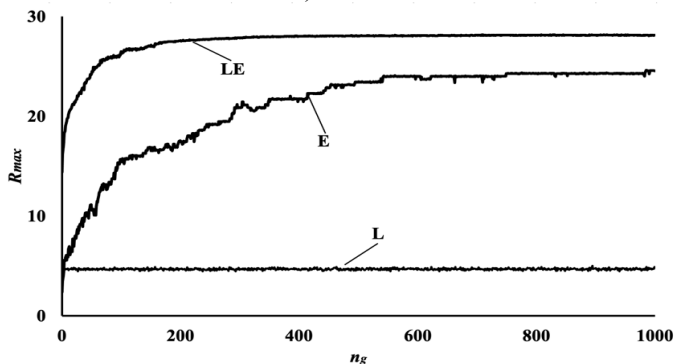
Теперь кратко изложим второй вариант модели и сравним результаты компьютерного моделирования для упрощенной модели с результатами нейросетевой модели.

Упрощенная модель. В этом варианте модели считалось, что агент упрощенно определяет имеющуюся ситуацию. Он рассматривает две возможные ситуации: 1) обычная ситуация $s_1 = 0$, при которой увеличение освещенности за один такт времени не превышает некоторый порог F_{prth} , 2) необычная ситуация $s_2 = 1$, при которой увеличение освещенности за один такт времени превышает некоторый порог F_{prth} . Ситуация $s_2 = 1$ соответствует рассвету, ситуация $s_1 = 0$ соответствует другим моментам времени суток. Имеется два действия, которые может выполнять агент: 1) $A_1 = 0$, что соответствует отдыху агента и 2) $A_2 = 1$, что соответствует питанию агента (поиску пищи и собственно питанию). Задача агента состоит в следующем: выполнять

действие $A_1 = 0$, когда он находится в ситуации s_1 и выполнять действие $A_2 = 1$ в ситуации s_2 , то есть агент должен питаться на рассвете и отдыхать в остальное время.



а) $\epsilon = 0.1$



б) $\epsilon = 0.01$

Рис.2. Зависимости $R_{max}(n_g)$. Кривая LE соответствует случаю обучения, объединенного с эволюцией, кривая E – случаю чистой эволюции, кривая L – случаю чистого обучения. Кривые усреднены по 100 различным расчетам

Для того чтобы обеспечить такое поведение, агент обучался методом SARSA [3].

Параметры метода SARSA: $\epsilon = 0.1$; $\gamma = 0.7$; скорость обучения $\alpha_C = 0.1$. Порог для сравнения скорости роста освещенности $F_{prth} = 0.5$. Освещённость ночью равна 1, освещённость днем равна 11.

Для сравнения были проведены расчеты и для нейросетевой модели со следующими параметрами: $N_A = 1$; $\epsilon = 0.1$; $\gamma = 0.7$; $\alpha_C = 0.1$; $\alpha_M = 0.01$; $N_{mi} = 320$; $N_{mj} = 20$; $N_{ci} = 2$; $N_{cj} = 10$; $N_{LTM} = 100$; $N_{LTC} = 1$;

число поколений $n_g = 1$; продолжительность жизни поколения $T_g = 10^5$; интенсивность мутаций $P_{mut} = 0.0$ (эволюция выключена).

На рис.3 показана зависимость суммарного ресурса R от времени для упрощенной (SARSA) и нейросетевой модели.

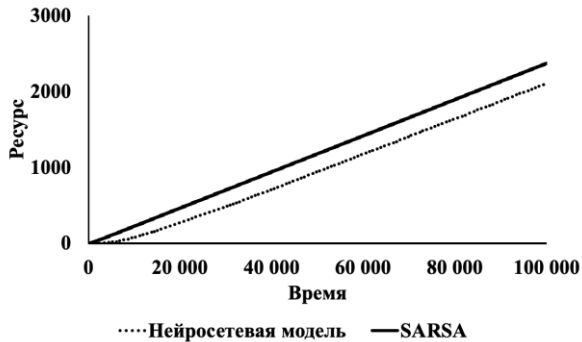


Рис.3. Зависимость суммарного ресурса агента от времени

Видно, что для этих двух вариантов модели результаты близки друг к другу, что дополнительно подтверждает правильность расчетов по нейросетевой модели.

4. ЗАКЛЮЧЕНИЕ

Таким образом, построена и изучена модель познания закономерностей автономным агентом. Продемонстрировано, что взаимодействие между обучением и эволюционной оптимизацией обеспечивает более эффективную стратегию поведения агентов, чем обучение или эволюция отдельно.

Литература

1. Редько В.Г. Моделирование чувства причинности. Первые результаты // Труды НИИСИ РАН. – 2019. – Т.9. – №1. – С.66–68.
2. Редько В.Г. Модель чувства причинности // Труды НИИСИ РАН. – 2020. – Т.10. – №2. – С.34–38.
3. Sutton R.S., Barto A.G. Reinforcement Learning: An Introduction. – MIT Press, 1998.
4. Rumelhart D.E., Hinton G.E., Williams R.G. Learning representation by back-propagating error // Nature. – 1986. – Vol.323. – №6088. – P.533-536.
5. Baldwin J.M. A new factor in evolution // American Naturalist. – 1896. – Vol.30. – P.441-451.
6. Turney P., Whitley D., Anderson R. (Eds.). Evolution, Learning, and Instinct: 100 Years of the Baldwin Effect // Special Issue of Evolutionary Computation on the Baldwin Effect. – 1996. – Vol.4. – №3.

СЕКЦИЯ 6

**ГИБРИДНЫЕ
ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ**

УДК 004.415.2

**ИССЛЕДОВАНИЕ РАЗВИТИЯ СИСТЕМ
ОБЪЯСНИТЕЛЬНОГО ИСКУССТВЕННОГО
ИНТЕЛЛЕКТА***

*Аверкин А.Н., к.ф.-м.н., доцент,
Вычислительный центр им. А.А. Дородницына
ФИЦ ИУ РАН*

e-mail: averkin2003@inbox.ru

Ярушев С.А., к.т.н., доцент

РЭУ им. Г.В. Плеханова

e-mail: sergey.yarushev@icloud.com

1. ВВЕДЕНИЕ

Масштабное развитие систем искусственного интеллекта (ИИ), в том числе приложений на основе искусственных нейронных сетей, открывает широчайшие возможности их использования в различных областях, от систем распознавания эмоций, до систем предиктивной аналитики, применения в медицине и военной области. Но, в то же время, существующие системы и приложения имеют один общий существенный недостаток - невозможность интерпретации полученных результатов и принятых решений. Широко известная проблема так называемого черного ящика накладывает существенные ограничения для использования подобных систем, в том числе

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

законодательных, так как нельзя проследить ход принятия решения нейронной сетью.

В настоящее время эти проблемы решаются в рамках направления объяснимого ИИ (explainable artificial intelligence или ХАИ). Системы на базе объяснимого ИИ помогают пользователю принять с помощью методов машинного обучения решения, что повышает доверие к этим системам и дает возможности принимать более эффективные решения на основе результатов работы системы. Все это позволяет разработчикам и пользователям исследовать факторы, которые используются нейронной сетью при решении конкретной задачи и понять, какие параметры нейронной сети нужно поменять, чтобы повысить точность ее работы.

Историю развития объяснительных систем и моделей извлечения правил можно разделить на три этапа: первый этап (с 1970 г.) – разработка экспертных систем, второй этап (середина 1980-х годов) - переход от экспертных систем к системам, основанным на знаниях, и третий (с 2010 г.) - изучение глубоких архитектур искусственных нейронных сетей, потребовавший новых глобальных исследований по построению объяснимых систем. Далее, в разделах 2, 3 и 4 рассмотрим данные этапы подробнее.

2. ЭКСПЕРТНЫЕ СИСТЕМЫ В ОБЪЯСНИТЕЛЬНЫХ МОДЕЛЯХ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Системы объяснения, основанные на правилах. Первое и второе поколения объяснимого ИИ связаны с экспертными системами, включавшими в себя принятие решений и постановку диагнозов. Главный этап развития экспертных систем пришелся на начало 1970-х годов и содержал представления, основанные на знаниях, использовании правил и отношений. Системы имели инструментарий вопрос-ответных интерфейсов для пользователей и могли давать рекомендации и ставить диагнозы, основанные на правилах. Исследования показывали, что объяснение, как компьютерные системы работают, позволяет повысить уровень доверия людей к рекомендательным системам [1]. При интерпретации результатов работы экспертной системы также различались объяснение принципа работы системы и обоснование выбора архитектуры системы [2].

На первом этапе, в эпоху разработки экспертных объяснительных систем, велись масштабные исследования, демонстрирующие, что экспертные системы, имеющие объяснительную составляющую, способны оказывать большее

влияние на принятие решения, причем эффект был прямо пропорционален уровню навыков пользователя. [3-6].

Но многие экспертные системы первого поколения не принесли ожидаемых преимуществ. При работе над медицинскими экспертными системами исследователи признали, что врачи будут игнорировать рекомендации экспертной системы, если не будет предоставлено обоснование (оправдание) того, почему система дала такие рекомендации. Системы первоначального объяснения пытались предоставить это обоснование, описывая основные цели и шаги, используемые для постановки диагноза. Этот подход, который Свартаут и Мур назвали «Резюме как миф объяснения» [7], также не полностью устраивал пользователей, и это привело к переосмыслению подходов, целей и задач систем ХАИ. Первое поколение появилось в конце 1970-х годов [8] и просуществовало около 10 лет.

Некоторое несовершенство экспертных систем первого поколения породила в свою очередь первое поколение систем, включающее блок объяснения как обязательный элемент, например, Mycin [9] и связанные с ним системы Digitalis Therapy Advisor [10], BLAN [11] и другие системы объяснения специального назначения. Применяя деревья вывода, эти системы работали, создавая логические и вероятностные правила для постановки диагноза или ответов на вопросы. В целом, поскольку знания и опыт были сформулированы в терминах правил, эти правила описывались на естественном языке. Простое объяснение выражалось правилом для принятия решения. Такие объяснения обычно писались на ограниченном естественном языке, но часто они были простым использованием продукционных правил «если-то» для текстовых описаний.

В [12] рассмотрены системы первого поколения, которые создавали объяснения, перефразируя правила для принятия решения. В целом, экспертные системы и системы объяснения первого поколения были сосредоточены на описании внутренних состояний интеллектуальной системы, ее целей и планов. Иногда это было довольно просто, потому что сами правила являлись формализацией правил, используемых экспертами. Иногда, однако, языковые описания мало походили на человеческое объяснение или вообще на естественный язык. Во-первых, они полагались только на формат логических и причинно-следственных правил «если-то», а не на предоставление объяснений на более высоком уровне стратегии рассуждений (например, сбор базовой информации о пациенте, создание сети для альтернативных объяснений, попытки поддержать конкретную гипотезу). Во-вторых, некоторые правила были логически

необходимы для того, чтобы система работала, не обязательно имели смысл для пользователей и не имели отношения к стандартным вопросам пользователя «как» и «почему». В-третьих, знания предметной области иногда компилировались (например, причинно-следственные связи между симптомами и диагнозами), поэтому они не включались в объяснение.

Когда был изначально разработан *Mycin*, неспособность полностью объяснить набор правил и их обоснований не считалась недостатком, потому что создание каких-либо объяснений в удобочитаемой форме на базе записанного дерева рассуждений программы уже было сложной задачей и значительным достижением в области ИИ. Более того, такие ассоциативные модели представляли собой практические правила, основанные на демографических и физиологических данных, которые обычно были знакомы пользователям. Последние, как предполагалось, просто следовали советам программы после того, как она была протестирована и сертифицирована экспертами. Однако попытки расширить или уточнить эти наборы правил, использовать их для обучения или объяснить высокоуровневые ассоциации были неудачными.

3. СИСТЕМЫ ОБЪЯСНЕНИЯ, ОСНОВАННЫЕ НА ЗНАНИЯХ

К середине 1980-х годов ограничения систем объяснения первого поколения столкнулись с проблемой, которая заключалась в том, что недостаточно просто резюмировать внутреннюю работу системы. Созданный текст может быть верным, но это не обязательно то, что хотел узнать пользователь, или ему было непонятным. Системы второго поколения часто стирали грань между консультационной программой [12], наставником [13], советником [14] и системой ввода данных [15], но перед исследователями стояла цель выйти за рамки методов, которые не помогли улучшить понимание или принятие экспертных систем. Основной движущей силой систем второго поколения была необходимость разработки более абстрактных структур, которые способствовали бы повторному использованию и простоте разработки систем. *NEOMYCIN* сделал это, предоставив диагностическую процедуру на основе задач и метаправил и связанный с ней язык таксономических и причинно-следственных связей [16, 17].

На раннем этапе разработки объяснительных экспертных систем было признано, что их база знаний, правила и объяснения могут быть использованы для создания интеллектуальных наставников.

Отличительной особенностью интеллектуальных систем обучения является то, что они выводят ментальную модель предметной области каждого учащегося (его базу знаний) на основе поведения учащегося. Эти системы сначала собирательно назывались интеллектуальными системами компьютерной аналитики и рассуждений (computational analytics and intelligence, CAI), чтобы отличать интеллектуальных наставников от простых обучающих машин 1960-х годов, и применялись сообществом ИИ в образовании, особенно в 1980-х и начале 1990-х годов.

Например, в экспертной системе GUIDON запрос студента о данных пациента и его заявленные гипотезы использовались для обратного поиска через сеть правил MYCIN (или любую другую сеть), чтобы определить, какие правила не учитывались. Объяснение представляло собой сеть умозаключений и иногда включало неопределенности (например, свидетельство того, что студент знает правило, базирующееся на предыдущих взаимодействиях, нужно отличать от его применения в конкретном случае). Вот почему системы иногда называли «учителями, основанными на знаниях», в отличие от обучающих машин 1960-х годов. Главная идея заключалась в том, что предметные знания экспертной системы (например, правила медицинской диагностики) были отделены от базы знаний обучения (например, правил управления диалогом). Кроме того, процесс интерпретации правил предметной области, аналогичный объяснению поведения экспертной системы на основе модели (трассировки) ее внутренних процессов, использовался для объяснения поведения студента путем построения модели того, как студент рассуждает.

4. НОВОЕ ПОКОЛЕНИЕ ОБЪЯСНИТЕЛЬНЫХ СИСТЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

После некоторого застоя, связанного с замедлением развития направления систем, основанных на знаниях, возникает третий этап, относящийся к 2017–2021 гг.

Возникновение третьего этапа можно наглядно продемонстрировать, проанализировав количество публикаций по темам извлечения правил и объяснимого ИИ. При анализе количества публикации за последние 20 лет можно отметить, что интерес к извлечению правил из нейросетей более или менее стабилен (по некоторым источникам даже периодичен, что связано с тремя рассмотренными поколениями систем объяснимого ИИ) и возрастает только с появлением третьего поколения систем объяснимого ИИ (рис. 1), что связано в основном с созданием программы DARPA.

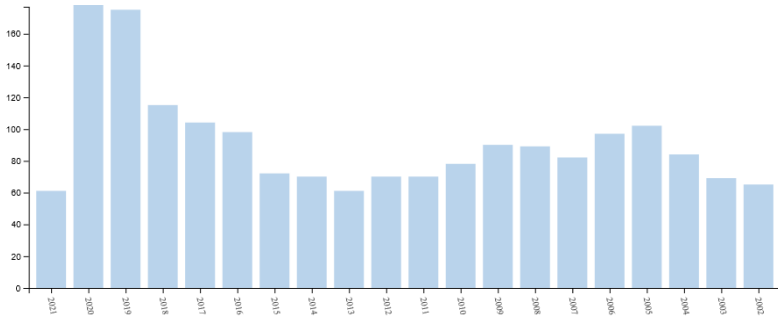


Рис.1. Количество публикаций в области извлечения правил из нейронных сетей

С появлением программы XAI в DARPA можно наблюдать взрывной рост количества публикаций непосредственно по теме объяснимого ИИ (начиная с 2018 г.) и его дальнейший рост, как показано на рис. 2.

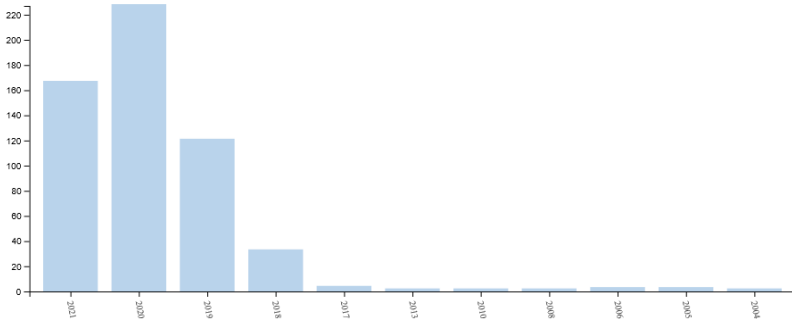


Рис.2. Количество публикаций по теме объяснимого искусственного интеллекта (XAI)

В системах третьего поколения, как и в системах первого поколения, осуществлялись попытки объяснить внутреннюю работу системы, что само по себе остается серьезной проблемой. Системы первого поколения встраивали экспертные знания в правила, часто получаемые непосредственно от экспертов, и пытались составить языковые описания на основе оценок экспертов. Эти правила часто трансформировались в выражения естественного языка, и большая часть этой работы была направлена на создание систем представления знаний. В системах третьего поколения эта задача оказалась значительно сложнее. Недостатки систем первого поколения, связанные со слабым уровнем детализации и непонятным языком,

могут стать проблемой для систем третьего поколения. Со времени систем первого поколения компьютерные технологии в визуализации данных, анимации, видео и т. п. значительно продвинулись вперед, и много новых идей было предложено в качестве потенциальных методов для генерации объяснений. Хотя системы первого поколения поддерживали диалоги на естественном языке и интерактивность в вопросно-ответных системах, современные системы делают это на более высоком уровне, чем экспертные системы. Примером может служить использование аргументированных объяснительных диалогов человек-машина для устранения несоответствий в базах знаний в процессе приобретения знаний с помощью программного обеспечения для экспертных систем [18].

Некоторые из объяснений, представленных в системах первого поколения, было легко создать по сравнению с объяснениями в третьем поколении, поскольку они были прямым повторением закодированных вручную правил [19]. Нынешнему поколению систем, возможно, будет труднее давать простые объяснения, производимые системами первого поколения. Но в некотором смысле системы третьего поколения, разрабатываемые в настоящее время, отражают достижения систем первого поколения. Ожидается, что многие из этих систем могут встретить проблемы, аналогичные тем, с которыми столкнулись системы первого поколения. И они могут быть решены методами, которые возникли в системах второго поколения.

5. ЗАКЛЮЧЕНИЕ

Извлечение правил является одним из способов помочь понять нейронные сети. Повышение прозрачности нейронных сетей путем извлечения из них правил имеет два основных преимущества. Это дает пользователю некоторое представление о том, как нейронная сеть использует входные переменные, чтобы принять решение, и позволяет выявить скрытые функции в нейросетях, когда правила применяются для объяснения работы отдельных нейронов. Таким образом, в дальнейших исследованиях мы планируем заняться разработкой моделей извлечения правил из различных архитектур нейронных сетей путем создания гибридных нейро-нечетких глубоких моделей искусственных нейронных сетей.

Литература

1. Bilgic M., Mooney R. J. Explaining Recommendations: Satisfaction vs. Promotion // Beyond Personalization Workshop. 2005. V.5. P. 153.

2. Swartout W.R., Moore J. D. Explanation in Second Generation Expert Systems. Second Generation Expert Systems. Springer, Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 1993. P. 543-585.
3. Chandrasekaran B., Tanner M.C., Josephson J.R. Explaining Control Strategies in Problem Solving // IEEE Expert. 1989. V. 4(1). P. 9–15.
4. Dhaliwal J. S., Benbasat I. The Use and Effects of Knowledge-Based System Explanations: Theoretical Foundations and a Framework for Empirical Evaluation // Information Systems Research. 1996. V. 7(3). P. 342–362.
5. Eining M. M., Dorr P. B. The Impact of Expert System Usage on Experiential Learning in an Auditing Setting // Information Systems. 1991. V. 5(1). P. 1–16.
6. Murphy D.S. Expert System Use and the Development of Expertise in Auditing: A preliminary investigation//Information System, 1990.V.4. P.18-35.
7. Lamberti D.M., Wallace W.A. Intelligent Interface Design: An Empirical Assessment of Knowledge Presentation in Expert Systems // MIS Quarterly. 1990. V. 14. P. 279-311.
8. Искусственный интеллект. Справочник в 3-х т. / Под ред. В. Н. Захарова, Э.В. Попова, Д.А. Поспелова, В.Ф. Хорошевского. М.:Радио и связь, 1990.
9. Попов Э. В. Экспертные системы: Решение неформализованных задач в диалоге с ЭВМ. М.: Наука, 1987. 288 с.
10. Swartout W.R. A Digitalis Therapy Advisor with Explanations // Proc. 5th International Joint Conf. on Artificial Intelligence. Cambridge, 1977. V.2. P.819-825.
11. Weiner J. L. BLAH, A System that Explains its Reasoning // Artificial Intelligence, 1980. V.15. P.19–48.
12. Swartout W.R., Paris C., Moore J. Explanations in Knowledge Systems: Design for Explainable Expert Systems // IEEE Expert. 1991. V. 6(3). P. 58–64.
13. Clancey W.J. Intelligent Tutoring Systems: A Tutorial Survey. Stanford: Stanford University Department of Computer Science, 1986. 56 p.
14. Sinha R., Swearingen K. The Role of Transparency in Recommender Systems // CHI'02 Extended Abstracts on Human Factors in Computing Systems. Minneapolis, 2002. P. 830–831.
15. Gruber T. Learning Why by Being Told What // IEEE Expert. 1991.V. 6(4), P.65–75.
16. Clancey W.J. Details of the Revised Therapy Algorithm. Rule-Based Expert Systems. The MYCIN Experiments of the Stanford Heuristic Programming Project. Reading, MA: Addison-Wesley, 1984. P. 133–146.
17. Clancey W.J. From GUIDON to NEOMYCIN and HERACLES in Twenty Short Lessons // AI Magazine. 1986. V. 7(3). P. 40.
18. Arioua A., Buche P., Croitoru M. Explanatory Dialogs with Argumentative Faculties Over Inconsistent Knowledge Bases // Expert Systems with Applications, 2017. V.80. P. 244-262.
19. Averkin, A. N., Yarushev S.A. Review of Research in the Field of Developing Methods to Extract Rules From Artificial Neural Networks / A. N. Averkin, S. A. Yarushev // Journal of Computer and Systems Sciences International. – 2021. – Vol. 60. – No 6. – P. 966-980. – DOI 10.1134/S1064230721060046. – EDN IADBXH.

КОНЦЕПЦИЯ ГИБРИДНОЙ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В ПРОЦЕССАХ ЛАЗЕРНОГО АДДИТИВНОГО ПРОИЗВОДСТВА*

Грибова В.В., д.т.н., с.н.с.,

e-mail: gribova@iacp.dvo.ru

Кульчин Ю.Н., д.ф.-м.н., академик,

e-mail: kulchin@iacp.dvo.ru

Никитин А.И.,

e-mail: anikitin@iacp.dvo.ru

Тимченко В.А., к.т.н.,

e-mail: vadim@dvo.ru

Институт автоматики и процессов управления ДВО РАН

1. ВВЕДЕНИЕ

В настоящее время аддитивные способы создания, восстановления металлических деталей, нанесения на поверхность изделий из металла функциональных покрытий и элементов уже хорошо зарекомендовали себя в качестве революционной технологии цифрового производства. Отметим, что в данном случае речь идет о штучном, максимум, серийном производстве. В производствах такого типа аддитивные технологические процессы являются реальной альтернативой традиционным (субтрактивным) методам изготовления и обработки металлических деталей. Их применение в высокотехнологичных отраслях производства позволяет создавать, восстанавливать, а также функционально модифицировать изделия, практически не накладывая ограничений на сложность их геометрии [1, 2].

Вместе с тем, существует ряд причин, по которым массовое внедрение динамично развивающихся технологий аддитивного производства (АП) с использованием лазерных технологических комплексов в промышленные производственные процессы сопряжено

* Работа выполнена при частичной финансовой поддержке РФФИ, проект №20-01-00449

с серьёзными трудностями [3-5]. Перечислим основные такие причины (препятствия).

1. *Высокий порог входа специалистов в сферу лазерного АП (ЛАП).* Как правило, требуется не один год практики и при этом большое количество дорогостоящих расходных материалов АП (металлические порошки, технологические газы и т.п.) для приобретения у специалиста-технолога необходимых знаний и опыта, позволяющих задать корректный режим работы оборудования для выполнения конкретной технологической операции.

2. *Необходимость учета влияния большого количества технологических параметров лазерной обработки на физико-механические характеристики результирующего изделия.* Исходя из характеристик обрабатываемой детали или подложки, особенностей оборудования и свойств используемого материала, режим работы оборудования (множество ключевых технологических параметров операции) должен быть настроен таким образом, чтобы обеспечить соответствие получаемых изделий требованиям действующих нормативных документов отрасли.

3. *Отсутствие должного уровня интеллектуальной поддержки таких процессов.* Ощущается острая нехватка долгоживущих программных систем поддержки принятия решений специалистами-технологами (операторами установок АП), постоянная адаптация которых под использование в аддитивных процессах различных обрабатываемых материалов, технологического оборудования и т.п. была бы оправданной по стоимости и трудозатратам (не в последнюю очередь из-за междисциплинарного характера данного процесса).

Перечисленные проблемы обуславливают актуальность создания программно-информационного комплекса интеллектуальной поддержки принятия аргументированных решений для операторов установок АП металлических деталей [6-8]. Одной из основных, решаемых с его использованием задач, является уменьшение числа ошибок, совершаемых оператором при подготовке управляющей программы для установки (системы) АП на этапе проектирования технологической операции.

В работах [9, 10] авторами был предложен гибридный подход к интеллектуальной поддержке операторов, основанный на использовании методов инженерии знаний (дедуктивный вывод на основе формализованных знаний); поиска по аналогии на основе прецедентов; индуктивного обобщения данных; а также методов традиционного математического моделирования физико-химических, термических процессов, сопровождающих технологические операции

АП. Использование методов поиска по аналогии и индуктивного обобщения данных делают также важной задачу консолидации информации о выполняемых операциях АП в едином технологическом пространстве. Настоящая работа является развитием предложенного подхода и посвящена описанию общей концептуальной архитектуры облачной технологической платформы – портала знаний о процессах АП металлических изделий с использованием лазерных технологических комплексов, его информационных и программных компонентов, а также их особенностей и подходов к разработке.

Рассматривается вид технологии АП, известный как «прямой подвод энергии лазера и материала» (laser directed energy deposition, LDED) [11]. В данном аддитивном процессе энергия сфокусированного излучения от внешнего источника (мощного лазера) используется для соединения материалов (например, металлосплавов) путём их сплавления в процессе нанесения.

2. ПОРТАЛ ЗНАНИЙ О ПРОЦЕССАХ ЛАЗЕРНОГО АДДИТИВНОГО ПРОИЗВОДСТВА

Концептуальная архитектура созданного на облачной платформе IASaaS [12] портала знаний о процессах ЛАП основана на онтологическом двухуровневом подходе к формированию данных и знаний (рис. 1). В таком подходе онтологии *явно отделены* от баз данных и знаний. Сначала инженерами знаний совместно со специалистами в области ЛАП создаются и формируются (а впоследствии, по мере необходимости, совершенствуются) необходимые онтологические модели представления знаний. Затем в соответствии с этими онтологиями специалисты в данной области могут *самостоятельно* (без участия инженеров знаний) формировать и развивать декларативные базы в терминологически понятном для них концептуальном представлении (как онтологии, так и базы данных и знаний представляются иерархическими орграфами). Структура портала знаний представляет собой дерево разделов (папок), терминальными вершинами которого являются информационные и/или программные компоненты (ресурсы) портала.

| IACPaaS | | Лазерное аддитивное производство | Онтологии (17) ↑ |
|--------------------------|--|----------------------------------|--|
| 🏠 О платформе | Переводы | | <ul style="list-style-type: none"> Онтология базы порошковых питателей Онтология базы дефектов металла |
| 📰 Новости | Онтологии | | <ul style="list-style-type: none"> Онтология базы технологических газов Онтология системы единиц физических величин |
| 👤 Фонд | Агенты | | <ul style="list-style-type: none"> Онтология базы технологических лазеров |
| 📁 Личный кабинет | Решатели | | <ul style="list-style-type: none"> Онтология базы оборудования для перемещения рабочего инструмента Онтология базы лазерных оптических головок |
| 👤 Мой Фонд | <ul style="list-style-type: none"> База прецедентов База технологических газов База дефектов металла База технологических лазеров База оптических головок Система единиц физических величин База порошковых питателей Классы процессов лазерной обработки База химических элементов Справочник по материалам База металлопорошковых композиций База знаний о настройках режимов лазерного аддитивного производства База технологических насадок для процессов лазерной обработки База оборудования для перемещения рабочего инструмента База документированных технологических процессов База математических моделей | | <ul style="list-style-type: none"> Онтология базы металлопорошковых композиций Онтология базы химических элементов Структура описания характеристик Онтология справочника по материалам Онтология базы прецедентов Онтология баз знаний о настройках режимов лазерной обработки Онтология технологических процессов лазерной обработки Онтология классификации процессов лазерной обработки Онтология базы математических моделей |
| 🔧 Работающие Сервисы | | | |
| 🔒 Совместный Доступ | | | |
| ⚙️ Рассмотрение Запросов | | | |
| 🔍 Мои Запросы | | | |
| 📧 Сообщения | | | |
| 👤 Профиль | | | |

Рис.1. Портал знаний о процессах АП металлических изделий с использованием лазерных технологических комплексов

Опишем далее состав, назначение и основные особенности информационных и программных компонентов портала знаний.

2.1. ИНФОРМАЦИОННЫЕ КОМПОНЕНТЫ ПОРТАЛА ЗНАНИЙ

К информационным компонентам портала относятся система онтологий, а также создаваемые на их основе: набор баз данных и справочников, используемых в ЛАП, база знаний о настройках режимов ЛАП, база структурированных прецедентов и база математических моделей, описывающих физико-химические, термические процессы, сопровождающие операции лазерной обработки материалов.

Система связанных онтологических моделей обеспечивает формирование всех справочников и баз данных, а также базы знаний в унифицированном, согласованном по терминологии виде, который соответствует стандартам и представлениям экспертов в этой области. Для формального представления онтологий используется язык

описания онтологий облачной платформы IASaaS. Для формирования онтологий на портале знаний используется инструментальное средство платформы «Редактор онтологий».

2.1.1. БАЗЫ ДАННЫХ И СПРАВОЧНИКИ

Базы данных и справочники содержат информацию об определяющих (ключевых) параметрах *компонентов системы ЛАП и обрабатываемых (расходных) материалов АП*. К первым относятся технологические лазеры, укомплектованные лазерные оптические головки, устройства, обеспечивающие перемещение головок относительно обрабатываемой поверхности (например, многоосевые промышленные роботы-манипуляторы), порошковые питатели, технологические насадки и т.д. Ко вторым – металлопорошковые композиции на основе различных металлических сплавов, технологические газы (транспортный, защитный, обжимающий). Определяющими (ключевыми) считаются основные параметры лазерной установки АП и вспомогательного оборудования (как системы АП), а также характеристики материалов АП, влияющие на ход процесса лазерной наплавки и определяющие его результат.

Такой интегрированный в едином информационном пространстве комплекс коллективно развиваемых облачных баз данных и справочников, во-первых,

- представляет самостоятельную ценность как централизованный, общедоступный и постоянно пополняемый источник справочной информации, которая будет полезна как специалистам в области ЛАП, так и операторам лазерных технологических комплексов (в том числе, в процессе их обучения); а, во-вторых,

- необходим для формирования на его основе базы формализованных протоколов технологических операций (прецедентов) ЛАП и базы знаний о настройках режимов ЛАП.

Во втором случае базы данных и справочники представляют набор терминологических словарей, определяющих множества допустимых названий (с синонимией и обозначениями) компонентов системы ЛАП и используемых материалов с их структурной детализацией и областями допустимых значений.

2.1.2. БАЗА ЗНАНИЙ

База знаний о настройках режимов ЛАП содержит формально представленные в декларативном виде зависимости, устанавливающие взаимосвязи между (1) составом и свойствами обрабатываемых

деталей, (2) составом и свойствами наплавляемых порошковых материалов, (3) создаваемой в процессе наплавки газовой средой, (4) режимом наплавки (набором ключевых регулируемых рабочих параметров аддитивного процесса), и свойствами результирующих деталей.

Для каждого ключевого параметра аддитивного процесса задается множество правил установки его возможных значений. Антецедент и консеквент производственных правил представляют собой множество связанных между собой логическими связками равенств термов.

База знаний формируется коллективно квалифицированными специалистами-технологами и экспертами в области ЛАП на основе собственного опыта и информации из различных неформализованных источников (в том числе государственных и отраслевых стандартов, документации всех видов, научно-технической, учебной и справочной литературы). Экспертное формирование базы осуществляется с использованием специализированного редактора, управляемого онтологией базы знаний, а также комплекса справочных информационных баз и онтологии базы формализованных протоколов технологических операций (определяющей множество допустимых названий ключевых рабочих параметров процесса). База знаний является развиваемой, работы по её наполнению продолжают.

2.1.3. БАЗА ПРЕЦЕДЕНТОВ

База структурированных прецедентов, формируемая на основе соответствующей онтологии, содержит набор формализованных протоколов проведённых технологических операций ЛАП, структурированных по видам обрабатываемых сплавов и типам аддитивного технологического процесса, а также распределённых по классам прецедентов в зависимости от результатов их выполнения.

База прецедентов – необходимый элемент системы управления качеством базы знаний и поиска по аналогии в случаях, когда в базе знаний отсутствуют знания, соответствующие входным данным. Чтобы система с базой знаний оставалась полезной для специалиста, мера качества (как минимум, мера правильности, а в идеале – точности) ее базы знаний в течение всего времени использования не должна ухудшаться (в том числе и при добавлении в нее новых знаний). Для достижения этой цели необходимо постоянно накапливать базу прецедентов, а также классифицировать их для оценки качества базы знаний. Функция накопления базы прецедентов – получение информации обо всех задачах, решенных системой

поддержки принятия решений (СППР), включая результаты верификации (оценки специалистом) каждого решения задачи. Предложена следующая классификация прецедентов:

1. СППР предложила правильное и точное решение;
2. СППР предложила правильное, но неточное решение;
3. СППР предложила неправильное решение.

С поддержкой эксперта (отвечающего за качество базы знаний) возможна расширенная классификация прецедентов 2 класса: входные данные задачи допускают более точное решение; входные данные задачи не допускают более точное решение. Множество прецедентов, отнесенных к классам 1 и 2 «входят» в меру качества. Допустимой является такая модификация базы знаний, которая не ухудшает ее меру качества.

Базу прецедентов предлагается использовать для выполнения двух процессов.

1. Для поиска по аналогии в базе прецедентов – в случае, когда невозможно получить решение на основе базы знаний (соответствующие знания в ней отсутствуют).

2. Для формирования базы знаний с использованием методов индуктивного обобщения данных (когда объем базы прецедентов станет достаточным для обучения).

С использованием редактора, управляемого соответствующей онтологией, а также комплекса справочных информационных баз предметными специалистами коллективно наполняется база формализованных протоколов технологических операций ЛАП. На сегодняшний день сформировано уже порядка 300 формализованных протоколов по наплавке металлопорошковых композиций на основе олова, железа, алюминия, меди, титана и магния; при этом база содержит 28 559 понятий и 31 098 отношений. Работы по коллективному наполнению базы формализованных протоколов продолжаются.

2.1.4. БАЗА МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ

Значения ключевых параметров технологического процесса необходимо подбирать, в том числе, на основе результатов численного моделирования макропроцессов тепломассопереноса, плавления, конвекции и прочих физико-химических процессов, сопровождающих технологию LDED. В связи с этим требуется обеспечить взаимодействие СППР с внешними средствами математического моделирования физико-химических процессов лазерной обработки материалов. При этом используются математические модели

процессов, протекающих при лазерном сплавлении металлического порошка, подаваемого в движущуюся ванну расплава, сформированную сфокусированным лазерным пучком. На основе существующих и разрабатываемых моделей, включающих согласованные уравнения динамики формирования ванны расплава, выполняются программно реализованные численные расчеты тепловых полей, структуры конвективных течений расплава и геометрических характеристик создаваемого физического объекта. Полученные значения необходимых параметров процесса сплавления материала используются для принятия решений по настройке режима лазерной обработки.

Для централизованного, структурированного хранения программных реализаций численных расчетов для моделируемых процессов (объектов), в состав информационного наполнения портала знаний входит база математических моделей, онтология которой представлена на рис. 2.

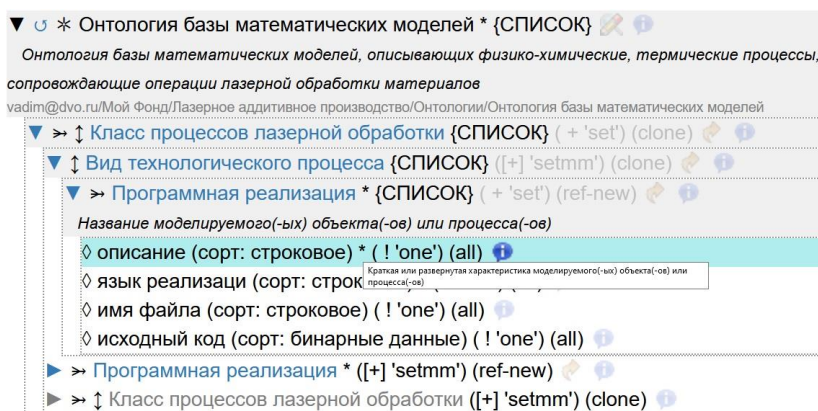


Рис.2. Онтология базы математических моделей, описывающих физико-химические, термические процессы лазерной обработки материалов

Структуризация данного хранилища организована в соответствии с классами процессов лазерной обработки (в том числе, с учетом видов обрабатываемых материалов).

База математических моделей является общедоступным коллективно развиваемым облачным ресурсом портала знаний о технологических процессах ЛАП, которая будет полезна специалистам в области лазерной обработки материалов. В настоящее время работы по её усовершенствованию и наполнению продолжаются.

2.2. ПРОГРАММНЫЕ КОМПОНЕНТЫ ПОРТАЛА ЗНАНИЙ

К программным компонентам портала относятся следующие средства.

Специализированные, управляемые онтологиями *редакторы для формирования и сопровождения всех баз данных и базы знаний*. При изменении онтологий такие редакторы (их пользовательский интерфейс и процесс редактирования) автоматически адаптируются под эти изменения.

Интеллектуальная *СППР*, основанная на методах инженерии знаний, поиска по аналогии, а также методах традиционного математического моделирования. Исходя из требуемых характеристик результирующего изделия, характеристик обрабатываемой детали или подложки, особенностей оборудования и свойств используемого материала, система должна давать оператору рекомендации по настройке оптимальных (по критерию качества процесса наплавки) режимов работы оборудования. В рамках решения данной задачи будут использованы различные методы и подходы, имитирующие различные способности естественного интеллекта.

1. Дедуктивный вывод (вывод на основе базы знаний) для случаев, когда имеются знания о настройках режимов ЛАП. Особенностью обрабатываемой базы знаний является использование при её формировании комплекса семантических структурных моделей представления информации, который позволит СППР обеспечивать их объяснимость, в том числе аргументацию предлагаемых рекомендаций по настройке режимов ЛАП (наплавки).

2. Рассуждения по аналогии (рассуждения на основе прецедентов), в случае, если в базе знаний отсутствуют соответствующие знания (при использовании, например, новых, уникальных расходных материалов, оборудования, производственные «ноу-хау» и т.п.). Для реализации поиска по аналогии в накапливаемой базе структурированных прецедентов будет использован метод *k*-ближайших соседей (*k*-*NN*). При этом для уточнения прецедентов предлагается семантическая индексация прецедентов, которая позволяет учесть знания экспертов предметной области – коэффициенты важности параметров, а также ввод экспертом пороговых значений степени сходства параметров [13].

3. Методы индуктивного обобщения данных будут использованы для случаев, когда накопленные массивы данных о проведенных технологических операциях лазерной обработки позволяют выявить новые закономерности и сформировать новые

знания. Принципиально важно, что эти методы будут также основаны на построении интерпретируемой базы знаний в соответствии с онтологией.

Наконец, для обеспечения возможности коммуникации СППР с *математическими программными средствами для моделирования физико-химических процессов*, сопровождающих технологию LDED, была предложена схема, разработаны модели и средства удаленного взаимодействия СППР с программным обеспечением – скриптами на языке Wolfram Language программной системы Mathematica [14].

Применение моделей и методов онтологического двухуровневого подхода направлено на обеспечение масштабируемости (оперативной расширяемости) предлагаемых в СППР решений. Появление новых видов расходных материалов, лазерного и прочего технологического оборудования не должно приводить к перепрограммированию разработанных алгоритмов, а также изменению моделей представления знаний. Таким образом, будет обеспечено одно из ключевых требований к таким программным системам – их жизнеспособность.

3. ЗАКЛЮЧЕНИЕ

В работе представлена общая концептуальная архитектура облачного портала знаний о процессах АП металлических изделий с использованием лазерных технологических комплексов. Описан состав и назначение его информационных и программных компонентов, а также подходы к их разработке. Портал знаний создан на платформе IACPaas (<https://iacpaas.dvo.ru>). К настоящему времени разработан ряд информационных и программных компонентов портала, ведутся работы по созданию новых, а также развитию и усовершенствованию существующих компонентов.

Создание и внедрение в ЛАП изделий из металлпорошковых композиций СППР для операторов лазерных технологических комплексов позволит снизить порог входа специалистов в эту область (и, соответственно, снизить затраты на обучение таких специалистов), а также производственные и утилизационные затраты. Вместе с этим, это будет способствовать увеличению спроса на выпуск компонентов лазерных технологических комплексов и снижению стоимости АП за счет более эффективного использования дорогостоящего высокотехнологичного оборудования и сырьевого материала.

Литература

1. Gu D., Meiners W., Wissenbach K., Poprawe R. Laser Additive Manufacturing Of Metallic Components: Materials, Processes And Mechanisms // *International Materials Reviews*. 2012. 57(3). P.133-164.
2. Razvi S., Feng S., Narayanan A., Lee, Y., Witherell, P. A Review Of Machine Learning Applications In Additive Manufacturing // *Proceedings of the ASME 2019 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Anaheim, CA, 2019.
3. Yang L., Hsu K., Baughman B., Godfrey, D. et al. Additive Manufacturing of Metals: The Technology, Materials, Design and Production. Springer, 2017.
4. Qi X., Chen G., Li Y., Cheng X., Li Ch. Applying Neural-Network-Based Machine Learning to Additive Manufacturing: Current Applications, Challenges, and Future Perspectives // *Engineering*. 2019. 5. P.721-729.
5. Wang Y.B., Zheng P., Peng T., Yang H.Y., Zou J. Smart Additive Manufacturing: Current Artificial Intelligence-Enabled Methods And Future Perspectives // *Science China Technological Sciences*. 2020. 63(9). P.1600-1611.
6. Sanfilippo E.M., Belkadi F., Bernard A. Ontology-Based Knowledge Representation For Additive Manufacturing // *Computers in Industry*. 2019. Vol. 109, P.182–194.
7. Mbow M.M., Grandvallet C., Vignat F., Marin, Ph. et al. Mathematization Of Expert's Knowledge: Example Of Part Orientation In Additive Manufacturing // *Journal of Intelligent Manufacturing*, Springer Verlag (Germany). 2021. P.1-19.
8. Ko H., Witherell P., Lu Y., Kim S., Rosen D.W. Machine Learning And Knowledge Graph Based Design Rule Construction For Additive Manufacturing // *Additive Manufacturing*. 2021. Vol. 37. 101620.
9. Gribova V., Kulchin Y., Nikitin A., Timchenko V. The Concept of Support for Laser-Based Additive Manufacturing on the Basis of Artificial Intelligence Methods // *Lecture Notes in Computer Science*. 2020. Vol 12412. P.403-415.
10. Грибова В.В., Тимченко В.А. Концепция поддержки лазерного аддитивного производства на основе онтологического подхода // *Онтология проектирования*. 2020. Т. 10. №2(36). С.176-189.
11. Yadav S., Paul C.P., Jinoop A.N., Rai A.K. et al. Laser Directed Energy Deposition based Additive Manufacturing of Copper: Process Development and Material Characterizations // *Journal of Manufacturing Processes*. 2020. 58. P.984-997.
12. Грибова В.В., Клещев А.С., Москаленко Ф.М., Тимченко В.А., Федорищев Л.А., Шалфеева Е.А. Облачная платформа IASaaS для разработки оболочек интеллектуальных сервисов: состояние и перспективы развития // *Программные продукты и системы*. 2018. Т. 31, №3. С.527-536.
13. Varshavskii P.R., Eremeev A.P. Modeling Of Case-Based Reasoning In Intelligent Decision Support Systems // *Scientific and Technical Information Processing*. 2010. 37(5). P.336-345.
14. Кульчин Ю.Н., Грибова В.В., Тимченко В.А., Полоник М.В., Пивоваров Д.С., Яцко Д.С., Никифоров П.А., Никитин А.И. Методология проектирования лазерного аддитивного процесса формирования решетчатой периодической структуры на поверхности пластины из алюминиевого сплава // *Автоматизация в промышленности*. 2021. №10. С.27-36.

ГИБРИДНЫЕ ОБЛАЧНЫЕ IASRAAS-СЕРВИСЫ ОЦЕНКИ РИСКОВ КРИТИЧЕСКИХ СОСТОЯНИЙ*

Грибова В.В., д.т.н., зам.дир.

e-mail: gribova@iacp.dvo.ru

Шалфеева Е.А., к.т.н., с.н.с.

e-mail: shalf@iacp.dvo.ru

Институт автоматизи и процессов управления ДВО РАН

1. ВВЕДЕНИЕ

В настоящее время появляется множество программных систем и сервисов поддержки задач профессиональной деятельности, для решения которых используются методы искусственного интеллекта, машинного обучения, математического моделирования.

Спектр приложений в медицине, для которых используются указанные выше методы, также достаточно широк. При этом часто для решения одной задачи или класса задач применяются различные подходы и методы их решения. При их использовании возникают следующие проблемы: как выбрать наиболее подходящий (точный) метод решения (как правило, это зависит от набора входных данных пациента в электронной медицинской карте); как сравнить полученные результаты (практика показывает, что каждому методу соответствует свой отдельно функционирующий сервис); как обеспечить объяснение и выдачу результата в понятных врачу терминах.

Данная проблема усугубляется еще и тем, что в настоящее время электронные медицинские карты (ЭМК) практически не интегрируются с такими сервисами, поэтому требуется двойной ввод данных врачами. Они заполняют ЭМК в рамках некоторой медицинской информационной системы, и повторно вводят данные в сервис для помощи врачу (это может быть система поддержки принятия решений, калькулятор и т.п.). Такие системы, как правило, не имеют объяснительной компоненты, вследствие чего доверие к ним у врачей крайне низкое. Ни одна из сред разработки пока не обеспечила такую семантическую структурированную модель

* Работа выполнена при частичной финансовой поддержке РФФИ, проекты № 19-29-01077 и № 20-0700670.

информации, которая бы позволяла создавать программные сервисы для помощи в решении важных врачебных задач (решаемых с использованием вербализуемых знаний и опыта), используя смысл данных об изменяющемся во времени состоянии организма, накопленных в ЭМК, а также имеющих объяснительную компоненту. Также нам не известны среды и средства объединения методов машинного обучения, математического моделирования и инженерии знаний для решения такого класса задач.

В качестве технологии для синтеза (многоагентных) интеллектуальных пространств и совместной работы искусственных агентов предложены онтологии. Однако стоит констатировать, что в качестве унифицированной интегрирующей основы при создании гибридных интеллектуальных систем с объяснением генерируемых результатов, в том числе альтернативных, они практически не используются. Под гибридной будем понимать такую систему, в которой для решения некоторой задачи используются различные методы (возможно и альтернативные, реализованные различными научными или экспертными группами).

В данной работе предлагается единая семантическая модель для описания знаний, аккумулирующих различные методы решения задач прогноза заболеваний/состояний, оценки рисков различных событий. Такая семантическая модель является основой для построения систем поддержки принятия врачебных решений, сервисов прогноза и оценки рисков критических состояний пациентов как по отдельно взятым заболеваниям/состояниям, так и произвольным их группам.

2. АВТОМАТИЗАЦИЯ ОЦЕНКИ РИСКОВ КРИТИЧЕСКИХ СОСТОЯНИЙ

В настоящее время имеется «широкая линейка» различных прогнозных и оценочных моделей, методов и платформ их реализации. Активно применяются методы машинного обучения, математического моделирования, инженерии знаний для решения сходных задач [1-6]. В большинстве это отдельные сервисы, реализующие одну прогнозную модель, хотя и есть примеры, объединяющие две или несколько моделей [7-9]. Важно, чтобы пользователь мог сравнить полученные результаты (решать одну задачу различными подходами) для «компенсации неточности» существующих методов.

Для поддержки решения задач прогноза критических состояний, оценки рисков применяют различные методы, а для реализации – различные подходы, модели реализации, средства и платформы. Это затрудняет возможность их комплексного использования: чтобы

использовать несколько сервисов, приходится для каждого заново вводить одну и ту же информацию. **Чаще всего** предложенные программные медицинские решения используют сокращенное количество параметров, тогда как **обращение к** электронной медицинской карте может обеспечить возможность в фоновом режиме производить оценивание пациента при появлении новых данных о нем в истории болезни или электронной медицинской карте.

Большинство готовых медицинских сервисов поддерживают анализ одной патологии (состояния или события), однако, есть примеры, когда сервис поддерживает анализ нескольких патологий (<http://webiomed.ai/>). Учитывая, что задачи оценки рисков, прогноза состояния/заболевания пациентов актуальны для большого числа состояний и заболеваний [5,6,10], семантическая модель знаний и методов прогноза состояний и оценки рисков не должна зависеть от конкретного заболевания. Единая семантическая модель обеспечит масштабируемость создаваемых прогнозных сервисов и систем и инструментов их создания.

Результатом таких прогнозных сервисов является процент вероятности события, состояния или заболевания, что недостаточно для принятия решений. Там, где поддержка решения делается не для профилактических мер или предупреждения, а для выбора тактики лечения, особенно персонифицированного, врачу необходимо объяснение ожидаемых исходов. Дополнением может быть приблизительно схожий процент вероятности при использовании других аналогичных сервисов. Однако для сервисов, реализованных на основе арифметических вычислений и методов машинного обучения, свойственно отсутствие обоснования решений – обнаруженных рисков, диагнозов и прогнозов (объяснения нет в сервисах: Webiomed – WML.SymptomChecker, WebMD, Isabel). В сервисах, использующих знания, объяснение может генерироваться. Например, у сервиса Infermedica [8] по каждому диагнозу из сформированного списка оно дается в виде «Грипп. Умеренные доказательства – наличие симптомов: быстрое сердцебиение, Головная боль, Сухой кашель, и т.д.). В некоторых случаях в качестве объяснения можно рассматривать описание типичных признаков, характеризующих состояние или заболевание (в klinica.com такое описание размещено на странице сайта с сервисом). Объяснением могут «служить» вопросы на экране, на которые дан ответ «да» (в health.mail.ru таковые вместе с фразой «можно предположить острый бронхит»). Иногда формируется фрагментарный объясняющий ответ, например: «гнойный налет с

миндалин не снимается. Можно предположить дифтерию» (в сервисах health.mail.ru или klinica.com. ua/simptom-checker).

Понятность объяснения обеспечивается используемой терминологией и онтологией предметной области (ПрОбл). Системы поддержки принятия решения на основе онтологических знаний [9] выдают детализированное объяснение. Для его генерации используется информация о соответствии вариантов решения, получаемым из базы знаний, – информации об исходных данных задачи. Объяснение сводится к информации обо всех рассмотренных гипотезах при формировании решения, а также источнике знаний, по которым была создана база знаний.

3. ОНТОЛОГИЧЕСКАЯ ОСНОВА ДЛЯ СОЗДАНИЯ ГИБКИХ СИСТЕМ ОЦЕНКИ КРИТИЧЕСКИХ СОСТОЯНИЙ

В процессе эксплуатации прикладных систем поддержки профессиональной деятельности возникает необходимость в адаптации к изменениям предметной области – ожидается вариативность знаний, в т.ч. о новых методах исследования, о новых выявляемых влияющих на организм факторах.

Медицинские знания (и клинические рекомендации как их разновидность) нельзя «вшить» в программы, т.к. они динамичны. Подход к созданию и обслуживанию систем поддержки принятия клинических решений должен учитывать эту особенность: системы должны иметь одну часть, которая постоянно развивается и другую часть, которая может ее читать и понимать, то есть быть интерпретатором.

Гибкость систем и сред их создания отвечает технологическому тренду Gartner на 2022 год – платформы, которые позволяют создавать новые архитектуры приложений, обладающих свойствами устойчивости, эластичности и гибкости, что позволяет отраслям экономики оперативно реагировать на быстрые цифровые изменения [11]. Платформа IASaaS [12] имеет инфраструктуру для формирования онтологической основы Порталов знаний и сервисов. Благодаря ей базы знаний клинических рекомендаций (БЗ) и данных понятны человеку и интерпретируемы ЭВМ. А благодаря инструментальной поддержке разработчиков, программные компоненты создаются как интерпретаторы БЗ и данных. Программные компоненты – Решатели – сохраняют свои результаты в единообразно интерпретируемых информационных ресурсах IASaaS, чем закладывается другой аспект гибкости отдельных сервисов через

комбинируемость (из решателей) пространств принятия решений. На основе информационной модели клинических знаний и документов на платформе IASPaas онтологии (для порталов) формируются в виде иерархических семантических сетей.

4. ОНТОЛОГИЯ ЗНАНИЙ О СОСТОЯНИЯХ ОРГАНИЗМА И СВЯЗИ РИСКОВ ЗАБОЛЕВАНИЙ С ИХ ПОКАЗАТЕЛЯМИ

Семантическое моделирование с помощью современных методов инженерии знаний позволяет представлять разные типы знаний о заболеваниях. Одни показывают зависимость состояний организма от сочетания наблюдаемых свойств и влияющих факторов (или вариантов развития патологий в зависимости от известных факторов), другие – математически выражают зависимости показателей здоровья.

Для **интегрирующего** описания рисков заболеваний, способа диагностики заболеваний и альтернативных способов их определения предложена следующая модель:

$\{ \langle \text{Критическое состояние} \rangle_i, \langle \{ \text{риск}, [\text{имя/автор методики},] \text{исходные данные}_i, [\text{условия применимости}], \text{способ определения} \rangle_j \}_{i=1, \text{ число состояний}, j=1, \text{ число методик}}$.

Элементы модели таковы:

Критическое состояние – имя заболевания или патологического состояния, для которого делается риск либо вычисляется прогноз;

риск = $\langle \text{вид угрозы}, \text{вероятность угрозы} \rangle$,

вид угрозы = (риск появления | риск перехода в тяжелую стадию | прогноз продолжительности жизни | риск неблагоприятного исхода (смерти));

вероятность угрозы = (Очень высокий риск | Высокий риск | Умеренный риск | Низкий риск | Отсутствие риска).

Исходные данные – предикторы, по значениям которых вычисляется риск или угроза:

Исходные данные_j = {Наблюдение_{jk}}.

Каждый элемент исходных данных (наблюдение) может быть нетемпоральным и темпоральным. Для нетемпоральных исходных данных описание имеет следующий вид: Наблюдение = (Фактор | Признак | Событие), для темпоральных дополнительно описывается временная характеристика: Наблюдение = $\langle (\text{Фактор} | \text{Признак} | \text{Событие}), \text{временная характеристика} \rangle$. Временная характеристика – интервал времени или числовое значение, в течение которого фактор/событие/признак наблюдались или происходили.

К факторам относятся метрики пациента, вредные привычки, имеющиеся заболевания и пр. Событием может быть лечебная процедура, операция, контакт с больным, веществом или и др. Признаки – это лабораторные или инструментальные исследования пациента, жалобы и другие показатели состояния организма пациента, наблюдаемые к моменту определения риска или прогноза.

При описании рисков заболеваний в Исходных данных к названию наблюдения добавляется ожидаемый диапазон значений: Фактор = <имя/название фактора, диапазон значений>. Событие, как и фактор, описывается своим именем, возможно, имеет уточняющие параметры (с именем и значением): Событие = <название, {имя параметра, диапазон значений}>.

Признак может быть простым или составным. Простой признак = <название признака, диапазон значений>. Составной признак = <название признака, {<название характеристики |элемента, диапазон значений}>>. Значения бывают двух типов – количественные (числовые) с единицами измерения (ед.изм.) и качественные (строковые, категориальные). Диапазон значений = ({значение категориального типа} | <числовой диапазон, ед.изм.>).

Способ определения (величины) задает вид вычисления прогноза/риска по сведениям о конкретном пациенте:

способ определения = <декларативные знания для распознавания угрозы | вычисляемая величина>.

Вычисляемая прогнозная шкала может опираться на формулу с вычислением, математическую модель, модель машинного обучения (стохастический градиентный бустинг| нейросеть или др.) или на другой способ получения вероятности риска, возможно явно показывающий «вклад» каждого предиктора в итоговую величину. Часто вычисленным значениям сопоставляются градации некоторой экспертной шкалы (если она полезнее вычисленной вероятности).

На основе этой модели на платформе IACPaas была сформирована онтология (см. рис. 1).

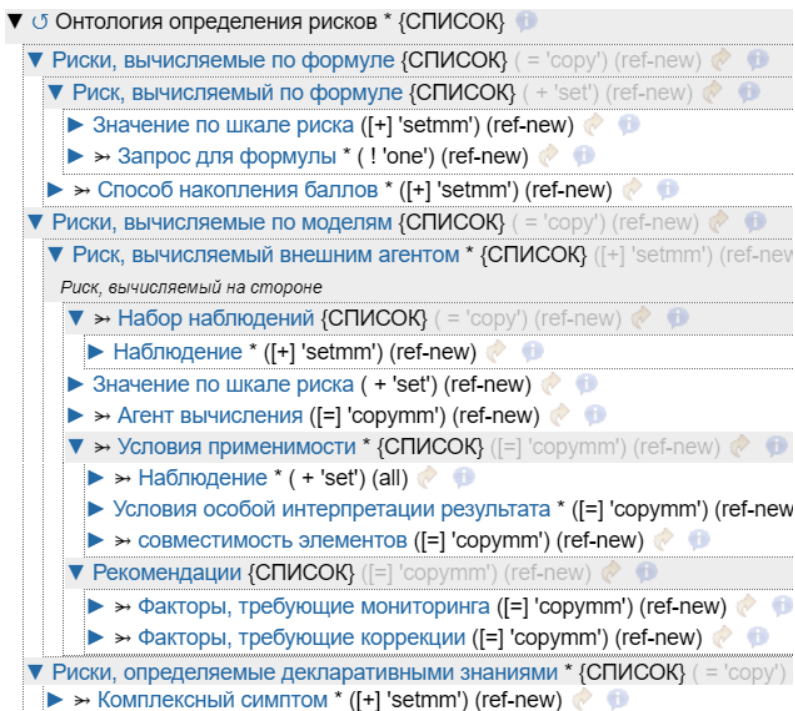


Рис.1. Онтология знаний о связи рисков заболеваний с их показателями

Знания, добываемые разными методами, формируются в единой структуре и форме (понятной специалистам предметной области).

5. ИСПОЛЬЗОВАНИЕ ОНТОЛОГИЧЕСКОЙ ОСНОВЫ IASRAAS И ОНТОЛОГИЯ ЗНАНИЙ ДЛЯ РАЗРАБОТКИ БАЗЫ ЗНАНИЙ

Методы извлечения знаний зависят от ее типа, источника и других причин, но результат формализации должен удовлетворять требованиям понятности, интерпретируемости.

Онтологические модели, реализованные на платформе IASraaS, позволяют формировать разные типы знаний в любых областях медицины, в т ч извлекаемых из клинич рекомендаций, статей и массивов данных, и интегрировать их в процессе поддержки решения важных задач прогноза, оценки состояния и рисков неблагоприятных процессов.

Наиболее востребованы базы знаний об оценках риска развития сердечно-сосудистых заболеваний (ССЗ). В настоящее время в них рассматривается более 200 модифицируемых и немодифицируемых предикторов, с различной степенью влияющих на развитие, прогрессирование и летальность ССЗ, которые используются в разных шкалах.

Сформированная БЗ ССЗ содержит описание основных прогностических шкал и моделей, широко используемых в России и за рубежом, новых прогностических моделей, более высокой прогностической точности для определения и оценки рисков развития заболеваний ССЗ и их осложнений на персональном и популяционном уровне. БЗ использует предикторы (признаки), которые описаны в Базе медицинской терминологии и наблюдений. Для каждой шкалы или модели в БЗ прописаны требуемые признаки, используемые в них. Если способом вычисления результата является формула, то в базе знаний создается модель такого вычисления, либо задается ссылка на агент, производящий такие вычисления. Градации риска в каждой шкале или модели определяется разным их количеством и качеством. Так риск неблагоприятного исхода коронарного шунтирования по шкале EuroScoreII имеет 3 градации: низкий, умеренный, высокий, Риск ССЗ по индексу Кетле - 5: низкий, среднепопуляционный, повышенный, высокий, очень высокий, чрезвычайно высокий.

Компонентом сервиса выдвижения гипотез о существующих рисках (рис.2) является решатель, проводящий анализ всех разделов медицинских документов (истории болезни или электронной медицинской картой) и определение рисков в соответствии с базой знаний. Для определения риска у конкретного пациента используют сведения о нем: набор исходных данных – названия наблюдений со значениями наблюдений.

6. ЗАКЛЮЧЕНИЕ

Данная модель реализована и апробирована на платформе IASPaas. Создана версия базы знаний о рисках. Реализован развиваемый словарь всех терминов, допускаемых в описании знаний и в документах из реальной медицинской практики.

На платформе реализован решатель, интегрированный с историей болезни и электронной медицинской картой, позволяющий на основе базы знаний и анализа всех разделов этих медицинских документов генерировать гипотезы с объяснением.

→ ИБ 2-им [История болезни или наблюдений (в. дек 2020)]

- ▶ Электронная медицинская карта
- ▼ Выявление рисков
 - ▼ Выявленные риски
 - ▼ Риск внутригоспитальной летальности после коронарного шунтирования [Комплексный]
 - ▼ имеется
 - ▼ Комплекс признаков 1 Минимальное количество признаков = 3 [Комплекс признако]
 - ▶ Возраст [Фактор]
 - ▶ Частота сердечных сокращений [Признак]
 - ▶ Фракция выброса левого желудочка [Признак]
 - ▶ Креатинин крови [Признак]
 - ▶ Объяснение результата
 - ▼ Вычисление рисков
 - ▼ Результат работы сервиса [Результат работы сервиса]
 - ▼ Обращение к внешнему агенту
 - ▶ Суммарный кардиоваскулярный риск по шкале SCORE [Риск]
 - ▶ Кардиоваскулярный риск по МЛР [Риск]
 - ◇ Пол Ж [знач (сорт: строковое)]
 - ◇ Возраст 52 [знач (сорт: строковое)]
 - ◇ Систолическое артериальное давление 150.0 [знач (сорт: строковое)]
 - ◇ Курение 1 [знач (сорт: строковое)]
 - ◇ Холестерин 5.5 [знач (сорт: строковое)]
 - ◇ С-реактивный белок 10.56 [знач (сорт: строковое)]
 - ◇ Глюкоза крови 6.8 [знач (сорт: строковое)]
 - ◇ Мозговой натрийуретический пропептид 177.4 [знач (сорт: строковое)]
 - ◇ Частота сердечных сокращений 86.0 [знач (сорт: строковое)]
 - ◇ **** Результат 10,35% Высокий риск [знач (сорт: строковое)]
 - ▶ Вызов процедуры
 - ▼ Вычисление по формуле заданной декларативно
 - ▶ Индекс массы тела (ИМТ) [Риск]
 - ▼ Индекс TIMI [Риск]
 - ◇ **** Результат 15,50 [знач (сорт: строковое)]
 - ◇ Возраст = 52.0, [знач (сорт: строковое)]
 - ◇ Систолическое артериальное давление = 150.0, [знач (сорт: строковое)]
 - ◇ Частота сердечных сокращений = 86.0, [знач (сорт: строковое)]
 - ▼ EuroSCORE II [Риск]
 - ◇ **** Результат Низкий риск [знач (сорт: строковое)]
 - ◇ Возраст = 52.0, [знач (сорт: строковое)]
 - ◇ Пол = Ж, [знач (сорт: строковое)]
 - ◇ Фракция выброса левого желудочка = 33.0, [знач (сорт: строковое)]
 - ◇ Нарушение функции почек = Нет, [знач (сорт: строковое)]
 - ◇ Заболевания периферических сосудов = Нет, [знач (сорт: строковое)]

Рис. 2. Скриншот фрагмента отчета о вычислении рисков

Литература

1. Мун С.А., Глушов А.Н., Штернис Т.А., Ларин С.А., Максимов С.А. Регрессионный анализ в медико-биологических исследованиях. – Кемерово: КемГМА. 2012.
2. Kolek M.J, Muehlschlegel J.D, Bush W.S, et al. Genetic and clinical risk prediction model for postoperative atrial fibrillation // *Circ Arrhythm Electrophysiol* – 2015; – №8(1). – P. 25-31.
3. Inohara T, Kohsaka S, Yamaji K, et al. Risk stratification model for in-hospital death in patients undergoing percutaneous coronary intervention: a nationwide retrospective cohort study in Japan. // *BMJ Open*. – 2019. – №9(5):e026683.
4. Shahian MD, Jeffrey P, Vinay B, et al. STS 2018 ADULT CARDIAC RISK MODELS PART 1. *Ann Thorac Surg*. – 2018. – №105:1411-8.
5. Бродская Т.А., Невзорова В.А., Шахгельдян К.И., Гельцер Б.И., Вражнов Д.А., Кистенев Ю.В. Оценка предиктивного потенциала факторов сердечно-сосудистого риска и их ассоциаций с жесткостью артерий у лиц европейской и корейской этнических групп // *Российский кардиологический журнал*. – 2021; – №26(5): 4230.
6. Гельцер Б.И., Шахгельдян К.И., Рублев В.Ю., Котельников В.Н., Кригер А.Б., Широкобов В.Г. Методы машинного обучения в прогнозировании летальных исходов в стационаре у больных ишемической болезнью сердца после коронарного шунтирования// *Кардиология*. – 2020, – №10 – С. 38-46.
7. Isabel [Электронный ресурс]: The Symptom Checker doctors use and trust https://symptomchecker.isabelhealthcare.com/suggest_diagnoses_advanced/landing_page
8. Symptomate [Электронный ресурс]: Diagnostic technology is powered by Infermedica. – Режим доступа: <https://symptomate.com/diagnosis/#0-66>.
9. Грибова В.В., Петряева М.В., Шалфеева Е.А. Облачный сервис поддержки принятия решений в кардиологии на основе формализованных знаний // *Сибирский журнал клинической и экспериментальной медицины*. 2020. – №4. – С. 32-38.
10. Гельцер Б.И., Котельников В.Н., Шахгельдян К.И., Курпатов И.Г. Результаты моделирования должных величин силы дыхательных мышц на основе метода искусственных нейронных сетей // *Российский физиологический журнал им. И.М.Сеченова*. – 2018; – 104 (9), – С. 1065–1074.
11. Gartner: 12 главных стратегических трендов в ИТ в 2022 году. 12.01.2022. <https://www.itcnews.ru/news/detail.php?ID=159453>
12. Грибова В.В., Клещев А.С., Москаленко Ф.М., Тимченко В.А., Федорищев Л.А., Шалфеева Е.А. Облачная платформа IASaaS для разработки оболочек интеллектуальных сервисов: состояние и перспективы развития // *Программные продукты и системы*. – 2018. – Т. 31. – № 3. – С.527-536.

РАЗРАБОТКА ПОДХОДА К ИЗВЛЕЧЕНИЮ И КЛАСТЕРИЗАЦИИ БИБЛИОГРАФИЧЕСКИХ ДАНЫХ СИСТЕМЫ ELIBRARY.RU

*Дырночкин А.А., ассистент
e-mail: dyrno4kin@gmail.com*

*Мошкин В.С., к.т.н., доцент
e-mail: v.moshkin@ulstu.ru*

Ульяновский государственный технический университет

1. ВВЕДЕНИЕ

При формировании рейтингов научной активности или подборе научных групп по определенной тематике, анализ наукометрических данных является важным аспектом.

Данные о научных публикациях хранятся в цифровых библиографических базах данных. Такие базы помогают отслеживать цитируемость статей, опубликованных в научных изданиях. Также они являются одним из источников получения наукометрических данных, необходимых при проведении различных оценочных исследований [1].

В данной статье представлено описание системы извлечения библиографической информации для последующего наукометрического анализа публикаций. Предложен подход к формированию научных групп по заданной тематике с использованием кластеризации статей. В статье также представлены результаты экспериментов с данными из научной библиотеки elibrary, подтверждающие эффективность предложенных моделей и алгоритмов.

2. АЛГОРИТМ ИЗВЛЕЧЕНИЯ БИБЛИОГРАФИЧЕСКОЙ ИНФОРМАЦИИ

Алгоритм загрузки неструктурированных данных с сайта elibrary состоит из следующих этапов:

1. На первом этапе были выгружены все научные идентификаторы сотрудников УлГТУ из внутренней учетной системы университета в формате JSON. Всего было выгружено порядка 200 сотрудников.

2. На втором этапе происходит загрузка статей всех авторов в формате: ID, Название статьи. Загрузка статей происходит с помощью java библиотеки JSOUP. На рисунке 1 показан пример загруженных статей.

| |
|---|
| 47092024,FUZZY RATING SYSTEM FOR COHESIVE DEVELOPMENT TEAMS |
| 47085567,APPLICATION OF NEURAL NETWORKS IN SENTIMENT ANALYSIS OF SOCIAL MEDIA TEXT DATA |
| 47056425,AN APPROACH TO THE PSYCHOLINGUISTIC ANALYSIS OF SOCIAL MEDIA TEXTS USING THE BIG FIVE PERSONALITY TRAITS |
| 44762013,ПРОГРАММНАЯ СИСТЕМА МАППИНГА ПРОФИЛЕЙ ПОЛЬЗОВАТЕЛЕЙ РАЗНЫХ СОЦИАЛЬНЫХ СЕТЕЙ |
| 44762014,ИНТЕЛЛЕКТУАЛЬНАЯ СИСТЕМА ДЛЯ OPINION MINING СОЦИАЛЬНЫХ МЕДИА |
| 45821682,ПРОГРАММНАЯ СИСТЕМА КЛАСТЕРИЗАЦИИ КОРОТКИХ ТЕКСТОВ |
| 46337292,ПРЕДИКТИВНАЯ АНАЛИТИКА НА ОСНОВЕ НЕЧЕТКИХ ВРЕМЕННЫХ РЯДОВ |
| 46337307,РАЗРАБОТКА СИСТЕМЫ НАХОЖДЕНИЯ ТЕКСТОВ ЭКСТРЕМИСТСКОЙ НАПРАВЛЕННОСТИ |
| 46677556,СРАВНЕНИЕ ПРИМЕНЕНИЯ РАЗЛИЧНЫХ НЕЙРОННЫХ СЕТЕЙ В ЗАДАЧАХ СЕНТИМЕНТ-АНАЛИЗА ДАННЫХ СОЦИАЛЬНЫХ СЕТЕЙ |
| 46677557,ОБЩИЙ АЛГОРИТМ ВЫЧИСЛЕНИЯ АГРЕГИРОВАННОГО ПРОГНОЗА ВРЕМЕННЫХ РЯДОВ |
| 47052758,DATA EXTRACTION OF CHARTS WITH HYBRID DEEP LEARNING MODEL |
| 47247677,НЕЧЁТКАЯ СИСТЕМА РЕЙТИНГА КОМПЛЕМЕНТАРНЫХ ПРОЕКТНЫХ КОМАНД |
| 45452543,TWO PHASE APPROACH TO DETECTION OF SOFTWARE PROJECTS WITH SIMILAR ARCHITECTURE BASED ON CLUSTERING AND ONTOLOGICAL METHODS |
| 45413213,DEVELOPMENT OF A SYSTEM BASED ON A FUZZY KNOWLEDGE BASE FILLED FROM VARIOUS FORMATS DOCUMENTS |
| 46026567,THE EXTENDING THE KNOWLEDGE BASE OF THE INTELLIGENT CAD OF A DESIGN ORGANIZATION USING SEMANTIC ANALYSIS OF WIKI-RESOURCES |
| 45094866,APPROACH TO THE USE OF LANGUAGE MODELS BERT AND WORD2VEC IN SENTIMENT ANALYSIS OF SOCIAL NETWORK TEXTS |

Рис. 1. Загруженные статьи

На следующем этапе производится загрузка полной информации по статьям. Для каждой статьи по ссылке <https://elibrary.ru/item.asp?id=> + *articleId* загружается документ в формате html, после чего производился его парсинг и извлечение основных данных по статье. На рисунке 2 показан пример загруженной статьи в формате JSON.

При большом количестве запросов к сайту elibrary появилась проблема блокировки сервером elibrary. После обхода 100 страниц происходила блокировка IP адреса, с которого работал парсер. Для решения этой задачи потребовалось оптимизировать скорость и частоту запросов таким образом, чтобы избежать блокировок.

В результате экспериментов было подобрано оптимальное время задержки между запросами к серверу. В итоге пауза между запросами составила от 3 до 35 секунд, в результате чего имитировалась работа обычного пользователя.

Также, во избежание блокировок использовался динамический подход для формирования данных User Agent. User agent — это строка с характеристиками устройства, по которым сервера могут определить тип браузера, операционную систему и производителя[5].

3. ПРЕДОБРАБОТКА СТАТЕЙ

Предобработка текста – это один из важнейших этапов интеллектуального анализа текста. Предобработка текста переводит текст на естественном языке в формат удобный для дальнейшей

работы. Предобработка состоит из различных этапов, которые могут отличаться в зависимости от задачи и реализации [3].

```
InCoreRisc:      false
InScopus:       false
InWOS:          false
rubricOecd:     "Computer and information sciences"
rubricGrnti:    "Кибернетика / Искусственный интеллект"
year:           "2014"
fullText:      null
inCoreRisc:    false
inWOS:         false
inScopus:      false
Id:            0
ArticleId:     "21716969"
▼ ArticleName: "ПРИМЕНЕНИЕ ОНТОЛОГИЧЕСКОГО ПОДХОДА К АНАЛИЗУ СОСТОЯНИЯ ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ"
▼ ArticleAbstract: "Описан семантический подход резюмирования временных рядов на примере показателей состо-
  модель OWL-онтологии рассматриваемой предметной области, онтологическая модель представ-
  в процессе оценки её состояния при искусственном повышении трафика. Решена задача объедине-
  онтологическую модель с использованием SWRL-правил. Рассмотрена реализация данного алго-
  Keywords:
  ▼ 0:
    keywordId: "942073"
    getKeywordName: "ОНТОЛОГИЯ"
  ▶ 1: [-]
  ▶ 2: [-]
  ▶ 3: [-]
  ▶ 4: [-]
  ▶ 5: [-]
  ▶ 6: [-]
  ▶ 7: [-]
  ▼ Authors:
    0: "10358"
    1: "762084"
```

Рис. 2. Пример загруженной статьи в формате JSON

Процесс предобработки текстовых статей представлен на рисунке 3.

Для перевода статей на русский язык, используется модуль, написанный на языке Google Apps Script, который использует стандартный модуль перевода LanguageApp от Google. Данный скрипт развернут на сервере Google и вызывается через XMLHttpRequest.

4. АЛГОРИТМ ВЕКТОРИЗАЦИИ ТЕКСТОВ

Векторизация текста – это процесс конвертации текста в числа по определенному алгоритму. Для векторизации предобработанных статей использовался метод TF-IDF [2].

TF-IDF – мера оценки важности слова в контексте документа, являющегося частью коллекции документов. Вес некоторого слова пропорционален частоте употребления этого слова в документе и обратно пропорционален частоте употребления слова во всех документах коллекции [6].

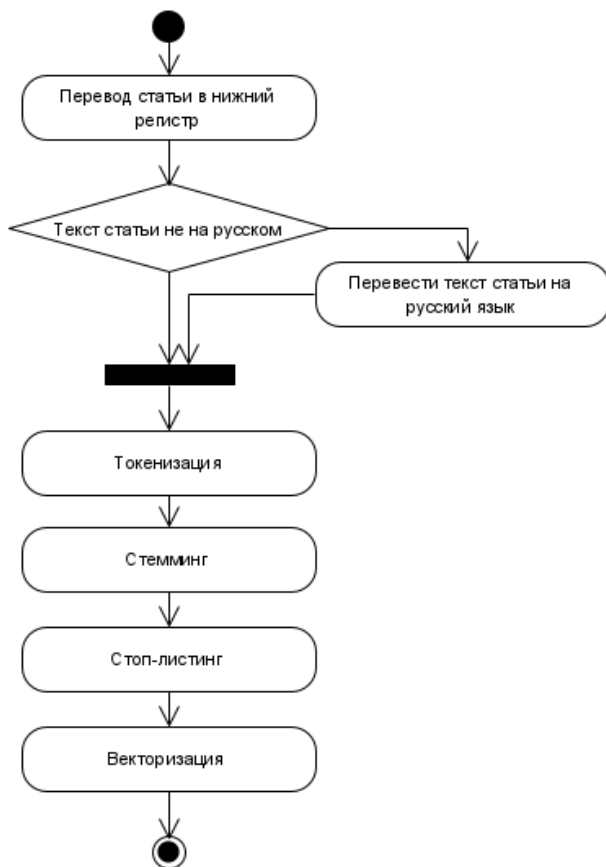


Рис. 3. Схема предобработки статей

Частота слова TF рассчитывается по формуле 1.

$$tf(t, d) = \frac{n_t}{\sum_k n_k} \quad (1)$$

где n_i – число вхождений слова в документ, а в знаменателе – общее число слов в данном документе.

IDF (обратная частота документа) – инверсия частоты, с которой слово встречается в документах корпуса. Учёт IDF уменьшает вес часто употребляемых слов [7]. Для каждого слова в пределах одной коллекции документов существует только одно значение IDF. Обратная частота рассчитывается по формуле 2.

$$idf(t, D) = \log \frac{|D|}{|d_i \ni t_i|} \quad (2)$$

где $|D|$ – количество документов в корпусе;

$|d_i \ni t_i|$ – количество документов, в которых встречается слово

Таким образом, TF-IDF вычисляется по формуле 3

$$tf - idf(t, d, D) = tf(t, d) \times idf(t, D) \quad (3)$$

5. ОПИСАНИЕ АЛГОРИТМА КЛАСТЕРИЗАЦИИ СТАТЕЙ

В данной работе был выбран метод кластеризации k -средних. Метод k -средних используется для кластеризации данных на основе алгоритма разбиения векторного пространства на заранее определенное число кластеров k , чтобы минимизировать суммарное квадратичное отклонение точек кластеров от центроидов этих кластеров [8]. Минимальное суммарное отклонение рассчитывается по формуле 4.

$$\min \left[\sum_{i=1}^k \sum_{x^{(j)} \in S_i} \|x^{(j)} - u_i\|^2 \right] \quad (4)$$

где u_i – центроид для кластера S_i

Для нахождения ближайшего кластера, к которому принадлежит документ, вычисляется минимальное косинусное расстояние между вектором документа и центром каждого кластера. Косинусное расстояние вычисляется по формуле 5[9].

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (5)$$

где

$\sum_{i=1}^n A_i * B_i$ произведение двух векторов

$\sqrt{\sum_{i=1}^n (A_i)^2}$ норма вектора A

$\sqrt{\sum_{i=1}^n (B_i)^2}$ норма вектора B

Остановка алгоритма производится тогда, когда границы кластеров не перестанут изменяться от итерации к итерации. Центроида каждого кластера рассчитывается по формуле 6.

$$\mu_i = \frac{1}{s_j} \sum A_j \quad (6)$$

где

s_j количество документов в кластере

$\sum A_j$ сумма всех векторов документов, принадлежащих данному кластеру

6. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Для проведения кластеризации была загружена библиографическая информация по сотрудникам УЛГТУ, всего было загружено порядка 14 тысяч статей в формате: название статьи и уникальный идентификатор, и в дальнейшем было загружено полное описание 1000 статей для проведения экспериментов.

В результате кластеризации корпус статей был разделен на 12 кластеров. На рисунке 6 представлен результат кластеризации и вывода рекомендаций по формированию научной группы.

Cluster no. 2 has 17 documents.

23820554 ПРИМЕНЕНИЕ АЛГОРИТМА ЛОГИЧЕСКОГО ВЫВОДА НА ОСНОВЕ FUZZYOWL-ОНТОЛОГИИ [10358, 762084]
 23886412 ИСПОЛЬЗОВАНИЕ СЕМАНТИЧЕСКОЙ МЕТРИКИ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ИЗВЛЕЧЕНИЯ ТЕРМИНОЛОГИИ ИЗ ТЕКСТА ПРОБЛЕМНОЙ ОБЛАСТИ
 24250264 ВЗАИМОДЕЙСТВИЕ ОНТОЛОГИЧЕСКОГО АНАЛИЗА И МЕХАНИЗМОВ ЛОГИЧЕСКОГО ВЫВОДА [762084, 10358]
 26270695 СЕМАНТИЧЕСКАЯ МЕТРИКА ""ТЕРМИН/НЕ ТЕРМИН"" НА ОСНОВЕ ОНТОЛОГИИ ПРОБЛЕМНОЙ ОБЛАСТИ" [762084, 842148, 843847]
 27076426 ГИБРИДИЗАЦИЯ НЕЧЕТКОГО ВЫВОДА И САМООБУЧАЮЩЕГОСЯ СЕМАНТИЧЕСКОГО АНАЛИЗА ДАННЫХ НА ОСНОВЕ НЕЧЕТКОЙ ОНТОЛОГИИ
 29267065 ПОСТРОЕНИЕ ОНТОЛОГИИ ПРОБЛЕМНОЙ ОБЛАСТИ НА ОСНОВЕ СИНТАГМАТИЧЕСКОГО АНАЛИЗА ТЕКСТОВЫХ ДОКУМЕНТОВ [325284, 93
 30808307 МОДИФИЦИРОВАННЫЙ МЕТОД ЛОГИЧЕСКОГО ВЫВОДА ЗНАНИЙ НА ОСНОВЕ НЕЧЕТКОЙ ОНТОЛОГИИ И БАЗЫ ПРЕЦЕДЕНТОВ [762084, 10
 32262706 РАЗРАБОТКА НЕЧЕТКОЙ БАЗЫ ЗНАНИЙ НА ОСНОВЕ АНАЛИЗА КОНТЕКСТОВ ПРОБЛЕМНОЙ ОБЛАСТИ [708454, 762084, 10358, 8120
 32323140 ПРИМЕНЕНИЕ НЕЧЕТКОЙ БАЗЫ ЗНАНИЙ ПРОБЛЕМНОЙ ОБЛАСТИ В ЗАДАЧЕ ПОИСКА АРХИТЕКТУРНО ПОДОБНЫХ ПРОГРАММНЫХ ПРОЕКТ
 32704420 РАЗРАБОТКА НЕЧЕТКОЙ БАЗЫ ЗНАНИЙ КАК ОСНОВА ПЛАТФОРМЫ АТНЕНЕ [708454, 762084, 10358]
 34895038 РАЗРАБОТКА БАЗЫ НЕЧЕТКИХ ЗНАНИЙ НА ОСНОВЕ КОНТЕКСТНОГО АНАЛИЗА ПРОБЛЕМНОЙ ОБЛАСТИ [10358, 708454, 762084]

Тематика: Нечеткий логический вывод и базы знаний принадлежит к кластеру 2

Авторы в данном классе: [10358, 762084, 843847, 842148, 325284, 933801, 708454, 812005, 316370]

Друшкина Н. Г., Мошкин В. С., Гуськов Г. Ю., Филиппов А. А., Андреев И. А., Зарубин А. А., Коваль А. Р., Дьяков И. В.

Рис. 4. Результаты кластеризации

7. ЗАКЛЮЧЕНИЕ

В результате исследования была представлена система, которая позволяет извлекать библиографическую информацию по авторам из системы elibrary.

Был описан алгоритм предобработки и векторизации полученных статей. Также представлен механизм кластеризации

научных статей и формирования рекомендаций при составлении научных групп по заданной тематике.

Литература

1. Николаев А.В., Жуков В.В. Тематическая кластеризация научной литературы // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем. – 2021. – С. 268-273.
2. Юферев В.И., Разин Н.А. Векторизация текстов на основе word-embedding моделей с использованием кластеризации // Моделирование и анализ информационных систем. – 2021. – Т. 28. – №. 3. – С. 292-311.
3. Кравченко Ю. А., Мансур А. М., Хуссайн М. Ж. Векторизация текста с использованием методов интеллектуального анализа данных // Известия Южного федерального университета. Технические науки. – 2021. – №. 2 (219). – С. 154-167.
4. Alam M. A Review on Clustering of Web Search Result. / M. Alam, K. Sadaf // Advances in Computing and Information Technology. Advances in Intelligent Systems and Computing, Springer, Berlin, Heidelberg. / Meghanathan N., Nagamalai D., Chaki N. (eds) - 2013. – Vol. 177.
5. Трубников В. С., Туральчук К. А. Проектирование системы сбора, анализа и визуализации наукометрических данных // Проблемы современной науки и образования. – 2015. – №. 6 (36).
6. Низомудинов Б. А., Тропников А. С. Автоматизированный сбор данных для наукометрического анализа // Научный сервис в сети Интернет. – Федеральное государственное учреждение "Федеральный исследовательский центр Институт прикладной математики им. МВ Келдыша Российской академии наук", 2019. – Т. 21. – С. 523-531.
7. Пархоменко П. А., Григорьев А. А., Астраханцев Н. А. Обзор и экспериментальное сравнение методов кластеризации текстов // Труды Института системного программирования РАН. – 2017. – Т. 29. – №. 2. – С. 161-200.
8. Мусаев А. А. и др. Обзор современных технологий извлечения знаний из текстовых сообщений // COMPUTER. – 2021. – Т. 13. – №. 6. – С. 1291-1315.
9. Yarushkina, N., Moshkin, V., Filippov, A. Development of a knowledge base based on context analysis of external information resources // DS-ITNT 2018// Proceedings of the International conference Information Technology and Nanotechnology. Session Data Science // Samara, Russia, 24-27 April, 2018. pp. 328-337

СИМВОЛИЧЕСКАЯ ДИНАМИКА И ФОРМИРОВАНИЕ КОГНИТИВНЫХ ПРОЦЕССОВ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

*Каганов Ю.Т., к.т.н., доцент
МГТУ им. Н.Э. Баумана
e-mail: yurijkaganov@gmail.com*

1. ВВЕДЕНИЕ

В 70-х годах XX века сформировалась новая парадигма науки, которая оформилась в научную концепцию, основанную на понятии сложности систем. Эта концепция базировалась на идеях самоорганизации сложных нелинейных динамических систем [1, 2, 3, 4, 5]. Одна из основных проблем теории самоорганизации сложных систем является проблема формирования информации и возникновения смысла [6, 7,]. Такая постановка проблемы переводит ее в плоскость, связанную с семиотикой [8]. Возможность такого перехода обусловлена развитием методов символической динамики, которая является мощным инструментом анализа сложных нелинейных динамических систем [9, 10, 11]. Дальнейшее развитие этого подхода было связано с использованием концепции метаграфов и гранулярных вычислений [12, 13, 14]. Это позволило выйти на анализ динамических процессов, которые предположительно определяют деятельность мозга и возникновение сознания [15, 16]. Дальнейшие исследования в этом направлении могут дать новый импульс теории искусственного интеллекта.

2. ОСНОВНЫЕ ПОНЯТИЯ И МОДЕЛИ СЛОЖНЫХ СИСТЕМ

Сложная система представляет собой многоуровневую систему взаимодействующих динамических подсистем, каждая из которых обладает определенной автономией и при этом целостная система не сводима к совокупности ее подсистем. Такую систему можно представить как систему автономных агентов, как правило обладающую некоторой целесообразностью поведения. Целесообразность поведения сложной системы проявляется в способности системы сохранять свою качественную определенность.

Это свойство определяет главную цель системы, которая, в свою очередь формирует частные цели подсистем.

Основными особенностями организации сложной системы являются следующие:

- Холоническая (холархическая) организация структур сложной системы и относительная автономность каждого уровня соответствующей структуры. При этом каждый уровень системы, будучи автономным, проявляет свойства части целого.
- Сложная нелинейная динамика взаимодействия подсистем на каждом однородном структурном уровне.
- Эмерджентность и появление новых свойств подсистем на каждом более высоком уровне.

3. ДИНАМИЧЕСКАЯ СИСТЕМА

Очень важной характеристикой сложных систем является их динамика поведения. В связи с этим основные понятия, связанные с нелинейной динамической системой, можно определить следующими образом:

1. **Аттрактор. (Притягивающее множество)**

Аттрактором называется математически предельное множество (особая точка, цикл, тор и т.д.), для которого множество всех траекторий, асимптотически приближающихся к этому множеству, открыто.

2. **Бифуркация. (Разветвление решения динамического уравнения)**

С изменением параметров динамической системы аттракторы могут меняться. При этом в результате изменений некоторых параметров могут произойти качественные изменения и их существенная перестройка. Например: устойчивый фокус может смениться предельным циклом. Такие значения параметров носят название **бифуркационных**, а сама перестройка **бифуркацией**.

3. **Устойчивость и граница между регулярной и хаотической динамикой**

Критерием границы между регулярной динамикой и динамическим хаосом является устойчивость системы к малым возмущениям. Такой подход приводит к необходимости определения хаотического поведения через чувствительную зависимость системы к начальным условиям и использованию показателей Ляпунова и энтропии в качестве критериев динамического хаоса.

Динамическая система может быть описана либо системой обыкновенных дифференциальных уравнений. В этом случае процесс, описываемый такой системой называется *поток*ом.

$$\frac{dx_i}{dt} = f_i(x_1, x_2, \dots, x_N; \mu_1, \mu_2, \dots, \mu_k). \quad (1)$$

Либо системой дискретных отображений:

$$x_{n+1}^i = f_i(x_n^1, x_n^2, \dots, x_n^N; \mu_1, \dots, \mu_k). \quad (2)$$

Такой процесс называется *каскадом*.

При этом $x_i(t)$ или x_n^i – переменные, определяющие состояние системы (фазовые координаты динамической системы);

$f_i(x, \mu)$ – в общем случае нелинейные функции;

μ – параметры динамической системы.

В соответствии с теоремой Коши решение динамической системы существует, единственно для данных начальных условий $(x_i(0), x_0^i)$ и гладко зависит от изменения начального состояния.

Для исследования динамических систем чрезвычайно эффективным оказался математический аппарат, предложенный в конце XIX века великим математиком и мыслителем А. Пуанкаре. Его идея состояла в использовании так называемых функций последования для исследования фазовых траекторий движения динамической системы. Прежде всего с помощью этого аппарата появилась возможность исследовать динамические системы с точки зрения существования аттракторов.

Функция последования используется для исследования замкнутых траекторий. Множество точек S является отображением последовательности точек фазовой траектории динамической системы в N -мерном пространстве при пересечении $N-1$ мерного многообразия гиперповерхности Пуанкаре в процессе её развития во времени. Для предельного цикла множество пересечений сходится к точке.

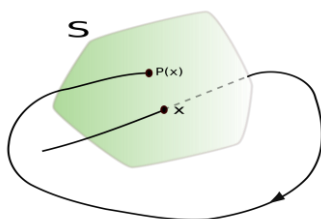


Рис.1 Функция последования

Исследование динамики диссипативных систем с помощью функции последования позволяет выявить бифуркационные процессы перехода от одного аттрактора к другому. В зависимости от вида

аттрактора в динамической системе может быть реализован либо регулярный режим движения, либо хаотический, характерный для странных аттракторов. Рассмотрим этот процесс на примере динамической системы, описываемой обыкновенными дифференциальными уравнениями

$$\frac{dx_i}{dt} = f_i(x_1, x_2, \dots, x_N; \mu_1, \mu_2, \dots, \mu_k). \quad (3)$$

Или, в общем виде $\dot{x} = f(x, \mu)$ [9, 10] Управляющий параметр μ определяет возможность перехода от одного аттрактора к другому.

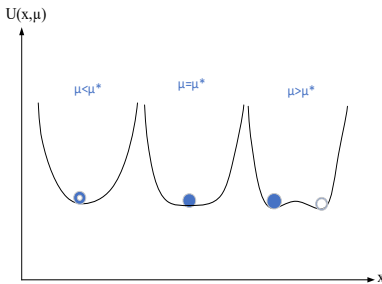


Рис. 2. Мягкая бифуркация

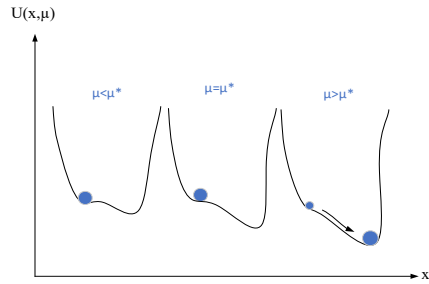


Рис. 3. Жесткая бифуркация

(Бифуркация $\frac{dx}{dt} = f(x, \mu)$, $f(x, \mu) = -\nabla_x U(x, \mu)$, μ^* - критическое значение параметра).

Каждый из аттракторов может интерпретироваться как определенный элемент информации. Поэтому при изменении вектора параметров μ возможен переход из одного аттрактора в другой (т.е. бифуркация). При этом возможны два варианта бифуркации. В случае мягкой бифуркации (рис. 2) возможно возникновение двух равноправных аттрактора, что приводит к амбивалентности. В случае же жесткой бифуркации возникает новый аттрактор (рис. 3). Тем самым возможно получение новых элементов информации. В частности, в качестве примера одного из странных аттракторов, к которому при изменении параметра μ приходит динамическая система, состоящая всего лишь из трех нелинейных ОДУ, можно привести аттрактор Э. Лоренца.

Уравнения Э. Лоренца:

$$\begin{aligned} \dot{x} &= \sigma(y - x); \\ \dot{y} &= x(r - z) - y; \\ \dot{z} &= xy - bz. \end{aligned} \quad (4)$$

При значениях параметров (вектор μ):

$$\sigma = 10; r = 28; b = 8/3$$

система уравнений описывает странный аттрактор, обладающий двумя основными свойствами:

- 1) Фрактальной структурой фазового пространства.
- 2) Стохастичностью и непредсказуемостью поведения фазовой траектории.

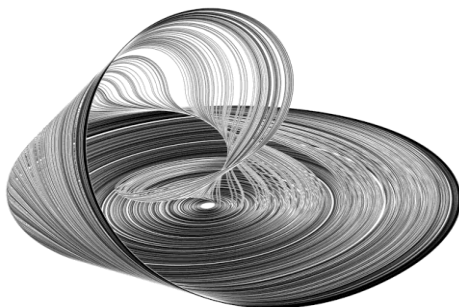


Рис. 4. Аттрактор Лоренца

4. СИМВОЛИЧЕСКАЯ ДИНАМИКА

Дальнейшее развитие исследований в области нелинейной динамики привели к формированию инструментария, с помощью которого появилась возможность более детально исследовать поведение сложных динамических систем. Таким инструментарием оказались методы символической динамики [11]. В 1935 году Г. Биркгоф впервые применил символическую динамику для кодировки траекторий вблизи гомоклинической орбиты. С. Смейл применил ту же технику при построении так называемой “подковы” — простой модели хаотической динамики. “Подкова Смейла” оказала существенное влияние на теорию хаоса, так как этот пример является типичным, а методы символической динамики оказались тем инструментом, который позволяет описать природу детерминированного хаоса [12].

- Основная идея метода — описание динамики системы при помощи допустимых последовательностей символов (допустимых слов) из конечного набора символов (алфавита).
- Множество возможных состояний системы (фазовое пространство) разбивается на конечное число ячеек. Каждая ячейка соответствует “символу” при этом в каждый момент времени появляется тот символ, который соответствует текущему состоянию системы.

Таким образом, фазовая траектория может быть представлена как система символов, а аттрактор – как некое слово в соответствующей семиотической системе.

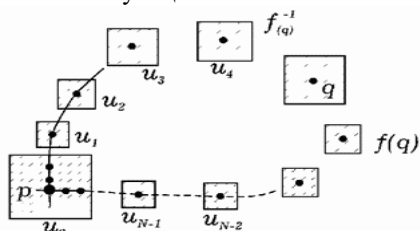


Рис. 5. Последовательность отображения Пуанкаре

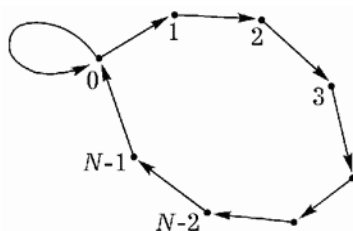


Рис. 6. Графовое отображение символической динамики

В работах Осипенко Г.С. [13, 14] показано каким образом возможен переход от символического образа к нагруженному графовому отображению. Этот подход позволяет перейти от анализа характера аттракторов на основе показателей Ляпунова к дискретному графовому представлению сложной динамической системы. Основная идея этого подхода состоит в следующем. Фазовое пространство динамической системы разбивается на ячейки M_i , устанавливается изоморфизм между ячейками фазового пространства и вершинами v_i плоского мультиграфа $G = \langle V, E \rangle, v_i \in V, \{e_{jk}\} \in E$, где вершина v_i принадлежит множеству вершин V , а подмножество ребер $\{e_{jk}\}$ принадлежит множеству направленных ребер E . Вершины j и k связаны ориентированным ребром e_{jk} только в том случае, если траектория движения системы в фазовом пространстве предусматривает возможность перехода из ячейки M_j в ячейку M_k . Таким образом, в монографии [13] предлагается двухуровневая модель и задается изоморфизм между верхним и нижним уровнями. Нижнему уровню соответствует фазовое пространство динамической системы, а верхнему уровню – плоский мультиграф G , который содержит информацию о возможных переходах между ячейками фазового пространства динамической системы. Рассматриваемая динамическая система может быть иерархической. В настоящее время такие системы все чаще привлекают внимание исследователей. Для обозначения такого подхода используется термин «иерархическая символическая динамика» (англ. «hierarchical symbolic dynamic») [15].

Рассматриваемая динамическая система может быть основой для нескольких категориальных признаков, что требует использование нескольких графов или более сложной графовой модели, которая позволяет включить несколько графов. Построение графовой модели основано на следующих соображениях.

Гипотеза

Форма фазовой траектории адекватно отражает (вскрывает) ключевые внутренние свойства нелинейных динамических систем.

Определение

Под формой траектории динамической системы понимаются некие её инварианты, сохраняющиеся при однородных и гладких сдвигах и растяжениях. Кодирование формы траекторий динамических систем можно представить следующим образом.

Формирование символического образа динамической системы

Строится ориентированный граф $G_{\alpha\varphi} = (V_{\alpha\varphi}, E_{\alpha\varphi})$:

$V_{\alpha\varphi}$ – вершины графа $G_{\alpha\varphi}$ – символы состояний;

$E_{\alpha\varphi}$ – рёбра графа $G_{\alpha\varphi}$ – переходы между состояниями.

Утверждение

Символический $G_{\alpha\varphi}$ -образ отражает глобальную структуру динамической системы $\{f_k(x, \mu), k \in K \subset Z\}$ [13]. Существует соответствие между траекториями системы и путями на символическом $G_{\alpha\varphi}$ -образе.

Таким образом можно поставить в соответствие топологические свойства энергии, т.е. гамильтониана $H(q, p)$ символическому образу, представляющему ориентированный граф $G_{\alpha\varphi}$.

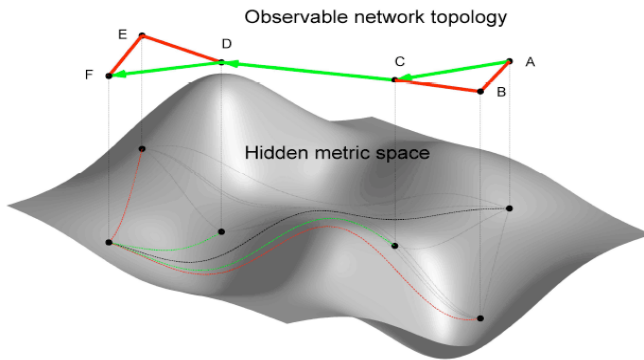


Рис. 7. Топология поверхности энергии

5. МЕТАГРАФОВЫЕ МОДЕЛИ И ИНФОРМАЦИЯ

Следующим этапом формализации является переход к метаграфовой модели, предполагающей возможность эмерджентности, которая позволяет описать возникновение новых уровней формирования информации. Метаграф – это холонически организованный граф.

$$MG = \langle V, MV, E \rangle, \quad (5)$$

Вершина метаграфа характеризуется множеством атрибутов:

$$v_i = \{atr_k\}, v_i \in V, \text{ где } v_i - \text{вершина метаграфа; } atr_k - \text{атрибут.}$$

Ребро метаграфа характеризуется множеством атрибутов, исходной и конечной вершиной:

$$e_i = \langle v_S, v_E, \{atr_k\} \rangle, e_i \in E,$$

где e_i – ребро метаграфа; v_S – исходная вершина (метавершина) ребра; v_E – конечная вершина (метавершина) ребра; atr_k – атрибут.

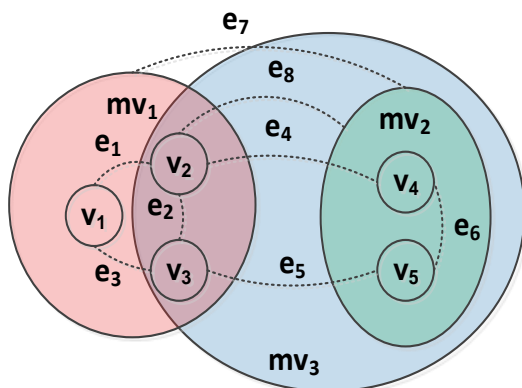


Рис.8. Модель метаграфа

Такое представление дает возможность рассмотрения динамических процессов с точки зрения формирования информации. По определению Д.С. Чернавского [7] информация есть запомненный выбор одного варианта из нескольких возможных и равноправных. При этом процесс запоминания можно интерпретировать как формирование аттракторов в процессе функционирования динамической системы. Количество информации, определяемой по К. Шеннону, можно получить исходя из знаменитой формулы для сообщения, содержащего N символов

$$I_N = -N \sum_{i=1}^M p_i \log_2 p_i \quad , \quad (6)$$

где M - число букв в алфавите, p_i - вероятность встречаемости i - го символа в сообщении.

В свою очередь информация тесно связана с целесообразностью поведения динамической системы. Такой целью может быть гомеостатическое состояние, которое позволяет системе сохранять свою качественную определенность при взаимодействии с внешней средой. То есть формирование глобального аттрактора, который определяет формирование аттракторов более низкого иерархического уровня. Информация, связанная с целесообразностью поведения, может быть оценена на основе введения семантической информации. Семантическая информация тесно связана с понятием цели. Впервые это понятие было введено А.А. Харкевичем, который связал ценность информации с целью деятельности, предложив рассматривать энтропийную меру Шеннона как меру вероятности попадания в цель, т.е. как меру целесообразности. В свою очередь Ю.А. Шрейдер предложил построить теорию семантической информации на основе концепции разнообразия, а не концепции снятия неопределенности, и в частности, на основе учета такого свойства информации как зависимость получаемой информации от априорной. Это может быть выражено следующим образом

$$V = \log_2 \frac{P}{q} \quad , \quad (7)$$

где P – вероятность достижения цели после получения информации, q – вероятность достижения цели после получения информации. V – ценность информации, зависящая от получаемой информацией, которая способствует достижению цели. Априорная информация q зависит от полного количества информации, т.е. $q = 2^I$. Апостериорная информация определяется тем, насколько она приближает к цели. Она может быть как больше, так и меньше q . В таком же смысле можно трактовать дивергенцию Кульбака-Лейблера $D_{KL}(q, p)$

$$D_{KL}(q, p) = \sum_i^N p_i \log \frac{p_i}{q_i} \quad . \quad (8)$$

Это соотношение часто используется как оценка ошибки в теории искусственных нейронных сетей. Целесообразность сложных динамических систем можно представить как иерархию целей. Особенно наглядно это проявляется в биологических системах. Можно сформулировать следующую систему иерархических уровней целесообразности.

- 1) Выживание вида. Формирование и фиксация целей на генетическом уровне.

- 2) Выживание организма (индивида, агента). Удовлетворение потребностей в питании, размножении и адаптации к внешней среде. Формирование соответствующих целей.
- 3) Сохранение гомеостаза и обучение как средство совершенствования взаимодействия с внешней средой.
- 4) Формирование частных целей, возникающих в процессе жизнедеятельности организма.

6. ЗАКЛЮЧЕНИЕ

С общей точки зрения теории динамических систем, такая иерархическая система с учетом ее целесообразности поведения может быть интерпретирована как семиотическая система. В наиболее развернутом виде теория семиотических систем как теория симеозиса, была развита в работах Ч.С. Пирса [20] и, особенно, в работах Ч.У. Морриса [21]. В контексте этих работ симеозис понимается как единство трех составляющих процесса формирования семиотической структуры, которое может быть выражено в виде отношения знаков к соответствующим объектам, внешней среды и к интерпретатору, в качестве которой выступает сама динамическая система:

- 1) Семантика – определяет отношение знака к тому объекту, который этот объект обозначает.
- 2) Синтактика – определяет взаимоотношения между знаками.
- 3) Прагматика – определяет каким образом данный объект может быть использован интерпретатором.

Аттракторы представляют собой устойчивые структуры, которые можно интерпретировать как слова, построенные на символах, составляющих их фазовое пространство. Их зависимость от воздействия внешней среды определяется изменением параметров динамических систем, что влечет за собой возникновение бифуркационных процессов и формирования новых аттракторов. Такой подход дает возможность понять суть процессов формирования информации и возникновения семантики. Иерархия целей, которые формируют динамические процессы, определяет иерархию различных видов и уровней информации. Цель представляет собой функционал, который определяет, насколько картина мира интеллектуальной системы отличается от реальной картины мира. Она может быть представлена в виде гиперповерхности, имеющей множество экстремумов. При этом она может меняться в соответствии с изменением окружающей среды. Аттрактор верхнего уровня формируется как устойчивое соответствие (минимум целевой функции) реального мира и картины мира интеллектуальной системы.

Поиск такого минимума может осуществляться на основе стохастического градиента, который позволяет учитывать не все переменные, а только их частичное множество. В связи с этим решается проблема возникновения семантики (т.е. смысла). Семантика возникает, когда воздействие внешней среды оценивается системой на основе некоторой шкалы, определяемой целью [19]. В результате формируется система аттракторов, которая определяет поведение системы. В свою очередь аттракторы формируют реакцию системы на внешние воздействия, что и характеризует её прагматику. Иерархическая структура аттракторов формирует в свою очередь систему уровней языков состоящих из различного типа символов (знаков). На каждом уровне имеется своя символика, определяемая той средой, в которую погружена система аттракторов и, соответственно той семантикой, определяемой этой средой. Взаимодействие между языками происходит путем формирования иерархии структур системы аттракторов. Структура аттракторов на самом верхнем уровне определяет характер будущих реакций, необходимых для адекватного ответа на воздействие внешней среды. В этом состоит смысл построения «КАРТИНЫ МИРА» у сложной нелинейной динамической системы. Это осуществляется на основе создания языковых структур верхнего уровня, отражающих с одной стороны наиболее важные для системы свойства окружающей среды, а с другой, позволяющих формировать необходимые реакции для «выживания» этой системы в окружающей среде. Понятие информации может быть связано с понятием гранулы, введенной Л. Заде в 1996 году. В этом случае мы можем рассматривать иерархическую (холоническую) организацию информационных структур как систему иерархии языков [22]. Такое представление позволяет представить сложную динамическую систему в терминах нечетких вычислений, где решающим является использование понятия «лингвистической переменной». При этом лингвистическая переменная выступает в трех ипостасях: 1) как знак, 2) как средство взаимодействия между знаками и 3) как средство использования этого знака для дальнейших действий.

Литература

1. Haken H. Synergetics. An Introduction. Nonequilibrium Phase Transitions and Self-Organization in Physics, Chemistry and Biology. – Springer, 1977. - 375 p.
2. Haken H. Advanced Synergetics. Instability Hierarchies of Self-Organizing Systems and devices. - Springer, 1983. 384 p.
3. Prigogine I. Introduction to Thermodynamics of Irreversible Processes. John Wiley, New York, 1967. 254 p.

4. Nicolis G., Prigogine I. Self-Organization in Nonequilibrium Systems. From Dissipative Structures to Order Through Fluctuations. - John Wiley, New York. 1977. 520 p.
5. Eigen M., Schuster P. The Hypercycle. A principle of Natural Self-Organization. - Springer, 1977. - 275 p.
6. Волькенштейн М.В. Энтропия и информация. – М.: Наука. 1986. – 192 с.
7. Чернавский Д.С. Синергетика и информация. Динамическая теория информации. - М.: Едиториал УРСС. 2004. – 288 с.
8. Haken, H.; Portugali, J. Information Adaptation: The Interplay between Shannonian and Semantic Information in Cognition; Springer: Berlin, Germany, 2015.
9. Майнцер К. Сложносистемное мышление. Материя, разум, человечество. Новый синтез. – М.: Книжный дом «ЛИБРОКОМ», 2009. – 464 с.
10. Лоскутов А.Ю., Михайлов А.С. Основы теории сложных систем. – М.-Ижевск: НИЦ РХД, 2007. – 620 с.
11. Williams S. Symbolic Dynamics And Its Applications. American Mathematical Society, 2002.
12. Боуэн Р. Методы символической динамики. - М.: Мир, 1979.
13. Osipenko G. Dynamical Systems, Graphs, And Algorithms. Springer-Verlag, 2007.
14. Осипенко Г.С. Оценка показателей Ляпунова методами символического анализа. Динамические системы, 2016, том 6(34), №1, 15–35.
15. Akintayo A., Sarkar S. Hierarchical Symbolic Dynamic Filtering of Streaming Non-stationary Time Series Data, 2017. <http://arxiv.org/pdf/1702.01811v1>
16. Johnson J. Hypernetworks in the Science of Complex Systems. Imperial College Press. London. 2013. – 330.
17. Basu, A., Blanning, R.: Metagraphs And Their Applications. Springer, New York, 2007.
18. Черненький В.М., Гапанюк Ю.Е., Ревунков Г.И., Терехов В.И., Каганов Ю.Т. Метаграфовый подход для описания гибридных интеллектуальных информационных систем // Прикладная информатика. 2017. № 3 (69), т. 12.
19. Nicolis J.S. Dynamics of Hierarchical Systems. An Evolutionary Approach. Springer Verlag, 1986.
20. Peirce Ch. S. Issues of Pragmatism. Harvard University Press. 1931 – 1958.
21. Morris Ch. W. Writings on the General Theory of Signs. Mouton and Co. Publishers 1971.
22. Tarasov V., Kaganov Y., Gapanjuk Y. The Metagraph Model for Complex Networks: Definition, Calculus, and Granulation Issues. In: Proceedings of the XIX International Conference on Data Analytics and Management in Data Intensive Domains (DAMDID/RCDL 2017), Moscow, Russia, pp. 342-349

РАЗРАБОТКА СИСТЕМЫ ПОДДЕРЖКИ ФОРМИРОВАНИЯ РЕКОМЕНДАЦИЙ НА ОСНОВЕ ОЦЕНКИ СЕРДЕЧНО-СОСУДИСТОГО ЗДОРОВЬЯ*

Платов П.В., аспирант

УлГТУ

e-mail: chipa.platov@gmail.com

1. ВВЕДЕНИЕ

В настоящее время активно проходит процесс цифровизации в области здравоохранения. Переводятся в машиночитаемый вид записи о приемах у врача, вызовах скорой медицинской помощи, результаты анализов. Это делается не только для того, чтобы не потерять медицинскую карту пациента, и врач всегда имел возможность посмотреть историю болезни при приёме пациента. Таким образом процесс вовлекается сам пациент. На данный момент, в научном сообществе это рассматривается как отдельная научная область - биомедицинская информатика (БМИ). *Биомедицинская информатика (Biomedical and health informatics)* - это междисциплинарная научная область, которая изучает и добивается эффективного использования биомедицинских данных, информации и знаний для научных исследований, решения проблем и принятия решений, мотивированных усилиями по улучшению здоровья человека [1].

Однако ещё недостаточно разработаны методы и модели для управления сердечно-сосудистым риском. Поэтому цель данной статьи - описание подхода и архитектурного решения данной проблемы.

2. ОБЗОР СУЩЕСТВУЮЩИХ МЕТОДОВ И МОДЕЛЕЙ МЕТОДОВ ПОДДЕРЖКИ ПАЦИЕНТА

На текущий момент выделяют несколько «горячих точек». Профессор Корбинский относит к ним такие проблемы как: извлечения и представления знаний, интерпретация и объяснение в технологиях МО, поэтапное ассистирование в принятии решений,

* Работа выполнена при финансовой поддержке РФФИ, проект №20-07-00672

персонализация пользователя СППР, погружение интеллектуальных СППР в информационные системы [2].

Система МТС 120/80 [3] упрощает мониторинг состояния пациентов с сердечно-сосудистыми заболеваниями: врач может дистанционно наблюдать актуальную картину самочувствия. Данная система работает как сервис, данные передаются пользователем, обеспечивается хранение показателей, в дальнейшем к ним имеет доступ врач. Аномальные значения показателей выявляются автоматически и оперативно передаются лечащему врачу. Ограничениями системы являются – малое количество показателей – данные передаются с личного тонометра, автоматизирован анализ аномальных значений, рекомендации выдаются лечащим врачом.

Проект СберЗдоровье [4] предоставляет различные медицинские услуги: онлайн-консультации, запись на прием, диагностику или комплексное обследование, вызов врача на дом, «Дневник здоровья» — дистанционный мониторинг пациентов с хроническими заболеваниями, круглосуточные онлайн-консультации с дежурными терапевтом или педиатром, запись к врачу. Ограничениями сервиса является отсутствие подсистемы анализа показателей здоровья с последующей выдачей пациент-ориентированных рекомендаций в области управления своим здоровьем.

Webiomed является первой российской системой ИИ для здравоохранения, успешно прошедшая независимые клинические испытания и зарегистрирована Росздравнадзором как программное медицинское изделие, а также первой ИИ-системой, подключенной к единой государственной информационной системе здравоохранения. Ведутся активные работы по применению автоматизированной СППВР при диспансеризации пациентов для оценки уровня сердечно-сосудистого риска [6]. Авторами этого проекта ведутся исследования в области совершенствования возможности оценки сердечно-сосудистого риска при помощи методов машинного обучения [5].

Авторы работы [7] использовали два метода для оценки риска развития СС событий. Один на основе Фрамингемской шкалы, второй на основе МО, в частности пятислойную нейронную сеть. В исследовании была использована база данных с 2 236 пациентами, которая была предварительно размечена минимум двумя специалистами. Архитектура этой системы включала в себя базу данных с размеченными показателями пациентов, два модуля определения риска развития СС событий и модуль с представлением

результатов. В результате метод на основе МО превзошел метод на основе правил Фрамингемской шкалы на 8,8%.

Авторы исследования [8] предлагают рассматривать рекомендательные системы с различных точек зрения – врача и пациента. При пациент-ориентированном подходе предлагается самому пациенту вести свою ЭМК, которая будет храниться в базе данных системы. Из этой базы данных ЭМК пациента будет подаваться на вход модулю выдачи рекомендаций по здоровью (СРЗ). Модуль СРЗ будет аккумулировать данные из различных источников данных по здоровью, а также справочников врача и классификации заболеваний.

3. ТРЕБОВАНИЯ К СИСТЕМЕ «КАРДИОАНАЛИТИК»

На основании проведенного анализа в предыдущем разделе, были сформулированы следующие требования к системе КардиоАналитик, которые представлены ниже.

3.1. ТРЕБОВАНИЯ К ДАННЫМ

Для подобного рода систем, характерной особенностью является наличие функциональных требований, предъявляемых к источникам, структурам, типам данных. К таким требованиям относится сбор данных, их преобразование, хранение в структурированной форме, а также получение агрегированных данных разного типа.

3.1.1. ТРЕБОВАНИЯ К ВХОДНЫМ ДАННЫМ

КардиоАналитик предполагает различные способы сбора данных. В первую очередь пациент сможет сам заполнять данные анализов, анкетные данные и данные инструментальных исследований вести свою ЭМК. Также будет доступен импорт данных с носимых устройств и интеграция с различными мобильными приложениями, отслеживающими различные физиологические показатели. КардиоАналитик будет иметь возможность интегрировать в себя, описанные выше данные из различных других источников данных с ЭМК пациента. Необходимым условием для анализа информации является интеграция и хранение данных о заболеваниях и медицинских справочников из различных источников.

Личные, анкетные и данные анализов представляют собой четко определенные структуры, которые хорошо ложатся на концепцию реляционных баз данных. Исходя из требований к входным данным становится понятно, что наиболее подходящий способ хранения

гетерогенных данных – реляционные базы данных для структурированных данных (ЭМК, персональные данные пациента, различные медицинские справочники, образцы клинических рекомендаций). Для данных медицинских исследований, таких как ЭКГ предлагается использовать файловые хранилища и хранить ссылки на такие исследования в реляционной базе данных.

3.1.2. ТРЕБОВАНИЯ К ВЫХОДНЫМ ДАННЫМ

Требования к выходным данным будут включать формат рекомендаций, который может содержать описательную часть, предиктивную часть, прескриптивную часть, содержащую цель и тактику достижения этой цели. А также различную графическую информацию, полученную при помощи описательного анализа. Такие данные удобнее всего будет хранить в реляционной базе данных. Если же понадобится хранить изображения или файлы других форматов, чтобы не генерировать их каждый раз, то предлагается воспользоваться файловым хранилищем и сохранять ссылки на эти файлы в таблицах с результатами реляционной базы данных.

Так же для хранения и обработки и деперсонализации персональных данных нужно следовать всем законам РФ [9, 10].

3.2. ТРЕБОВАНИЯ К ФУНКЦИЯМ СИСТЕМЫ

Основываясь на приведенном обзоре, система КардиоАналитик должна включать в себя функции регистрации пользователя, сбора, хранения, интеграции, данных, анализа данных как текущего состояния, так и определения тенденций изменения этого состояния. Деперсонализация персональных данных их защищенное хранение. Необходима функция выбора релевантной рекомендации по текущему состоянию и тенденциям его изменения, функция отслеживания тенденций результатов медицинских, инструментальных исследований, анкетных данных о физиологическом и психологическом состояниях. Необходима функция выдачи пациент-ориентированных рекомендаций и визуализации сгенерированных рекомендаций и тенденций.

3.3. ТРЕБОВАНИЯ К АРХИТЕКТУРЕ

На текущем уровне развития системы предполагается подход SOA(service oriented architecture) [11]. Таким образом обеспечиваются требования кроссплатформенности и доступности для большого числа пользователей системы КардиоАналитик. Система должна иметь возможность хранить и обрабатывать гетерогенные данные. Для

современных SOA систем выдвигаются также требования модульности и микросервисный подход.

3.3.1. ВАРИАНТЫ ИСПОЛЬЗОВАНИЯ СИСТЕМЫ

Основным пользователем системы КардиоАналитик будет пациент. Ему будут доступны такие варианты использования как регистрация в системе, вход в систему, заполнение личных данных. При заполнении личных данных будет обязательен процесс заполнения необходимых персональных данных для идентификации пациента. Далее, в зависимости от предпочтений пациента и наличия данных, он может самостоятельно заполнить данные о своих анализах, инструментальных исследованиях, анкетные данные или импортировать их из другой системы, при условии, что подобная интеграция реализована на стороне системы «КардиоАналитик». Также пациенту доступен вариант использования с проведением анализа своих показателей, в результате которого, должны быть получены рекомендации по управлению риском сердечно-сосудистых событий. Если пациент давно ведет свои данные в системе (есть как минимум два результата анализов), то становится доступен вариант использования с анализом тенденций поведения показателей пациента. Вариант использования с анализом показателей пациента включает в себя такие варианты как: анализ показателей на наличие хронических заболеваний, анализ показателей на наличие базовых факторов риска сердечно-сосудистых событий, анализ данных медицинских исследований пациента в области СС системы. Все варианты использования отражены на рис. 1.

3.3.2. АРХИТЕКТУРА СИСТЕМЫ

В самом начале использования системы пользователю требуется зарегистрировать личный аккаунт или же войти в него, если он имеется, для этого он должен подать соответствующие данные на вход модулю регистрации/входа. При регистрации система запросит у пользователя персональные данные для регистрации в системе. Эти данные будут сохранены в модуле хранения данных.

Пользователь может настроить параметры интеграции со сторонними системами, например, у него уже имеется своя ЭМК в другом сервисе, и он хочет импортировать данные из неё. Эти данные сохранятся в модуле хранения данных системы «КардиоАналитик», чтобы в дальнейшем использовать их при обращении API сторонних систем. Гетерогенные данные, полученные от API сторонней системы, будут поступать в модуль интеграции со сторонними системами, после

чего будут сохранены согласно модели данных системы «КардиоАналитик».

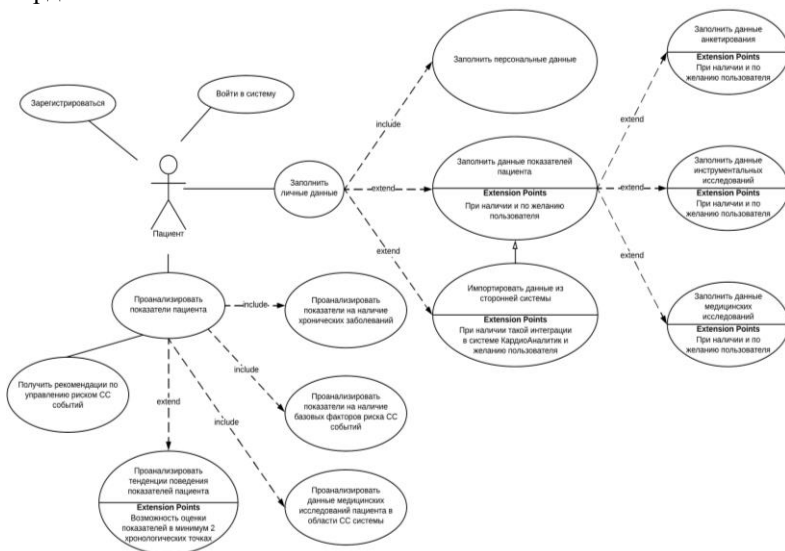


Рис.1. Диаграмма вариантов использования системы «КардиоАналитик»

Пользователь может вручную ввести свои данные в системе «КардиоАналитик». В этом случае ему придется самому заполнять данные всех медицинских анализов и исследований, проходить анкетирование. Эти данные будут сохранены в модуле хранения данных. Этот подход требует больше времени для конечного пользователя, однако, при нем данные сразу поступают в формате модели данных системы «КардиоАналитик» и готовы к употреблению модулем анализа данных, минуя предобработку данных в модуле интеграции со сторонними системами.

После того, как данные по пользователю сохранены в системе, они могут быть использованы модулем анализа показателей пациента. Система «КардиоАналитик» анализирует данные пользователя на предмет наличия и развития хронических заболеваний, наличия базовых факторов риска развития сердечно-сосудистых событий, медицинские исследования в области сердечно-сосудистой системы (пока что медицинские исследования ограничены только анализом ЭКГ). Модуль анализа показателей пациента помимо данных из модуля хранения данных системы «КардиоАналитик» также получает

данные из внешних источников: медицинские справочники и медицинские артефакты сторонних сервисов.

Модуль анализа хронических заболеваний использует описательный анализ для резюмирования общего состояния пациента, исходя из данных его анализов.

Модуль анализа показателей наличия базовых факторов СС событий использует показатели из результатов анализов и анкетных данных пациента для того, чтобы оценить риск возникновения СС событий при помощи шкалы SCORE.

В случае если у пациента есть обширные данные хронологически дифференцируемых показателей, то в анализе также примет участи и модуль анализа тенденций показателей пациента.

В результате проведения анализа показателей пользователя, система «КардиоАналитик» должна сгенерировать и выдать пациент-ориентированные рекомендации по управлению риском сердечно-сосудистых событий. Так же представить наглядную сводку произведенных типов анализа при помощи модуля графического представления результатов анализа.

Архитектура системы «КардиоАналитик» представлена на рисунке 2. Предложенная архитектура отличается от аналогов тем, что она ориентирована на СС события, использует методы интеллектуального анализа показателей пациента, а также работает только с показателями, полученными из медицинских, инструментальных исследований и анкетирования, а не с полной ЭМК пациента.

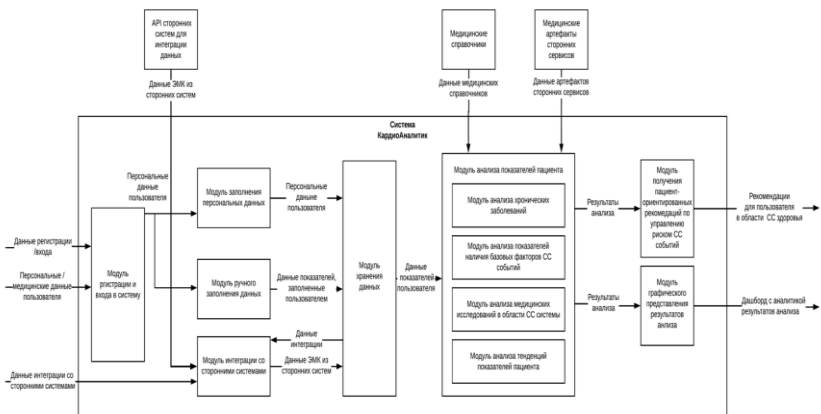


Рис.2. Архитектура системы «КардиоАналитик»

4. АНАЛИЗ ДАННЫХ

Аналитика разрабатываемой системы будет происходить с нескольких точек зрения. Предлагается интерпретировать данные, полученные путем анкетирования, инструментальных и медицинских исследований с точки зрения образа жизни пациента, его медицинских показателей, а также следования врачебным рекомендациям. При рассмотрении поведения пациента с позиции здравоохранения будут учитываться такие признаки как активность, вредные привычки, уровень физического и психологического здоровья. Следование врачебным рекомендациям будет анализироваться при фиксации динамики описанных выше показателей и анкетировании пациента. Также возможна корректировка при оценке риска развития сердечно-сосудистых событий исходя из динамики этих показателей, например это позволяют сделать Фрамингемская шкала [14] и шкала SCORE [12,13]. Анализ медицинских показателей пациента будет вестись при помощи таких методов интеллектуального анализа как дескриптивный анализ [15], классификация исследований ЭКГ при помощи ИНС (система «Кардио+»).

Фрагмент набора данных, используемого в исследовании, представлен на рис 3.

| var | age | dementia | XBP | SAD | DAD | Education | Family | Work | IMT | OT | IML _{LJ} | FV | Anemia | Triglycerids | XC | AG | FP | IM | YABJ | ONMK | SD _{2type} | Creatin | SFK | |
|-----|-----|----------|-----|-----|-------|-----------|--------|------|-----|------|-------------------|-------|--------|--------------|------|-----|-----|-----|------|------|---------------------|---------|-------|------|
| 0 | 1.0 | 77.0 | 0.0 | 0.0 | 150.0 | 95.0 | 1.0 | 0.0 | 0.0 | 24.8 | 98.0 | 125.2 | 54.0 | 0.0 | 0.45 | 5.0 | 1.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 79.0 | 82.3 |
| 1 | 2.0 | 48.0 | 0.0 | 1.0 | 170.0 | 100.0 | 1.0 | 1.0 | 1.0 | 32.6 | 102.0 | 212.3 | 56.0 | 0.0 | 3.08 | 4.5 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 120.5 | 60.2 |
| 2 | 3.0 | 84.0 | 0.0 | 0.0 | 145.0 | 90.0 | 1.0 | 1.0 | 0.0 | 31.9 | 109.0 | 142.8 | 47.0 | 0.0 | 0.42 | 4.2 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 87.0 | 70.1 |
| 3 | 4.0 | 79.0 | 0.0 | 0.0 | 170.0 | 110.0 | 2.0 | 1.0 | 0.0 | 27.7 | 105.0 | 251.3 | 34.0 | 0.0 | 0.36 | 5.3 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 99.0 | 62.1 |
| 4 | 5.0 | 64.0 | 0.0 | 0.0 | 160.0 | 100.0 | 1.0 | 1.0 | 0.0 | 20.8 | 90.0 | 175.4 | 50.0 | 0.0 | 0.45 | 5.1 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 96.0 | 71.7 |

Рис.3. Фрагмент набора данных

4.1. ОПИСАТЕЛЬНЫЙ АНАЛИЗ

Этот тип анализа отражает количественное описание данных. Он покрывает только ту выборку данных, по которой производится анализ, но не описывает совокупность, из которой взят данный набор данных. Часто его результаты используются на различных витринах данных, в нашем случае это может быть общее количество проанализированных системой показателей с разбиением их на группы и точки зрения, которые описывались выше [16]. А также статистику принадлежности к различным лингвистическим меткам отдельных показателей и групп показателей пациента. Этот вид анализа важен

для разрабатываемой системы из-за разнородности предоставляемых пациентом данных, так как пациент сам ведет свою медицинскую карту в системе. Например, какой-то пользователь может иметь какие-то специфичные медицинские анализы в своей ЭМК. Этот анализ позволит в дальнейшем оценивать качество предоставляемых системой результатов и повысит доверие пользователя к системе в целом.

В разрабатываемой системе суть описательного анализа заключается в том, что результаты анализов пациента можно представить в виде многозначного контекста. Этот контекст должен включать в себя множество пациентов, множество показателей анализа и их ограничения, множество значений для отдельного показателя. В системе используется 2 лингвистические шкалы для описания количественных и качественных показателей соответственно см. таблицу 1.

Таблица 1. Разработанные нечёткие лингвистические шкалы

| Тип показателя | Нечёткая лингвистическая шкала |
|----------------|--|
| Количественный | Ниже нормы – в рамках нормы – выше нормы |
| Качественный | В рамках нормы – вне нормы |

Пример соотношения количественного показателя индекса массы тела к шкале с указанием степени принадлежности представлено в таблице 2.

Таблица 2. Пример фазификации первых пяти значений критерия ИМТ

| ИМТ crisp | ИМТ fuzzy | Степень принадлежности |
|-----------|----------------|------------------------|
| 24.8 | Ниже нормы | 0.886862 |
| 32.6 | В рамках нормы | 0.823644 |
| 31.9 | В рамках нормы | 0.623654 |
| 27.7 | Ниже нормы | 0.852578 |
| 20.8 | Ниже нормы | 0.323037 |

Авторами данной статьи также была разработана система «Кардио+». Она используется для оценки состояния ССС на основе классификации ЭКГ. Эта система обучена на наборе данных МПТ-ВН

Arrhythmia, который является своего рода бенчмарком в области анализа ЭКГ. В данной работе предлагается сегментировать ЭКГ перед анализом. Основная идея заключается в том, чтобы разбить непрерывный сигнал ЭКГ разбить на окна продолжительностью в 10с, нормализовать сигнал, найти и оценить сигнал между соседними R – зубцами ЭКГ. Так же система способна выдавать рекомендации на основании классификации ЭКГ и степени риска развития ССЗ, который заранее определяется специалистом. Система может классифицировать 5 типов биения сердца.

4.2. ПРЕДИКТИВНЫЙ АНАЛИЗ

Цель данного анализа заключается в том, чтобы изучить взаимосвязи между медицинскими показателями, на ЭМК пациентов и разработать статистическую модель для новых, неполных и будущих посещений и корректировать оценку сердечно-сосудистых событий. Этот вид анализа в разрабатываемой системе будет представлен при помощи лингвистического резюмирования показателей и выявления тенденций в рамках отдельных показателей и их совокупности. Так же при увеличении пользователей системы станет возможно прогнозирование поведения этих тенденций при помощи метода цифрового двойника [17]

В настоящее время для определения ССР на ближайшие 10 лет активно используется шкала SCORE (Systematic COronary Risk Evaluation) [56]. Есть два варианта шкалы SCORE: для стран с низким риском и стран с высоким риском сердечно-сосудистых заболеваний (к ним относится Россия). У шкалы SCORE можно выделить следующие преимущества: она интуитивна и проста в использовании; является универсальным инструментом определения ССР для медицинских работников; позволяет осуществлять объективную оценку ССР; учитывает многофакторный характер ССЗ; довольно гибкая с точки зрения ведения пациента, т.к. если идеальный уровень фактора риска не может быть достигнут, то общий риск может быть уменьшен путем снижения других факторов риска. Однако у данной шкалы есть и ограничения – шкала SCORE оценивает ССР со смертельным исходом, а не общего, который складывается из случаев с фатальным и нефатальным исходом [55]. Также, т.к. создавалась в Европе, то адаптирована к различным европейским популяциям, но не к отдельным этническим группам даже в рамках исследуемых популяций. Шкала SCORE рекомендуется к использованию в определённой возрастной категории, а именно 40 - 65 лет. На начальном этапе развития данной системы планируется реализовать

калькулятор вычисления оценке по шкале SCORE для медицинских показателей пациента, а также динамику этой оценки при изменении показателей в рамках рекомендаций.

4.3. ПРЕСКРИПТИВНЫЙ АНАЛИЗ

Прескриптивный анализ – одна из углубленных форм аналитики, которая изучает данные и, в контексте рассматриваемой системы, отвечает на вопрос что может произойти с пациентом в дальнейшем и какие шаги стоит предпринять, чтобы избежать результатов негативных прогнозов. В рамках системы «КардиоАналитик» будут проанализированы тенденции развития показателей и дан прогноз дальнейшего развития СС событий, однако для этого потребуются данные пациента за несколько лет.

5. ПОДРОБНОСТИ РЕАЛИЗАЦИИ

Система «КардиоАналитик» будет поставляться как многопользовательский WEB-сервис для каждого желающего. Поэтому на начальном этапе пользователи будут вынуждены сами заполнять личные анкетные данные и данные медицинских исследований, анализов. Однако уже ищутся возможные пути интеграции с различными медицинскими сервисами для получения данных в машиночитаемом виде и использования описываемой системой.

Личные, анкетные и данные анализов представляют собой чётко определенные структуры, которые хорошо ложатся на концепцию реляционных баз данных. Для хранения таких данных планируется использовать базу данных PostgreSQL. PostgreSQL – это реляционная база данных с открытым исходным кодом, которая обладает такими качествами как поддержка пользовательских объектов и их поведения, включая типы данных, функции, операции, домены и индексы; неограниченный размер БД, стремится соответствовать стандарту ANSI-SQL:2008, отвечает требованиям ACID.

Для хранения различных графических файлов исследования, а также отчетов об аналитике, которую будет предоставлять сама разрабатываемая система, предлагается использовать сервер хранения объектов с открытым исходным кодом MinIO. Из его достоинств можно выделить простоту использования, большую производительность, хорошую документацию, полную интеграцию с S3 API, кроссплатформенность, популярность на отечественном рынке производителей ПО.

Для задач анализа, манипуляций с данными и внедрения методов машинного обучения используется язык Python и такие библиотеки как Pandas, Keras, TensorFlow.

6. ЗАКЛЮЧЕНИЕ

В данной статье была предложена архитектура информационной системы оценки сердечно-сосудистого здоровья, состоящая из модуля регистрации, сбора данных, их хранения, различных видов системного анализа, которая в первую очередь ориентирована на пациента и поддерживает требования безопасности и деперсонализации данных. Стоит отметить, что данное исследование пока ещё не завершено. Поэтому рассмотрены лишь отдельные подсистемы и представлена реализация некоторых из них.

Литература

1. Informatics: Research and Practice – [Электронный ресурс] – Режим доступа: [https://amia.org/about-amia/why-informatics/informatics-research-and-practice#:~:text=Clinical%20Research%20Informatics%20involves%20the,research%20use%20of%20clinical%20data.\(%D0%B4%D0%B0%D1%82%D0%B0](https://amia.org/about-amia/why-informatics/informatics-research-and-practice#:~:text=Clinical%20Research%20Informatics%20involves%20the,research%20use%20of%20clinical%20data.(%D0%B4%D0%B0%D1%82%D0%B0) (дата обращения 20.02.2022)
2. Кобринский Б. А. Искусственный интеллект в медицине: горячие точки // Девятнадцатая Национальная конференция по искусственному интеллекту с международным участием КИИ-2021 (11–16 октября 2021 г.). Труды конференции / под ред. В. В. Борисова, Б. А. Кобринского. – Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2021. – С. 13-29
3. Проект МТС 120/80 – [Электронный ресурс] – Режим доступа: <https://a.mts.ru/cardiorgo> (дата обращения 24.02.2022)
4. Проект СберЗдоровье – [Электронный ресурс] – Режим доступа: <https://www.sberbank.com/ru/eco/sberhealth> (дата обращения 25.02.2022)
5. Гусев А.В., Гаврилов Д.В., Новицкий Р.Э., Кузнецова Т.Ю., Бойцов С.А. Совершенствование возможностей оценки сердечно-сосудистого риска при помощи методов машинного обучения. Российский кардиологический журнал. 2021;26(12):4618. <https://doi.org/10.15829/1560-4071-2021-4618>
6. Гусев А. В., Токарев С. А., Гаврилов Д. В., Кузнецова Т. Ю. Применение автоматизированной системы поддержки принятия врачебных решений при диспансеризации взрослого населения для контроля правильности оценки уровня сердечно-сосудистого риска // Менеджмент качества в медицине, 2022.- № 1. - С. 72-79.
7. А.В. Гусев, Д.В. Гаврилов, И.Н. Ксаков и др. - Перспективы использования методов машинного обучения для предсказания сердечно-сосудистых заболеваний - Журнал врач и информационные технологии №3 2019.
8. Martin Wiesner and Daniel Pfeifer - Health Recommender Systems: Concepts, Requirements, Technical Basics and Challenges [Электронный ресурс] –

- режим доступа: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3968965/>
(дата обращения 21.11.2021)
9. Указ президента Российской Федерации об утверждении перечня сведений конфиденциального характера – [Электронный ресурс] – Режим доступа: <http://pravo.gov.ru/proxy/ips/?docbody=&nd=102046005&intelsearch=%D3%E A%E0%E7+%CF%F0%E5%E7%E8%E4%E5%ED%F2%E0+%D0%EE%F1%F1%E8%E9%F1%EA%EE%E9+%F4%E5%E4%E5%F0%E0%F6%E8%E8+%EE%F2+6+%EC%E0%F0%F2%E0+1997+%E3.+%B9+188> (дата обращения 26.02.2022)
 10. Федеральный закон "О персональных данных" от 27.07.2006 N 152-ФЗ – [Электронный ресурс] – Режим доступа: http://www.consultant.ru/document/cons_doc_LAW_61801/ (дата обращения 02.03.2022)
 11. Service-Oriented Architecture. In: Services Computing. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-38284-3_5
 12. Massimo F. Piepoli, Arno W. Hoes, Stefan Agewall и др. Европейские рекомендации по профилактике сердечно-сосудистых заболеваний в клинической практике (пересмотр 2016) - Российский кардиологический журнал No 6 (146) | 2017
 13. Conroy R. M., et al. SCORE project group. Estimation of ten-year risk of fatal cardiovascular disease in Europe: the SCORE project (англ.)// Eur Heart J (англ.)русск. : journal. — 2003. — June (vol. 24, no. 11). — P. 987—1003. — PMID 12788299.
 14. Максимов С.А. Применение метода оценки популяционного риска развития сердечно-сосудистых заболеваний: обоснование и примеры использования. Кардиология. 2019;59(7):44–51.
 15. T.V. Afanasieva, P.V. Platov - Cardiovascular Health Analysis System Using Machine Learning Model - Proceedings of the 10th International Scientific and Practical Conference named after A. I. Kitov "Information Technologies and Mathematical Methods in Economics and Management (IT&MM-2020)
 16. К. Андерсон – Аналитическая культура от сбора данных до бизнес-результатов – O'Reilly – 2017г – 336 с.
 17. Васильев Александр Николаевич; Тархов, Дмитрий Альбертович; Малыхина, Галина Федоровна. Методы создания цифровых двойников на основе нейросетевого моделирования. Международный научный журнал «Современные информационные технологии и ИТ-образование», [S.l.], v. 14, № 3, p. 521-532, sep. 2018. ISSN 2411-1473.

ОСОБЕННОСТИ ПОСТРОЕНИЯ ПРИКЛАДНОЙ ОНТОЛОГИИ ТИПОВЫХ АРХИТЕКТУР ИНТЕГРИРОВАННЫХ ЭКСПЕРТНЫХ СИСТЕМ С ИСПОЛЬЗОВАНИЕМ СРЕДСТВ ИНТЕЛЛЕКТУАЛЬНОЙ ПРОГРАММНОЙ СРЕДЫ КОМПЛЕКСА АТ-ТЕХНОЛОГИЯ

Рыбина Г.В., д.т.н., профессор

e-mail: GVRybina@mephi.ru

Слиньков А.А., аспирант

e-mail: sanch971@mail.ru

Григорьев А.А., магистрант

e-mail: grigandal625@gmail.com

НИЯУ МИФИ

1. ВВЕДЕНИЕ

Понятие онтологии типовых архитектур интегрированных экспертных систем (ИЭС) [1, 2] является естественным развитием задачно-ориентированной методологии построения ИЭС и интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ [3], предназначенных для автоматизации и интеллектуализации процессов проектирования программного обеспечения различных классов ИЭС, особенно на трудоемких этапах системного анализа.

Детальное описание методологии и технологии построения прикладных ИЭС приводится в базовых работах [3] и др., а здесь важно отметить, что принципиальная особенность данной методологии заключается в концептуальном и программном моделировании архитектур разрабатываемых ИЭС на всех уровнях конкретизации процессов интеграции в ИЭС, что эффективно поддерживается мощной функциональностью интеллектуального планировщика и технологической базы знаний (БЗ) [4], включающей типовые проектные процедуры (ТПП) построения наиболее распространенных архитектур ИЭС (статических, динамических и обучающих ИЭС), совокупность операционных и информационных повторно-используемых компонентов (ПИК), а также прикладную онтологию типовых архитектур ИЭС.

Основными предпосылками для расширения технологической БЗ путем создания прикладной онтологии типовых архитектур ИЭС послужили, с одной стороны, значительный объем накопленного информационного и программного обеспечения различных классов ИЭС, разработанных в последние годы, а с другой стороны, в контексте взаимодействия с интеллектуальным планировщиком - необходимость снижения степени доступности к семантически разнородным ПИК с неявно выраженным функционалом при их поиске и инициализации в условиях реализации конкретной ТПП.

В целом, если рассматривать новые подходы к автоматизации и интеллектуализации процессов проектирования программных систем с использованием или под управлением онтологий [1,2], то место и роль онтологий здесь существенным образом определяется уровнем сложности моделей архитектур разрабатываемых систем, наличием адекватных моделей жизненного цикла (ЖЦ) и семантически корректным отражением базовых процессов проектирования на всех этапах ЖЦ с учетом онтологического представления проектируемой архитектуры, состава, структуры и спецификаций отдельных компонентов и связей между ними.

Тем не менее, к настоящему времени, несмотря на широкий спектр работ по онтологиям и онтологическому инжинирингу в рамках направлений OBSE или ODSE [5-11] и др., вопросы, связанные с разработкой практически значимой и семантически адекватной онтологической модели процессов проектирования программных систем, в частности, интеллектуальных систем, практически, не рассматриваются. В связи с этим расширение исследований в рамках создания интеллектуальной технологии построения ИЭС как одного из наиболее распространенных классов интеллектуальных систем, обладающих развитыми масштабируемыми архитектурами, является сегодня особенно важным, в том числе с точки зрения интеграции разработанных методов интеллектуального планирования с онтологическим подходом [4].

В данной работе основное внимание уделяется обсуждению результатов экспериментального программного исследования текущей версии прикладной онтологии типовых архитектур ИЭС, модель и методы построения которой позволяют совместно учитывать семантические особенности моделей архитектур проектируемых ИЭС и особенности покомпонентной функциональности в виде совокупности ПИК для каждой ТПП.

Следует отметить, что поскольку для реализации именно архитектур обучающих ИЭС в настоящее время накоплено

наибольшее число программных компонентов и средств, прошедших апробацию и реинжиниринг в составе инструментального комплекса АТ-ТЕХНОЛОГИЯ и оформленных в виде операционных и информационных ПИК, то соответствующая ТПП «Построение обучающих ИЭС и веб-ИЭС» была выбрана в качестве базового «полигона» для различных исследований и экспериментального программного моделирования.

2. КРАТКАЯ ХАРАКТЕРИСТИКА МОДЕЛИ ОНТОЛОГИИ ТИПОВЫХ АРХИТЕКТУР ИЭС И ОСОБЕННОСТИ ПОСТРОЕНИЯ ПРИКЛАДНОЙ ОНТОЛОГИИ

Как было показано в [1,2], на выбор базовой модели онтологии типовых архитектур большое влияние оказал положительный опыт создания для обучающих ИЭС и практического использования в учебном процессе нескольких онтологий курсов/дисциплин, разработка которых осуществлялась на основе модели в виде семантической сети специального типа [3,12].

Поэтому в качестве модели онтологии здесь используется модифицированная семантическая сеть, но более простого вида [1,2]: $M_{om} = \langle V_{om}, U_{om}, PD_{om} \rangle$, где V_{om} – множество элементов модели архитектуры ($M_{исс}$), построенной исходя из идей глубинной интеграции компонентов (на всех уровнях интеграции), причем каждый элемент включает название вершины онтологии, вес (в интервале 0...100) и информацию об используемых ПИК; U_{om} – множество связей нескольких типов между элементами модели $M_{исс}$ (родительские и дочерние вершины онтологии), причем семантика типов этих связей может широко варьироваться (агрегация, ассоциация, иерархия, сильная, средняя и слабая связи и др.) и интерпретироваться в зависимости от используемой ТПП; PD_{om} (опционально) множество специальных данных, т.е. информация различного характера, специфицирующая особенности и/или нестандартные подходы к разработке отдельных компонентов прототипа ИЭС (параметры, тексты, кодификаторы, сведения о внешних подсистемах, компонентах, приложениях и т.д.).

Соответственно, текущая версия прикладной онтологии [1,2] представляется в виде: $O_{арх} = \langle M_{om}, F_{арх} \rangle$, где M_{om} – модифицированная модель типовых архитектур ИЭС; $F_{арх}$ – множество базовых и модифицированных операций (процедур) построения элементов онтологии, реализованных в виде программных компонентов, каждый из которых в соответствии с требованиями

интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ оформлен как операционный ПИК.

Теперь рассмотрим кратко некоторые аспекты построения текущей версии прикладной онтологии типовых архитектур ИЭС, основная цель и назначение которой заключается в учете и отражении на разных уровнях онтологии особенностей и логических взаимосвязей модели архитектуры проектируемой ИЭС, а также конкретизации методов и алгоритмов, реализованных в соответствующих ПИК.

Здесь следует отметить, что модель архитектуры прототипа ИЭС, представляемая в виде иерархии расширенных диаграмм потоков данных (РДПД) [3], является одной из важных составляющих процесса прототипирования ИЭС, поскольку ее структура во многом определяет состав прототипа ИЭС и его функциональные и интеграционные возможности для реализации конкретного класса решаемых задач. Следствием многоуровневости интеграционных процессов, отражаемых в иерархии РДПД, является появление элементов архитектуры на разных уровнях вложенности, что приводит к использованию различных методов и алгоритмических решений, включая разнообразный спектр информационных и операционных ПИК.

Поэтому структура прикладной онтологии типовых архитектур ИЭС, включает несколько уровней: уровень актуализированных архитектур; уровень элементов этих архитектур, содержащий неограниченное количество подуровней, зависящее от вложенности компонентов, входящих в состав элементов архитектур ИЭС; далее представлены операции, которые выполняют компоненты элементов архитектур; на нижнем уровне находятся ПИКи, реализующие операции компонентов элементов архитектур ИЭС.

Что касается используемых типов связей, предусмотренных моделью онтологии в виде упрощенной семантической сети, то в настоящее время в текущей версии прикладной онтологии типовых архитектур ИЭС, программно поддерживаются три типа связей между элементами онтологии: связь типа «часть-целое» (агрегация) для связывания элементов онтологии, находящихся на разных, но смежных уровнях; связь типа «ассоциация» для связывания элементов онтологии, находящихся на одном уровне; «слабая» связь для связывания элементов, находящихся как на одном, так и на смежных уровнях.

С учетом этих концептуальных и функциональных особенностей прототипирования ИЭС [3,4], все алгоритмы и

процедуры построения, хранения и сопровождения прикладной онтологии типовых архитектур ИЭС как важной части технологической БЗ были разработаны таким образом, чтобы обеспечить возможность привлечения и гибкой настройки соответствующих ПИК для выполнения с помощью интеллектуального планировщика всех плановых задач в зависимости от особенностей модели архитектуры проектируемой ИЭС (при этом некоторые задачи инженер по знаниям может выполнить самостоятельно или совместно с экспертом).

Для автоматизированной поддержки процессов построения прикладной онтологии типовых архитектур ИЭС использовались модифицированные инструментальные средства, функционирующие в составе базовых средств комплекса АТ-ТЕХНОЛОГИЯ и позволяющие полностью реализовать необходимую функциональность, а также для моделирования взаимодействия с интеллектуальным планировщиком [4] привлекались разработанные средства управления взаимодействием с технологической БЗ.

3. ПРИМЕР ФРАГМЕНТА ПРИКЛАДНОЙ ОНТОЛОГИИ ТИПОВЫХ АРХИТЕКТУР ИЭС (ТПП «ПОСТРОЕНИЕ ОБУЧАЮЩИХ ИЭС И ВЕБ-ИЭС»)

В качестве примера приведем фрагмент прикладной онтологии типовых архитектур ИЭС, где показана концептуальная (логическая) и программная взаимосвязь между совокупностью стандартных для обучающих ИЭС процессов, связанных с отображением текущей модели обучаемого, построенной в результате веб-тестирования, на онтологию конкретного курса/дисциплины и формированием в зависимости от полученных результатов индивидуальной стратегии (плана) обучения [3,12].

Как было отмечено выше, общее управление этими и другими процессами в двух базовых режимах функционирования обучающих ИЭС (DesignTime и RunTime) [3,4,12] поддерживается интеллектуальным планировщиком, ТПП «Построение обучающих ИЭС и веб-ИЭС» и совокупностью ПИК, реализованных на основе использования значительного объема различных программных и информационных средств, зарегистрированных в разные годы в комплексе АТ-ТЕХНОЛОГИЯ и входящих в состав подсистемы поддержки построения обучающих ИЭС.

Поэтому важное значение для построения прикладной онтологии типовых архитектур ИЭС приобретают вопросы, связанные с предварительным анализом, структурной и функциональной

идентификацией и формальным представлением всех программных средств и компонентов в соответствии с базовой моделью ПИК [3]. Ниже представлены формальные описания (конкретизации) двух концептуально связанных ПИК, реализующих упомянутую выше функциональность.

Конкретизация ПИК «Отображение текущей модели обучаемого на онтологию курса/дисциплины» в соответствии с базовой моделью ПИК $\langle N, Arg, F, PINT, FN \rangle$ [3] определяется как:

N – имя зарегистрированного компонента «Отображение текущей модели обучаемого на онтологию курса/дисциплины»;

$Arg = \langle Arg1, Arg2 \rangle$,

где $Arg1$ – онтология курса/дисциплины, (предварительно построенная в режиме DesignTime [3,12]); $Arg2$ – текущая модель обучаемого ($M1_{тек}$);

$F: Arg1 \times Arg2 \rightarrow R$ – метод оценивания результатов тестирования обучаемых,

где $R = \{r_j\}$, ($j = 1 \div m$) – множество «проблемных зон» конкретного обучаемого [3,12];

$PINT$ – интерфейс ПИК «Формирование стратегий обучения»;

FN – функция формирования текущего уровня компетенций обучаемого в соответствии с онтологией курса/дисциплины [12].

Соответственно, конкретизация ПИК «Формирование стратегий обучения» определяется как:

N – имя зарегистрированного компонента «Формирование стратегий обучения»;

$Arg = \langle Arg1, Arg2 \rangle$,

где $Arg1 = \{arg1_i\}$ – множество онтологий курсов/дисциплин, ($i=1 \div n$), (n –количество онтологий курсов/дисциплин);

$Arg2 = \{arg2_j\}$ – множество $M1_{тек}$, ($j=1 \div p$), где p – количество $M1_{тек}$;

$F = \langle F1, F2 \rangle$,

где $F1: Arg1 \times Arg2 \rightarrow A$ – метод формирования множества стратегий (планов) обучения,

где $A = \{a_i\}$ ($i=1 \div k$) – множество планов обучения,

где k – количество планов;

$F2: Arg1 \times A \rightarrow B$ – метод формирования множества обучающих воздействий на основе планов,

где $B = \{B_i\}$ ($i=1 \div m$) – множество обучающих воздействий, где m – количество обучающих воздействий;

$PINT$ – интерфейс ПИК «Управление реализацией обучающих воздействий»;

FN - формирование множества планов (стратегий) обучения, содержащего упорядоченное множество обучающих воздействий.

На рис.1 представлен фрагмент прикладной онтологии типовых архитектур ИЭС, верхние уровни которой отражают функционально-структурные особенности подсистемы поддержки построения обучающих ИЭС, функционирующей в базовой версии комплекса АТ-ТЕХНОЛОГИЯ [3,12]. Средние и нижние уровни онтологии формируются на основе формальных конкретизаций ПИК.

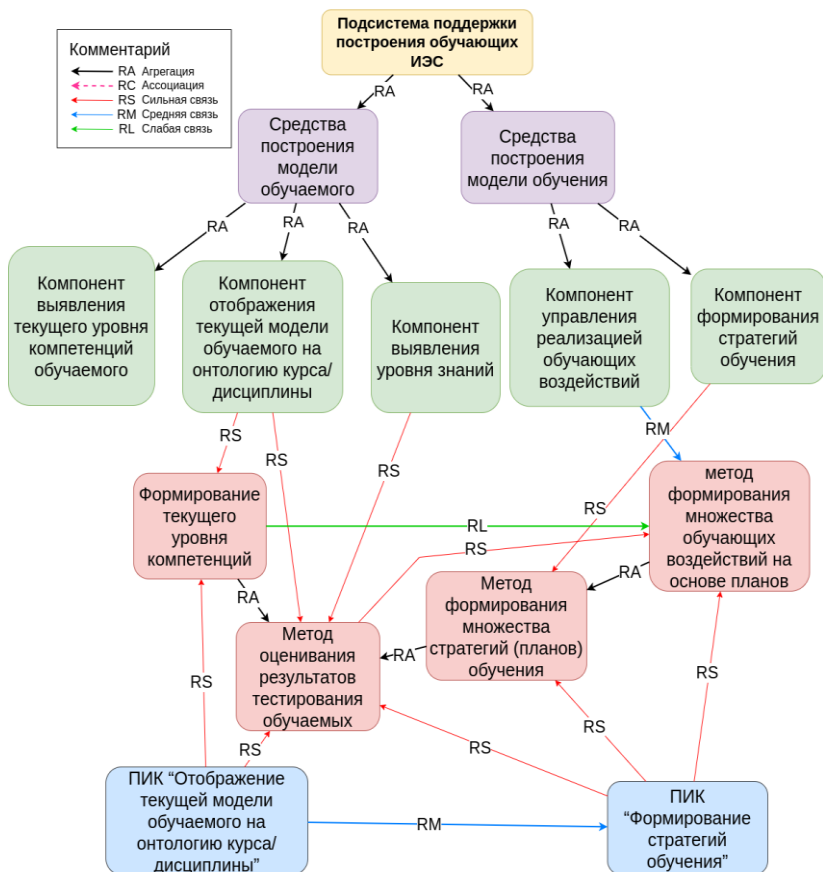


Рис.1. Фрагмент прикладной онтологии типовых архитектур для ТПП «Построение обучающих ИЭС».

Наибольшее разнообразие связей наблюдается на уровнях методов/операций, где используются такие типы связей, как, например, связи типа RA (агрегация) между методом формирования множества обучающих воздействий и методом формирования множества стратегий обучения, а также связей типа RS (сильная) между методом оценивания результатов тестирования и ПИК «Формирование стратегий обучения» и др.

4. ЗАКЛЮЧЕНИЕ

Проводимые исследования и экспериментальное программное моделирование методов проектирования программного обеспечения прикладных ИЭС с различной архитектурной типологией под управлением онтологий являются достаточно новыми для технологий искусственного интеллекта и программной инженерии в целом.

Здесь еще рано ожидать практических результатов в области создания эффективных онтологических моделей и мощных инструментальных средств и платформ для автоматизации и интеллектуализации процессов разработки программного обеспечения интеллектуальных систем. Тем не менее накопленный опыт и постоянно развивающаяся технологическая база в виде интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ позволяют поэтапно решать поставленные научные и практические задачи.

Литература

1. Rybina G.V., Slinkov A.A. The Implementation of the Ontological Approach to Control of the Processes of Designing Integrated Expert Systems Based on the Problem-Oriented Methodology. // Artificial Intelligence. 19th Russian Conference, RCAI 2021, Taganrog, Russia, October 11-16, 2021. Proceedings. Springer Nature Switzerland AG 2021. P 354-364.
2. Rybina G., Slinkov A., Sorokin I. Ontological Approach to Software Development for Integrated Expert Systems Created on the Basis of the Problem-Oriented Methodology // Proceedings of the XXIII International Conference "Enterprise Engineering and Knowledge Management" (EEKM 2020). Moscow, Russia. CEUR Workshop Proceedings. 2021. Vol. 2919, P. 209-217.
3. Рыбина Г.В. Теория и технология построения интегрированных экспертных систем. Монография. – М.: Научтехлитиздат, 2008.

4. Рыбина Г.В., Блохин Ю.М. Методы и программные средства интеллектуального планирования для построения интегрированных экспертных систем. // Искусственный интеллект и принятие решений. 2018. №1. С. 12-28.
5. Ontologies for Software Engineering and Software Technology. 1st Ed. / C. Calero, F. Ruiz, M. Piattini – Springer, 2006.
6. Happel, H.J. KOntoR: An Ontology-enabled Approach to Software Reuse / H.J. Happel, A. Korthaus, S. Seedorf, P. Tomczyk // Proc. SEKE 2006: the 18th International Conference on Software Engineering & Knowledge Engineering (July 5-7, 2006, California, USA, 2006).
7. Bossche, M.V. Ontology Driven Software Engineering for Real Life Applications / M.V. Bossche, P. Ross, I. MacLarty, B. van Nuffelen, N. Pelov // Proc. 3rd International Workshop on Semantic Web Enabled Software Engineering, SWESE 2007 (Innsbruck, Austria, 2007).
8. Jabar, M.A. General Domain Ontology in Enterprise Software Development Process / M.A. Jabar, M.S. Khalefa // International Journal of Engineering and Advanced Technology. – 2019. - Vol. 8, Is. 3S.
9. Хорошевский В.Ф. Проектирование систем программного обеспечения под управлением онтологий: модели, методы, реализации //Онтология проектирования. 2019. Т.9. №4. С.429-448.
10. Ерженин Р.В.,Массель Л.В. Онтологический подход к представлению знаний о методологии моделирования сложной системы управления. // Онтология проектирования. 2020. Т.10.№4.С. 463-474.
11. Негода В.Н.,Куликова А.А. Сквозное проектирование автоматизированных систем на основе онтологий // Онтология проектирования. 2021.Т.11.№4.С.450-463.
12. Г.В. Рыбина. Интеллектуальная технология построения обучающих интегрированных экспертных систем: новые возможности // Открытое образование. 2017. №4. С. 43-57.

ТРЕХМОДАЛЬНАЯ СИСТЕМА ДЛЯ ТЕКСТОВОГО ОПИСАНИЯ ВИДИМОГО МИРА

*Харламов А.А., д.т.н., с.н.с.
ИВНДуНФ РАН
e-mail: kharlamov@analyst.ru
Жаркой Р.М., президент
ISS, Corp.
e-mail: roman.jarkoi@iss.ru*

1. ВВЕДЕНИЕ

Появление второй сигнальной системы ознаменовало появление на Земле нового вида *Homo Sapiens* не случайно: именно появление языка позволило перейти от сильновариативного представления мира в сознании в виде его экстралингвистической модели к менее вариативному языковому представлению. Все существующие технологии анализа этого мира таким образом фактически сводятся к его анализу в экстралингвистическом представлении и к анализу мира в его языковом представлении. Мы видим успехи лингвистического анализа описания мира и в меньшей мере успехи экстралингвистического анализа. На самом деле эти два вида анализа не должны противопоставляться друг другу, а работать вместе дополняя друг друга. Дело в том, что зрительная модальность приносит в мозг до 90% значимой информации. Но практически все знания мира хранятся в текстовой форме. Далее мы рассмотрим механизм, который позволит перейти от анализа мира в его экстралингвистическом представлении к его анализу в языковом представлении.

Экстралингвистическая модель мира в обычно представляется зрительной модальностью. Это происходит потому, что другие модальности несут меньше значимой информации, но также и потому, что моделировать представление других экстралингвистических модальностей сложнее, чем зрительной, в основном из-за трудности реализации соответствующих сенсоров.

Тем не менее, зрительный анализатор, будучи дистантным, имеет много неудобств в однозначном представлении информации о мире: например, исчезнувший и вновь появившийся зрительный образ

движущегося объекта нельзя с уверенностью отнести к одному и тому же отображаемому объекту. И таких примеров можно привести сколько угодно. Другими словами, помимо чисто зрительного отображения требуется отображение еще одной сенсорной модальности, которая сохраняет физическую непрерывность мира в своем представлении. Такой модальностью является соматосенсорная модальность. Условно можно отнести к моделям соматосенсорного представления мира так называемые 3D модели (иногда используют неполные – 2.5D модели). Рассмотрим все это более подробно.

Традиционные способы анализа видеоряда сводятся к классификации изображений [1]. Попытки классифицировать сцены сводятся к перечислению объектов на сцене [2, 3], а анализ динамических сцен еще даже не начинался [4]. Сложность анализа видеорядов заключается в большой вариативности изображений, снять которую привычными методами, даже с использованием глубоких нейронных сетей, не представляется возможным.

Текст, описывающий видеоряд, в отличие от собственно видеоряда значительно менее вариативен, и потому, подход к анализу видеоряда через анализ текстов, описывающих видеоряд, кажется более плодотворным [5]. Попытаемся рассмотреть его более содержательно. Для этого вместо анализа видеорядов попробуем рассмотреть анализ текстов, их описывающих. Но для этого необходимо автоматизировать процедуру описания видеоряда текстом. Такая процедура имеется: кадры имеющегося видеоряда распознаются как ситуации, и им ставится в соответствие их (ситуаций) описание, которое порождается, например, конкретным продукционным правилом интерпретирующей ситуацию экспертной системы.

Однако, как это чаще всего бывает (см. например, [6]), такое описание оказывается не очень точным в силу большой вариативности визуализированной ситуации. Чтобы улучшить результат, оценке (а потом и описанию) ситуации предшествует дополнительный шаг, который сводится к моделированию ситуации с помощью 2.5D модели. Введение контекста (исключение лишнего и выявление необходимого) позволяет снизить вариативность представления сцены. Этот контекст включает два компонента: (1) модель мира в целом; и (2) модель отдельной ситуации. Рассмотрим этот вопрос более подробно. Для этого обратимся к представлению модели мира в сознании человека [4].

Модель мира в сознании человека представлена двумя компонентами [7] языковым (включающим модель языка и модель мира, описанную текстами языка) и экстралингвистическим (модель,

описанную другими, помимо текстовой, модальностями – в первую очередь, зрительной модальностью, но также и соматосенсорной модальностью тоже). Оба эти компонента представляют собой иерархии словарей образов событий разной сложности разных модальностей, которые формируются в процессе структурной обработки кодовых сенсорных последовательностей.

Так языковая иерархия представлена словарями уровнеобразующих элементов языка разной сложности от словаря фонем для звучащей речи, или графем – для письменных текстов – до словаря попарной сочетаемости корневых основ, являющегося словарем допустимой сочетаемости слов в языке – словарем семантического уровня. Экстралингвистическая иерархия также представлена словарями уровнеобразующих элементов квази-текстов (в рассматриваемом случае – зрительной модальности), на верхнем уровне которой представлена также сочетаемость, теперь уже – образов объектов и событий, что тоже соответствует семантическим ограничениям на сочетаемость.

И в языковом, и в экстралингвистическом компонентах пары корневых основ и соответствующих им объектов и событий виртуально объединяются в однородные семантические сети, которые и представляют эти два компонента модели мира. Означающие и представляемые ими в сознании человека означаемые оказываются связанными ассоциативно – объект и обозначающее его слово – то есть эти две сети работают как единое целое и представляют модель мира человека.

Экстралингвистическая часть модели мира не ограничивается иерархией словарей зрительной модальности. Большая вариативность отражений видимого мира в зрительной модальности компенсируется ограничениями соматосенсорной модели мира человека, которая опирается на личный соматосенсорный опыт конкретного человека. В частности, проявляющаяся время от времени негладкость восприятия мира зрительным анализатором, возникающая из-за физических законов процедуры преобразования трехмерного представления мира в двумерную картинку, сглаживается соматосенсорной моделью мира. В этом случае соматосенсорная модель включает в свой состав строго все, что описывает конкретную ситуацию, и не включает ничего, что ее не касается.

Модель отдельной ситуации в сознании человека представлена в ламелях гиппокампа [7]. В ассоциативной памяти Хопфилда [8] отдельной ламели гиппокампа формируется и хранится комбинация образов объектов и событий в пространстве и во времени,

представленных (и образов объектов и событий в экстралингвистической части модели, и их означающих в языковой части модели), как они включены в отображаемую ситуацию.

В ламели гиппокампа, таким образом, может быть представлена как сама ситуация, так и предложение текста, ее описывающее. Последовательность таких предложений представляет собой текст, описывающий сценарий. Тексты, описывающие разные видеоряды, можно анализировать, например, формировать однородную семантическую сеть, описывающую содержание конкретного текста [7]. В этом случае эти тексты можно сравнивать (посредством сравнения их семантических сетей) классифицировать (посредством классификации их сетей), что решает задачу не только анализа отдельных ситуаций, но и анализа целых сценариев поведения.

Модель такого анализа, включающая анализ видеорядов, полученных с видеокамер, 2.5D модель мира, вносящая ограничения на допустимые объекты и события и исключающая лишние объекты и события, перечень шаблонов ситуаций, объединяющих объекты и события как комбинации в пространстве и во времени, формирование сценариев, характеризующих поведение моделей агентов на 2.5D сцене, как последовательностей этих предложений, анализ однородных семантических сетей этих текстов, и представлены в этой работе.

2. ТРЕХМОДАЛЬНАЯ СИСТЕМА ДЛЯ ТЕКСТОВОГО ОПИСАНИЯ ВИДИМОГО МИРА

Анализ видеоряда видимого мира. Видимый мир воспринимается видеокамерой (группой видеокамер), оснащенных современными системами видеоаналитики. Такая видеокамера позволяет сегментировать объекты на сцене, оценить их координаты, направление движения (если они движутся), скорость и ускорение. Продвинутое средство описания ситуации позволяет также трекировать траекторию движения объектов на сцене. Еще более продвинутое средство позволяет оценивать позу живых объектов (скелет изображения объекта), и таким образом, классифицировать позу человека. И даже распознавать всю иерархию зрительных представлений от текстуры объектов до частей объектов, объектов и комбинаций объектов – сцен [9].

2.5D модель мира. Наличие инструментов, позволяющих моделировать в 2.5D представлении видимый мир [9], исключив из модели все, что не представляет интереса, и упростив остальное, мы получаем идеальную модель, которая легко интерпретируется системой

продукционных правил, результатом работы которых является, в том числе, генерация предложения (или нескольких предложений) текста, описывающего анализируемую ситуацию.

Продукционное правило в качестве шаблона сцены. Конкретная ситуация, например мытье рук перед входом в рабочую зону, может быть описана иерархией продукционных правил, описывающей специфические положения тела и рук агента, сводящейся в конечном итоге к двум базовым состояниям: (1) агент вымыл руки; (2) агент не вымыл руки. Продукционное правило порождает соответствующий ответ, который (помимо функции взаимодействия с оператором – если это необходимо) используется далее для формирования текстового описания сценария поведения агента.

Формирование текстового описания. Последовательность предложений, описывающих поведение агента на сцене, составляет целый связный осмысленный текст, который характеризует сценарий поведения агента. Этот сценарий далее может быть использован для сравнения со сценариями легитимного и нелегитимного поведения агентов, описываемых регламентирующими документами (например, инструкциями).

Анализ текстового описания. Текст сформированного сценария подвергается семантическому анализу: формируется однородная семантическая сеть, которая подвергается классификации – отнесению к одной из заданных предметных областей, а потом – сравнению с текстом инструкции, описывающей предписанное поведение агента в рамках указанной предметной области. Выявляется отличие сценария от предписанного поведения и анализируются отличия.

Анализ инструкций. Для анализа сформированного сценария текущего поведения агента текст инструкции подвергается такому же анализу, как и текст сценария. Формируется однородная семантическая сеть инструкции, которая далее сравнивается с такой же сетью сценария, что позволяет выявить похожие части семантической сети сценария и семантической сети инструкции.

Далее, анализируются и похожие части, и различия сетей. Этот анализ позволяет оценить, в каких частях поведение агента укладывается в рамки инструкции, в каких – выходит за эти рамки.

Анализ инструкций позволяет выявить отдельные части поведения, представленные как подсети семантической сети, удалением слабых связей (дуг, имеющих малый вес). Такой анализ позволяет оценить эффективность инструкций, их полноту, точность et setera.

Сравнение описаний. Сравнение текстов описания сценария и инструкции осуществляется вычислением пересечения сетей – пересечения вершин и пересечения дуг, которое численно показывает степень подобия текстов, а содержательно – составом вершин и дуг пересечения – подобие текстов, что позволяет детально понять сложившуюся ситуацию.

Классификация описаний. Получение семантической сети описания сценария поведения агента позволяет также автоматически отнести сценарий к одному из классов поведения агента. Для этого множество сценариев, сформировавшее обучающую выборку, кластеризуется на содержательные классы, которым присваиваются экспертами имена реальных ситуаций (легитимных и нелегитимных). И тогда в процессе классификации реальных сценариев происходит их отнесение к тому или иному классу.

3. РЕАЛИЗАЦИЯ ТРЕХМОДАЛЬНОЙ СИСТЕМЫ ДЛЯ ТЕКСТОВОГО ОПИСАНИЯ ВИДИМОГО МИРА

Макет работающей системы был реализован в фирме Интеллектуальные системы безопасности (ISS), г. Москва. Моделировалось помещение офиса фирмы с целью оценки бизнес-процессов в помещении офиса.

Система анализа бизнес-процесса включает следующие компоненты: (1) видеопериферия в составе нескольких видеокамер; (2) видеоаналитика, включающая распознавание изображений объектов на сцене, а также – распознавание сцен; (3) 2.5D модель офисного помещения с представленными на ней агентами; (4) перечень шаблонов сцен, возникающих на 2.5D модели; (5) подсистему формирования описания сценария поведения агента; (6) подсистему анализа текстов, описывающих сценарии; а также (7) подсистему проверки непротиворечивости системы продукционных правил, составляющих шаблоны сцен.

Офисное помещение на видео. Офисное помещение воспринимается видеокамерой (несколькими видеокамерами) – см. Рис. 1.a – и отображается на 2.5D модель (Рис. 1.b).

Офисное помещение может быть представлено иерархией зрительных образов разной степени общности [11].

Видеоаналитика. Видеоаналитика включает в свой состав функции: (1) выявления отдельных объектов на сцене; (2) оценки их размеров, а также координат и скорости перемещения; и (3) распознавание сцен [11]. Распознавание сцен реализуется с использованием перечня объектов на сцене, а также – с применением механизма оценки позы.

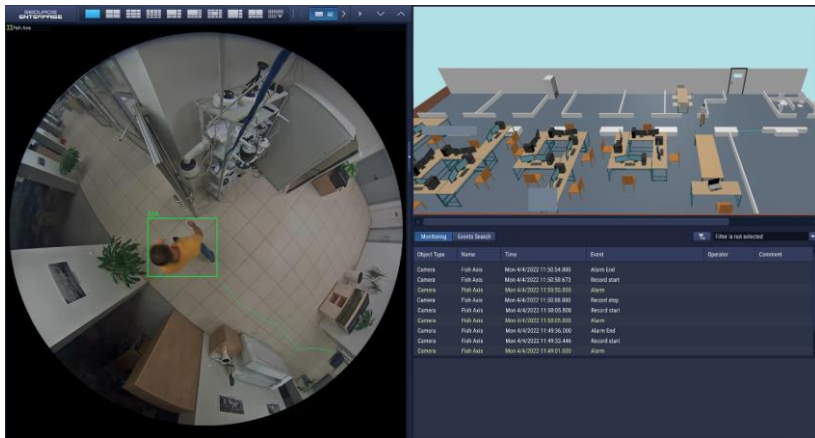


Рис. 1.а Офисное помещение воспринимается видеокamerой (несколькими видеокamerами)

Рис. 1.б Офисное помещение отображается на 2.5D модель

2.5D модель офисного помещения. 2.5D модель реализована на основе заготовок фирмы AnyLogic. Агенты, представленные на модели, перемещаются со скоростями реальных объектов и учитывают физику реального мира. Модель позволяет сглаживать поведенческие траектории агентов и склеивает их разрывы, возникающие вследствие ошибок восприятия реального мира: частичные или полные закрытия объектов, переход из поля зрения одной камеры в поле зрения другой камеры и подобное.

Система продукционных правил. Состав анализируемой сцены – множество объектов на сцене, отфильтрованное шаблоном сцены, – позволяет распознать сцену. Далее сцена описывается текстовым шаблоном (одно или несколько предложений), который выдается всякий раз, когда сцена появляется на входе видеопериферии.

Шаблон ситуации как система продукционных правил. Шаблон ситуации представляет собой граф, описывающий сцену. Вершинами графа являются имена объектов, входящих в сцену. Дуги характеризуют отношения этих объектов на сцене.

Первоначальный граф сцены – ассоциативный граф: вершины – просто перечень имен объектов на сцене. Дуги означают ассоциативные отношения.

Распознавание сцен. Первоначальный анализ сцены осуществляется просто классификацией групп объектов и событий, входящих в сцену. Разные множества характеризуются как разные по составу сцены.

Поскольку бизнес-процесс, сценарий которого используется для анализа, это некоторый контекст, не всякая сцена может войти в этот контекст: то есть как правило немного сцен конкурируют между собой за участие в анализе. Тем не менее, в случае наличия конкурентов дуги размечаются типами отношений [10], что снимает неоднозначность распознавания сцены.

Формирование предложений текста. После того, как сцена распознана, она включается в сценарий бизнес-процесса. Распознанная сцена описывается стандартным предложением, характеризующим эту сцену, которое включено в состав продукционного правила и производится при распознавании сцены.

Сценарий бизнес-процесса. Последовательность предложений текста, соответствующая последовательности сцен, привязанных к конкретному агенту, описывают сценарий бизнес-процесса. Такие тексты далее используются: (1) для выяснения непротиворечивости нового продукционного правила (нового предложения текста) с системой имеющихся продукционных правил, описывающих ситуацию; (2) для анализа текстов множества реализаций конкретного бизнес-процесса с точки зрения важности отдельных его частей; (3) для сравнения разных реализаций одного и того же бизнес-процесса (в том числе для сравнения конкретной реализации и шаблонной ситуации); и (4) для классификации бизнес-процессов (в том числе на фоне всех бизнес-процессов конкретного предприятия).

Вычисления на сценариях. Наличие таким образом представленного сценария позволяет провести некоторый его анализ. Например, оценить последовательность выполнения ситуаций, предписанных инструкцией. Помимо качественного анализа, что является наиболее сложной процедурой современных алгоритмов анализа ситуаций, производится численные оценки поведения агентов. Например, оценивается суммарное время нахождения агентом в легитимном или нелегитимном состоянии: например, вычисление времени, которое агент тратит на кухню, или на беседы с другими агентами.

4. ЗАКЛЮЧЕНИЕ

Для реализации эффективного информационного взаимодействия (диалогового управления) между человеком и когнитивным агентом необходимы не только (и не столько) В работе представлен комплексный подход к анализу бизнес-процессов на

промышленном предприятии, комплексность которого заключается в трех ипостасях: (1) в модели процессов используются три сенсорные модальности – зрительная, соматосенсорная и текстовая; (2) эти три ипостаси взаимно дополняют друг друга, формируя контекст, практически исключающий множественные толкования одной и той же сцены в процессе распознавания; и (3) такая модель позволяет работать с сильно вариативными образами зрительной модальности, сглаживая их представления образами соматосенсорной модальности, и с маловариативными описаниями текстовой модальности.

Литература

1. Moshiko Raboh, Roei Herzig, Jonathan Berant, Gal Chechik, Amir Globerson Differentiable Scene Graphs arXiv:1902.10200v4 [cs.CV] 26 Jan 2020
2. Semantic Understanding of Scenes through the ADE20K Dataset Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso & Antonio Torralba Semantic Understanding of Scenes through the ADE20K Dataset, arXiv:1608.05442v2 [cs.CV] 16 Oct 2018
3. Shanu Kumar, Shubham Atreja, Anjali Singh & Mohit Jain Adversarial Adaptation of Scene Graph Models for Understanding Civic Issues, arXiv:1901.10124v1 [cs.AI] 29 Jan 2019
4. Tao Hu, Xinyan Zhu, Wei Guo Shaohua Wang & Jianfeng Zhu Human Action Recognition Based On Scene Semantics. Oct 2019. Published on-line: <https://doi.org/10.1007/s11042-017-5496-x>
5. Linjie Li, Zhe Gan, Yu Cheng, Jingjing Liu Relation-Aware Graph Attention Network for Visual Question Answering. <https://doi.org/10.48550/arXiv.1903.12314>
6. Vincent S. Chen, Paroma Varma, Ranjay Krishna, Michael Bernstein, Christopher Re, Li Fei-Fei Scene Graph Prediction with Limited Labels, Nov 2019 <https://doi.org/10.48550/arXiv.1904.11622>
7. Neuroinformatics and Semantic Representations. Theory and Applications. Alexander Kharlamov & Maria Pilgun eds. 317 P. Cambridge Scholars Publishing. 2020
8. Raymond P. Kesner, Edmund T. Rolls A Computational Theory Of Hippocampal Function, And Tests Of The Theory: New developments. Neuroscience and Biobehavioral Reviews 48. Pp. 92-147. (2015)
9. Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, Jian Sun Unified Perceptual Parsing for Scene Understanding. 2018. <https://doi.org/10.48550/arXiv.1807.10221>
10. Ivan Smirnov, Maksim Stankevich, Yulia Kuznetsova, Margarita Suvorova, Daniil Larionov, Elena Nikitina, Mikhail Savelov, and Oleg Grigoriev TITANIS: A Tool for Intelligent Text Analysis in Social Media. RCAI 2021, LNAI 12948, pp. 232–247, 2021

СЕКЦИЯ 7

**ИНТЕЛЛЕКТУАЛЬНЫЕ АГЕНТЫ,
РОБОТЫ И КОЛЛЕКТИВЫ
РОБОТОВ**

УДК 681.51

ЗАДАЧА ЛОКАЛИЗАЦИИ МИНИРОБОТА С ИСПОЛЬЗОВАНИЕМ SOSL МЕТОДА

*Добрынин Д.А., к.т.н., с.н.с.
ФИЦ ИУ РАН, Москва
e-mail: rabota51@yandex.ru*

1. ВВЕДЕНИЕ

Одной из важнейших функций интеллектуальных роботов является способность определять свое местоположение в пространстве по отношению к окружающим предметам. В настоящее время различают глобальное и локальное позиционирование роботов в пространстве. Глобальные методы используют привязку к земной поверхности и специальным системам определения координат – GPS, GLONASS и т.п. Они хорошо работают для перемещения в воздухе и на поверхности земли. Локальное позиционирование роботов внутри помещений требует других технических решений. При нахождении робота внутри замкнутых пространств использование систем глобальной навигации становится проблематичным. Это связано с плохими условиями приема сигналов GPS и GLONASS, что обусловлено электромагнитной экранировкой внутри железобетонных зданий, а также недостаточной точностью глобальной навигации при перемещении в тесных пространствах.

Особый интерес представляют методы определения положения робота в помещениях без специальных систем позиционирования. Для

решения задач такого локального позиционирования используются различные подходы – привязка к визуальным характеристикам предметов в помещении (различные методы SLAM), измерение длины пути, пройденного колесами робота (одометрия), инерционные датчики (IMU – Inertial Measurement Unit). Инерциальная навигация устойчиво работает только на небольших промежутках времени. Высокий уровень шума акселерометров, которые используются для измерения ускорения, приводит к быстрой потере точности измерения расстояния и скорости. Одометрия чувствительна к проскальзыванию колес робота, неровностям поверхности и деформации шин. Все эти факторы дают быстрое нарастание погрешности измерения пути и приводят к потере точности позиционирования. Следует отметить, что некоторые шасси роботов – без поворота колес, с гусеницами или омниколесами, шагающие роботы не могут использовать одометрию. Одометрия и инерциальная навигация требует мало ресурсов для своей работы, поэтому эти виды локализации часто используются для роботов с дифференциальным колесным приводом.

Технологии SLAM (Simultaneous Localization and Mapping) используются для локализации и одновременного построения карты территории. В настоящее время разработано множество различных методов SLAM, доступных в виде готовых пакетов ROS [1, 2]. Для измерения расстояния используются лидары, 3D камеры с измерением глубины, одометрия и т.п. Карты также представляются различными способами – в виде сеток, графов и т.п. [3, 4]. Для использования SLAM требуются значительные вычислительные ресурсы, как для обработки информации, так и для хранения карты. Эта особенность делает проблематичным использование методов SLAM для небольших роботов, обладающих скромными вычислительными средствами.

К альтернативным способам локализации можно отнести методы сегментации изображений с дальнейшей обработкой с помощью нейронных сетей. Например, в работе [5] используется алгоритм визуальной локализации с использованием граней изображений как основного характера визуального признака. Этот метод требует графических сопроцессоров для обработки изображений в реальном времени и мощного компьютера для обучения и работы нейронной сети.

Важным направлением разработки методов локализации является снижение вычислительной сложности и используемых ресурсов. В данной работе предпринята попытка построения модели такой системы локализации с применением недорогих датчиков

измерения расстояния. В качестве системы принятия решений используется интеллектуальная SOSL система.

2. ЛОКАЛИЗАЦИЯ РОБОТА В ПОМЕЩЕНИИ

Для решения задачи локализации – определения положения робота в помещении выберем следующие допущения:

используются датчики расстояния, обеспечивающие полный круговой обзор;

для выделения направления используется компас. На роботе можно использовать либо магнитный компас, либо твердотельный гироскоп с высокой стабильностью;

для построения карты (во время обучения системы) используется дополнительная система позиционирования.

На рис. 1 приведена общая схема позиционирования робота в помещении. Вся площадь помещения разбивается внешней системой позиционирования на прямоугольную сетку с некоторым шагом. На роботе установлен вращающийся датчик расстояния, способный определять расстояния до стен и препятствий 1, 2 и 3. Робот отсчитывает угол на препятствие от выделенного положения в направлении против часовой стрелки.

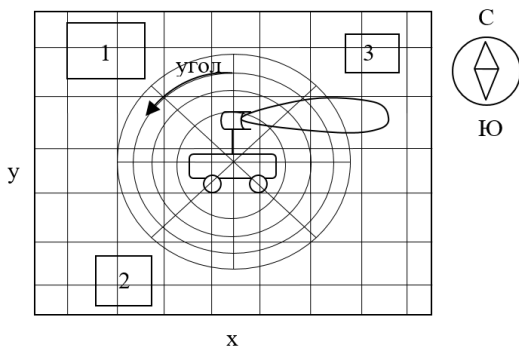


Рис. 1. Схема помещения и позиционирование робота в нем

Примерная зависимость расстояния, определяемая датчиком, приведена на рис. 2. Она зависит от положения робота по отношению к стенам помещения и окружающим его объектам. Провалы на графике соответствуют обнаруженным препятствиям. Расстояния до стен также меняются в зависимости от угла. При дискретном сканировании пространства в итоге получается набор точек. Плотность точек зависит от используемых датчиков. Для инфракрасных

дальномеров и ультразвуковых датчиков, скорость сканирования которых невелика, плотность точек низкая. Для лазерных дальномеров плотность точек существенно выше.

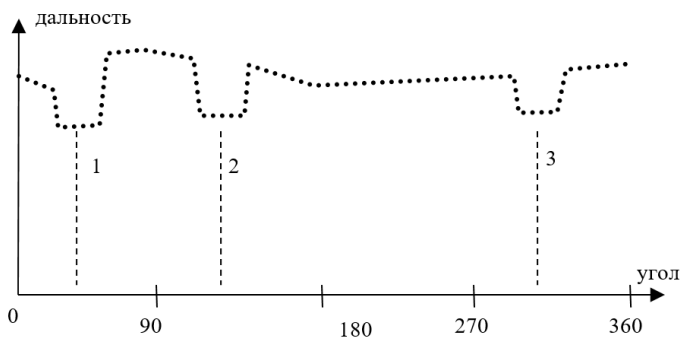


Рис. 2. Зависимость дальности от угла поворота датчика

Следует отметить, что вид зависимости расстояния изменяется при перемещении робота внутри помещения и для каждой области является уникальным. Эту особенность можно использовать для локализации робота, используя интеллектуальную систему на основе SOSL метода.

Компас используется для выделения направления при сканировании объектов. Если не использовать компас, то возможно восстановление ориентации робота по сигналам окружающей обстановки. Заметим, что это требует дополнительных вычислительных затрат, и в некоторых случаях в сетке локализации могут возникнуть области с одинаковыми кривыми расстояния (они будут сдвинуты по углу относительно друг друга). Этот случай локализации в данной работе не рассматривается.

3. ОБУЧАЕМАЯ SOSL СИСТЕМА

SOSL метод (Similarity Operation Supervised Learning) является развитием динамического ДСМ метода, предложенного в [6, 7]. В отличие от классического ДСМ метода [8], который работает с замкнутым множеством исходных примеров и заранее определенными их свойствами, SOSL метод позволяет работать в открытой среде с неизвестным заранее количеством примеров.

SOSL система работает в двух режимах [7]:

– режим обучения, когда происходит заполнение базы примеров (множество обучающих примеров) и генерируются гипотезы, составляющие базу знаний. В этом режиме система

использует внешнюю систему определения координат для формирования гипотез о локальном положении робота;

– рабочий режим, когда полученные ранее гипотезы используются для определения принадлежности к области локализации.

Для эффективной работы SOSL метода необходимо дискретизировать зависимость расстояния от угла и диапазон углов (рис. 2). Один из возможных способов представления примера e_i :

$$\begin{aligned} e_i &= (X_i, u^i) \\ X_i &= \{[d_{\min}, d_{\max}]\} \\ u^i &= \langle x, y \rangle \end{aligned}$$

где X_i – упорядоченное множество диапазонов расстояний,

u^i – область локализации,

$[d_{\min}, d_{\max}]$ – интервал расстояний от d_{\min} до d_{\max} ,

x – индекс текущего местоположения робота по горизонтали,

y – индекс текущего местоположения робота по вертикали.

Угловой диапазон сканирования датчика расстояний разбивается на N интервалов. Множество интервалов X_i будет представлять собой N элементов вида $[d_{\min}, d_{\max}]$, которые определяют диапазон изменения расстояний для каждого интервала по углу.

Сам процесс обучения SOSL-системы выглядит следующим образом (рис. 3):

– внешняя измерительная система определяет положение робота, затем принимается решение, к какой области локализации он относится в текущий момент. Информация с системы измерения представляет собой множество расстояний для каждого угла. Вся эта информация подается в SOSL-систему, при этом формируется т.н. «обучающий пример», который SOSL-система заносит в базу примеров.

– далее пример подается в решатель, который обновляет существующие гипотезы и генерирует новые. При совпадении областей локализации u^i примера и гипотезы происходит расширение диапазонов $[d_{\min}, d_{\max}]$ для всех элементов множества X_i . Таким образом, набирается информация обо всех возможных значениях дальности при нахождении робота в одном квадрате локализации. Если пример содержит новую уникальную область локализации, то происходит формирование новой гипотезы, соответствующей этой области.

– пополнение базы примеров и получение новых гипотез производится до тех пор, пока работает режим обучения.

Отметим, что режим обучения SOSL системы эквивалентен работе простого несимметричного ДСМ метода [8], когда присутствуют только положительные гипотезы.

Внешняя система локализации может использовать те же датчики расстояния, которые стоят на мобильном роботе. Главным условием работы этой системы является необходимость обеспечения нужной точности измерения расстояния при проведении обучения. Функционирование такой системы в данной работе не рассматривается.

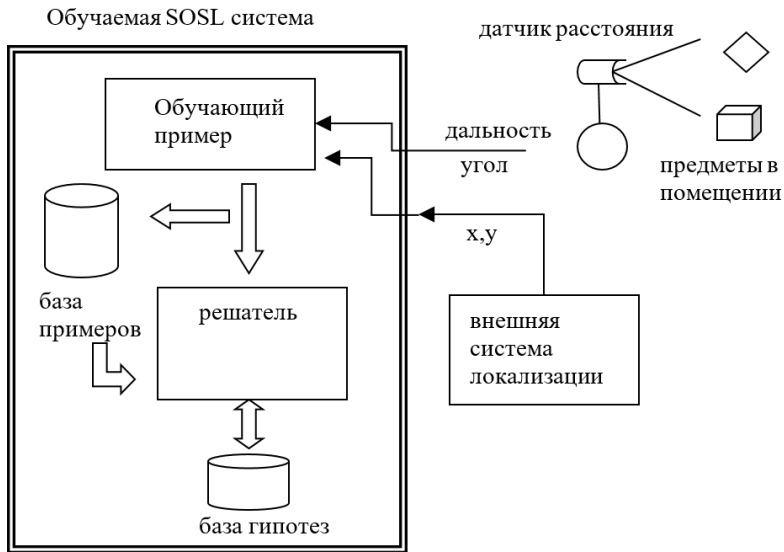


Рис. 3. Обучение SOSL системы

Рабочий режим определения местоположения робота работает следующим образом:

- измерительная система робота сканирует окружающее пространство с помощью датчика расстояний. Для каждого дискретного значения угла измеряется дальность. После завершения сканирования формируется тестовый вектор, содержащий расстояния до объектов для всего диапазона углов.

- Тестовый вектор передается в решатель SOSL системы, который проверяет его на вхождение в сформированные на этапе обучения гипотезы. Вложение проверяется для всех элементов гипотезы по правилу:

$$(d_{\min} \leq d) \text{ AND } (d \leq d_{\max}),$$

где d – значение расстояния для элемента тестового вектора.

–После нахождения гипотезы, для которой найдено полное вложение, выдается координаты ее области локализации.

Если найдено более одной гипотезы, для которых выполняются условия вложения, то это означает, что робот находится на границе областей. Это позволяет повысить точность определения координат робота в помещении.

Если не найдено ни одной гипотезы, то это означает, что робот попал в неисследованную область. При полном покрытии (полной карте) такие случаи не могут происходить.

4. ОЦЕНКА ПРИМЕНИМОСТИ ДАТЧИКОВ РАЗЛИЧНОГО ТИПА

Для построения системы локализации необходимо использовать датчики расстояний. На сегодняшний день можно выделить три основных класса таких датчиков: инфракрасные дальномеры (ИК), ультразвуковые дальномеры (УЗ), лазерные дальномеры (лидары).

Инфракрасные дальномеры измеряют расстояние до объектов в пределах от 5 см до 100 см. Их применяют для обнаружения препятствий в ближней зоне робота. Для сканирования помещений ИК датчики обычно не используются из-за малой дальности обнаружения препятствий.

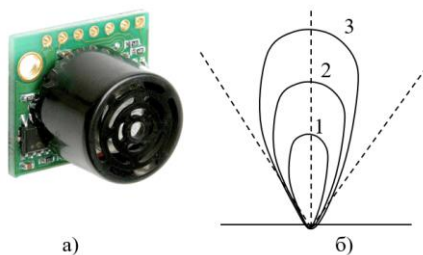


Рис. 4. а) внешний вид датчика MaxSonar; б) типовая диаграмма направленности (1 – объект 5 мм, 2 – объект 2 см, 3 – объект 10 см)

УЗ датчики являются самым дешевым видом оборудования для измерения средних расстояний. Использование таких датчиков является безопасным, поскольку уровни звукового давления, генерируемые датчиками, имеют малую величину. Диапазоны расстояний, которые измеряют УЗ датчики, лежат в пределах от 10 см до 10 м. УЗ датчики довольно просто комплексированы в группы. Заметим, что из-за особенностей распространения звука

одновременная работа датчиков затруднена. Типичным представителем УЗ датчиков для небольших роботов является изделия фирмы MaxSonar [9] (рис.4).

Особенностями приема сигнала УЗ датчиками является зависимость дальности и ширины диаграммы направленности от размеров и формы сканируемого объекта. На рис. 4б приведена типовая диаграмма направленности для датчика MaxSonar. Ширина самого большого лепестка диаграммы составляет 55 градусов. Из нее следует, что объекты больших размеров обнаруживаются датчиков на больших расстояниях и ширина диаграммы направленности шире, чем для малых. Для дальнейшей зоны ширина диаграммы направленности составляет 6 градусов [9].

Скорость сканирования УЗ датчиками определяется скоростью распространения звука в воздухе. Типовое значение – 20 миллисекунд на одно измерение дальности. УЗ датчики можно использовать для сканирования помещений, но при этом необходимо учитывать особенности их работы.



**Рис. 5. а) Промышленные дальномеры Velodyne;
б) дальномер для роботов Slamtec RPLidar A1**

Лазерные дальномеры измеряют расстояние до объектов на дальностях от 20 см до 100-300 м. Этот вид датчиков широко используется для сканирования помещений и открытого пространства. Цена таких систем, особенно профессионального класса (рис. 5а), является очень высокой.

Например, лазерный лидар Velodyne VLP-32 сканирует окружающее пространство на расстоянии до 200 метров 32-мя лучами на 360 градусов. Скорость вращения составляет до 20 оборотов/сек. Скорость выдачи информации составляет 0.6 миллиона точек в секунду [10]. Для обработки дальномерной информации с такого лазерного дальномера требуется производительный компьютер.

В последнее время появились лазерные дальномеры китайского производства, которые имеют существенно меньшую цену (рис. 5б). Типичным представителем является лидар для роботов RPLidar A1 компании Slamtec. Он сканирует окружающее пространство на расстоянии от 0.15 до 12 м одним лучом на 360 градусов со скоростью от 2 до 10 оборотов/сек. Частота сканирования составляет 8 кГц [11]. При средней скорости вращения 5 оборотов/сек лидар выдает 1600 точек на один оборот. Однако из-за особенностей измерения ширина диаграммы направленности составляет около одного градуса.

Из рассмотренных выше систем для определения расстояния для небольших мобильных роботов, имеющих на борту не слишком производительные компьютеры, подходят следующие системы:

– ИК дальномеры (5 – 100 см) подходят только для определения препятствий в ближней зоне;

– УЗ датчики (10 см – 10 м) подходят для сканирования помещений, низкая скорость передачи данных, низкая стоимость;

– лидары (50 см – 30 м) подходят для сканирования помещений, высокая скорость передачи данных, высокая стоимость.

5. ОЦЕНКА ВЫЧИСЛИТЕЛЬНЫХ РЕСУРСОВ

Оценим необходимые вычислительные ресурсы по объему памяти и количеству операций сравнения целых чисел.

Пусть для хранения расстояния используется 16 разрядное значение (2 байта). Объем памяти для хранения одной гипотезы:

$$V_1 = N * 2 * 2 \text{байта}$$

где N – количество интервалов углов.

Примем такое количество интервалов, чтобы один интервал занимал половину ширину диаграммы направленности датчика расстояния.

Общее количество областей локализации

$$N_s = N_x * N_y$$

где N_x и N_y – количество областей по осям X и Y для помещения (рис. 1).

Общая память для хранения всей карты

$$V_s = V_1 * N_s$$

Выберем размер области локализации равным половине длины мобильного робота. Для небольшого робота 40x40 см и размерах помещения 5x4 м получаем:

$$N_x = 5 \text{ м} / 0.2 \text{ м} = 25, \quad N_y = 4 \text{ м} / 0.2 \text{ м} = 20$$

$$N_s = 25 * 20 = 500$$

Количество операций сравнения

$$N_{eq} = 2 * N/2 * N_s/2$$

где $N/2$ – среднее количество операций сравнения для одной гипотезы, $N_s/2$ – среднее количество гипотез для сравнения.

В Табл.1 приведены некоторые расчетные значения объемов памяти, необходимых для хранения карты при указанных ранее условиях.

Таблица 1. Расчетные значения объема памяти

| Тип датчика | Ширина ДН, град. | Количество точек на 360^0 , N | Объем одной гипотезы, V_1 , байт | Объем карты, V_s , байт | Кол-во операций сравнения, N_{eq} |
|-------------|------------------|---------------------------------|------------------------------------|---------------------------|-------------------------------------|
| УЗ | 6 | 120 | 480 | 240 000 | 30 000 |
| УЗ | 10 | 72 | 288 | 144 000 | 18 000 |
| УЗ | 20 | 36 | 144 | 72 000 | 9 000 |
| Лидар | 2 | 360 | 1440 | 720 000 | 90 000 |
| Лидар | 1 | 720 | 2880 | 1 440 000 | 180 000 |
| Лидар | 0.5 | 1440 | 5760 | 2 880 000 | 360 000 |

При выборе разрешения карты следует учитывать несколько факторов. При низком разрешении карты помещений достигается высокая скорость поиска по карте и требуется небольшой объем памяти для хранения карты. Однако точность позиционирования робота может быть недостаточной для выполнения им своих задач. При высоком разрешении карты наблюдаются обратные эффекты – скорость поиска местоположения робота резко падает, а объем памяти для хранения быстро растет. Из этого следует, что для эффективной работы системы позиционирования необходимо выбирать оптимальное разрешение карты.

Для оценки быстродействия необходимо знать скорость выполнения операции сравнения для используемого вычислительного устройства. Известно, что для разных платформ на микроконтроллерах и ARM процессорах многие машинные команды выполняются за один такт процессора. Тогда примерная оценка времени операции сравнения $a[i] >= b[j]$ принимает вид:

- вычисление адреса элемента массива $i*2 + adr_a - 2$ такта
- загрузка регистра из памяти – 1 такт
- вычисление адреса элемента массива $j*2 + adr_b - 2$ такта
- загрузка регистра из памяти – 1 такт
- сравнение – 1 такт
- условный переход – 1 такт

– модификация счетчиков массивов – 2 такт

всего 10 тактов.

Примем среднее время выполнения 10 тактов. Оценим общее время поиска для следующих систем: ATMEGA128 16МГц, STM32 150 МГц, Raspberry Pi 1.0 ГГц (Табл. 2). Для ATMEGA128 количество тактов на операцию сравнения в два раза больше, поскольку это 8-ми разрядный микроконтроллер, а сравниваются 16 разрядные числа.

Таблица 2. Оценка быстродействия

| Тип датчика | Ширина ДН, град. | Количество точек на 360^0 , N | ATMEGA128 16МГц, тактов, время | STM32 150 МГц, тактов, время | Raspberry Pi 1 ГГц, тактов, время |
|-------------|------------------|---------------------------------|--------------------------------|------------------------------|-----------------------------------|
| УЗ | 6 | 120 | 600 000 37.5 мс | 300 000 2 мс | 300 000 0.3 мс |
| УЗ | 10 | 72 | 360 000 22.5 мс | 180 000 1.2 мс | 180 000 0.18 мс |
| УЗ | 20 | 36 | 180 000 11.2 мс | 90 000 0.6 мс | 90 000 0.09 мс |
| Лидар | 2 | 360 | 1 800 000 112 мс | 900 000 6 мс | 900 000 0.9 мс |
| Лидар | 1 | 720 | 3 600 000 225 мс | 1 800 000 12 мс | 1 800 000 1.8 мс |
| Лидар | 0.5 | 1440 | 7 200 000 450 мс | 3 600 000 24 мс | 3 600 000 3.6 мс |

Из приблизительных расчетов быстродействия видно, что предлагаемый метод способен обеспечить приемлимое быстродействие даже на маломощных микроконтроллерах и ARM процессорах.

6. ЗАКЛЮЧЕНИЕ

Для эффективного перемещения мобильных роботов внутри помещений необходима система навигации, требующая небольших вычислительных ресурсов. Предложенный метод локализации на основе датчиков сканирования и интеллектуального SOSL метода позволяет построить один из основных компонентов такой системы – систему определения положения робота в помещении. Использование SOSL метода позволяет эффективно обрабатывать информацию и проводить определение координат робота за короткое время, используя вычислительное оборудование небольшой производительности.

Точность локализации должна выбираться, исходя из задач, которые должен решать мобильный робот. SOSL метод позволяет выбрать необходимую точность локализации, при этом снижая требования к вычислительным системам мобильного робота. Такая система локализации может быть использована на малых мобильных роботах, не обладающими мощными вычислительными средствами.

Для практического применения предложенного метода необходимо моделирование процессов обучения и локализации, а также практическая проверка предложенных решений на мобильном роботе. Эти этапы предполагается выполнить в процессе дальнейшей работы.

Литература

1. 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 // J Field Robotics. – 2019. – Vol. 35. – P. 416–446.
2. Labbé M., Michaud F. Online Global Loop Closure Detection for LargeScale Multi-Session Graph-Based SLAM // IEEE/RSJ International Conference on Intelligent Robots and Systems, 2014. – P. 2661–2666.
3. Gmapping [Электронный ресурс] URL: <http://wiki.ros.org/gmapping>
4. Google Cartographer ROS [Электронный ресурс] URL: <https://google-cartographerros.readthedocs.io/en/latest/#>
5. Buyval A., Gavrilin M., Magid E. A multithreaded algorithm of UAVvisual localization based on a 3D model of environment: implementation with CUDATEchnology and CNN filtering of minor importance objects // In Proceedings of the 2017 International Conference on Artificial Life and Robotics (ICAROB 2017). – 2017, Miyazaki, Japan. – P.356–359.
6. Добрынин Д.А. Динамический ДСМ-метод в задаче управления интеллектуальным роботом // Десятая национальная конференция по искусственному интеллекту КИИ-2006, 25-28 сентября 2006 г., Обнинск, Труды конференции. – М.: Физматлит, 2006. – Т.2.
7. Добрынин Д.А. Задача локализации беспилотного транспортного средства с использованием ДСМ-метода // Пятый Всероссийский научно-практический семинар «Беспилотные транспортные средства с элементами искусственного интеллекта» (БТС-ИИ-2019): Труды семинара. – Переславль-Залесский: Российская ассоциация искусственного интеллекта, 2019. – С. 48-55.
8. Автоматическое порождение гипотез в интеллектуальных системах / под. общ. ред. В.К.Финна. – М.: URSS, 2020. – 528 с.
9. Ультразвуковой сонар компании MaxBotics [Электронный ресурс] URL: https://www.maxbotix.com/ultrasonic_sensors/mb1000.htm
10. Лидары компании Velodyne [Электронный ресурс] URL: <https://velodynelidar.com/products>
11. Лидар SlamTec RPLidar A1 [Электронный ресурс] URL: <https://www.slamtec.com/en/Lidar/A1Spec>

МОДЕЛЬ ПРОСТРАНСТВЕННОЙ ПАМЯТИ АНИМАТА

*Карпов В.Э., к.т.н., доцент, зав. лабораторией
НИЦ "Курчатовский институт"
e-mail: karpov-ve@yandex.ru*

1. ВВЕДЕНИЕ

Задача запоминания и описания маршрута движения с его последующей обработкой является одной из базовых в мобильной робототехнике. Решается она с помощью самых разнообразных подходов. В простейшем случае – это обработка множества маршрутных точек с использованием средств глобальной и локальной навигации, вплоть до одометрии. Весьма активно применяются методы семейства т.н. SLAM (Simultaneous localization and mapping – одновременная локализация и построение карты). Сравнительные характеристики некоторых методов SLAM см., например, в [1] и [2], а достаточно показательный общий обзор представлен в [3]. Есть также работы, в которых запоминание и обработка маршрута осуществляется на основе обучения нейронных сетей. Так, в работе [4] рассматривается алгоритм навигации мобильного робота на основе рекуррентных нейронных сетей. Альтернативным подходом является абстрактное представление маршрута, оперирующего множеством ориентиров. В этом случае маршрут представляет собой множество описаний сцен, которые должен распознать робот. При этом обработка двигательных функций в той или иной сцене определяется механизмами нижнего уровня – например, конечными автоматами, типичными задачами которых является обход объектов-ориентиров справа/слева/между и пр., [5]. Этот подход характерен для задач мобильной робототехники, в которых важна, прежде всего, поведенческая составляющая.

В отличие от представленных выше подходов нас интересует схема организации памяти универсального характера, позволяющая решать не только навигационные задачи, но и хранить информацию произвольной семантики.

2. ПОСТАНОВКА ЗАДАЧИ

Рассмотрим следующую задачу. Пусть робот снабжен парой датчиков-локаторов, способных определить объекты, попадающие в зоны их действия – левый и правый секторы. Кроме того, робот оснащен еще одним датчиком – компасом. Условно это изображено на Рис. 1. Задача заключается в том, чтобы робот, пройдя по маршруту, запомнил его и смог бы впоследствии воспроизвести.

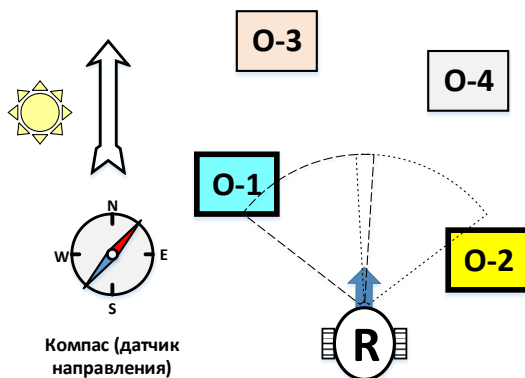


Рис. 1. Условная схема органов чувств робота. В зону видимости робота R попали объекты $O-1$ и $O-2$

Простоты ради будем полагать, что детекторы-локаторы определяют лишь наличие обнаруженных объектов левым и правым "глазом". Наличие "компаса" не является искусственным допущением. Так, многие насекомые могут определять направление движения, используя сенсор поляризации солнечного света, см., например, [6]. Пусть робот может распознавать ограниченное число объектов $O = \{o_i\}$, а сенсорная система робота может быть представлена множеством элементарных детекторов, каждый из которых отвечает за регистрацию элементарного события (левый глаз видит объект o_1 , правый – o_2 , угол ориентации равен α_2 и т.п.). Компас также может быть дискретным, определяя угол направления, скажем, в румбах. При таких соглашениях будем считать, что робот оснащен множеством двоичных датчиков. Это означает, что наблюдаемая в момент времени сцена представлена множеством кадров, каждый из которых – это вектор

$$S = \{s_i\}, s_i=0,1, i=1..n \quad (1)$$

Здесь и далее мы будем использовать следующую терминологию: кадр – это текущее состояние сенсорных элементов, а

сцена – это уже некоторая последовательность кадров вместе с тем, что можно назвать контекстом (далее в роли контекста будет выступать множество ассоциативных вершин и вершины-счетчики, об этом ниже). Пусть робот движется по некоторому маршруту, пример которого представлен на Рис. 2,а. На Рис. 2,б представлен тот же маршрут с точки зрения сенсорной системы робота.

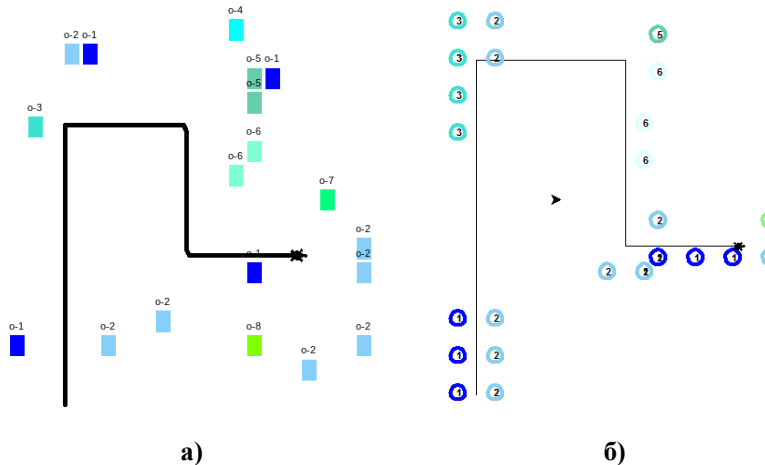


Рис. 2. а) Реальный маршрут движения робота, б) маршрут с точки зрения сенсорной системы робота

Видно, что часть объектов не попадает в зону действия сенсорной системы, а многие объекты продублированы. Это связано с тем, что во время движения наблюдаемые объекты какое-то время остаются в зоне видимости.

3. ОРГАНИЗАЦИЯ ПАМЯТИ

Рассмотрим устройство памяти робота. Во избежание ненужных ассоциаций с биологическими и искусственными нейронными сетями, будем использовать слово "элемент". Память состоит из элементов 4-х типов: сенсорные элементы $S=\{s_i\}$, ассоциативные $T=\{t_i\}$, элементы-счетчики $C=\{c_i\}$ и эффекторные (или моторные) элементы $A=\{a_i\}$. Все эти элементы могут находиться в одном из двух состояний – 0 или 1.

Сенсорные элементы отвечают за регистрацию текущего наблюдения (кадра) согласно (1). Эффекторные элементы определяют производимое в данный момент времени действие (движение вперед, поворот налево/направо). Элементы t_i отвечают за процессы формирования ассоциаций (это – рабочие элементы), а элементы-

счетчики отвечают за представление времени в системе. Связи между элементами формируются динамически. При этом ассоциативные элементы возбуждаются только тогда, когда возбуждены все входящие в них связи (активны источники), а само возбужденное состояние сохраняется в течение некоторого времени. Длительность этого периода остаточной активности, как будет видно далее, играет важную роль при воспроизведении маршрута.

Описание процесса запоминания. Изначально все элементы ассоциативного типа находятся в невозбужденном состоянии. Более того, нет связи как между ними самими, так и между ними и прочими элементами. В некоторый момент времени возбуждается подмножество сенсорных вершин. Среди множества ассоциативных элементов ищется такой, у которого есть связь со всеми возбужденными элементами сенсорного слоя. Если таковой не найден, то из множества свободных ассоциативных элементов выбирается некоторый, и ему назначаются связи с текущими активными элементами сенсорного слоя. При этом сформированный элемент получает тип – ассоциацию 1-го уровня.

В общем виде элементы ассоциативного слоя T описываются тройкой вида

$$t = \langle I, L, a \rangle \quad (2)$$

Здесь I – список вершин сенсорного уровня, L – уровень элемента, a – значение активности (0, 1). Для элементов первого уровня ассоциаций $I = \{s_i / s_i \in S, s_i = 1\}$, $L = 1$. Будем обозначать такие элементы как t^1_i , где верхний индекс указывает на их принадлежность к уровню 1. Таким образом, элементы t^1 отвечают за представление текущего наблюдаемого кадра.

Фактически, здесь происходит отображение множества сенсоров S в множество элементов 1-го уровня $L1 = \{t^1_i\}$. Запишем это в операторной нотации:

$$L1 = F(S) \quad (3)$$

Поскольку активность ассоциативных вершин сохраняется в течение некоторого времени, то одновременно могут быть активны несколько вершин уровня $L1$. Аналогичным образом формируются ассоциативные вершины уровня 2:

$$L2 = F(L1) \quad (4)$$

Если вершины уровня $L1$ – это мгновенное состояние сенсорной системы, то ассоциации уровня 2 определяют уже комбинации (но не последовательности) кадров.

Для определения последовательности наблюдаемых роботом кадров используются специальные элементы-счетчики. Здесь мы

подходим к вопросу представления времени. Разумеется, можно ввести элементы, которые будут возбуждаться циклически через некоторые интервалы времени. Некоторые аналоги таких вершин-счетчиков наблюдаются в природе. Так, нейробиологи говорят о наличии специальных временных клеток (time cells), отвечающих за организацию внутренних часов, см. [7], [8]. Интересно, что в работе [9] отмечается роль временных нейронов для устранения неоднозначностей при прохождении маршрута крысами. Наличие внутренних "часов", конечно, весьма упрощает рассуждения и схему работы, однако чревато появлением дополнительных сложностей и искусственных допущений, связанных с определением периода таймера, привязкой к двигательной активности и пр. Поэтому здесь мы можем попробовать пойти иным путем. Вообще говоря, представление о времени определяется наступлением некоторых внешних или внутренних событий или регистрацией происходящих изменений. Эти события могут возникать неравномерно, но сути это не меняет: если нет регистрируемых изменений, то нет времени. В предлагаемой модели элементы-счетчики c_i последовательно возбуждаются по наступлению событий – изменению состояния множества вершин уровня 2. Это не противоречит нейробиологическим моделям, в которых описывается наличие активности пространственно-временных нейронов при регистрации внешних событий (изменений) [9].

Элементы третьего уровня ассоциаций $L3$ активируются элементами $L1$, сигналами от счетчиков C и эффекторных элементов A :

$$L3 = F(L2 \cup C \cup A) \quad (5)$$

Условно такая схема образования ассоциаций представлена на Рис. 3,а.

Реальная схема приведена на Рис 3,б. На ней изображено состояние памяти в некоторый момент времени. Черными линиями показано имеющееся возбуждение сети – текущая активность. Таким образом, в ходе следования по маршруту формируется его образ в виде множества связей между элементами ассоциативного слоя.

Воспроизведение маршрута. Если при запоминании маршрута элементы уровня $L3$ определялись комбинацией элементов уровней C , $L2$ и A , то при его воспроизведении, в отсутствие возбуждения A (задача воспроизведения как раз и заключается в определении требуемой активности элементов A), определяются вершин-кандидаты $L3'$, которые могли бы быть активированы элементами C и $L2$ в данный момент времени. Далее определяется множество вершин A' , которые связаны с активными вершинами слоя $L3'$. При этом,

естественно, множество A' может содержать более одного элемента (конфликтное множество) или быть пустым (неопределенность ситуации). Это связано с возможной неоднозначностью воспроизведения вследствие недостаточности элементов памяти T и/или элементов-счетчиков. Итак, воспроизведение маршрута заключается не столько в однозначном определении выполняемых действий, а в определении моментов времени, когда возникают некоторые события, связанные с необходимостью выбора действия.

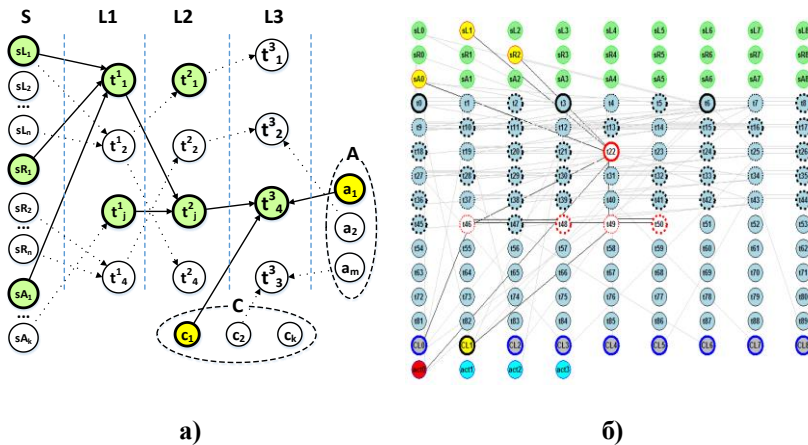


Рис. 3. а) иллюстративная схема формирования ассоциаций: вершины с именами s^* - сенсорные элементы (уровень S), t_i^j - рабочие ассоциативные элементы (уровни $L1, L2, L3$), c_i - счетчики, a_i - эффекторные элементы; б) реальная схема: верхние 3 ряда – сенсорные элементы, нижний ряд ($act1, \dots, act4$) – эффекторные элементы, элементы $CL0, \dots, CL8$ – элементы-счетчики

Схема воспроизведения маршрута выглядит так:

Шаг 1. Прямая активация элементов

- сформировать значения элементов сенсорного слоя S : $S = E(Input)$
- сформировать ассоциации уровня $L1$: $L1 = F(S)$
- сформировать ассоциации уровня $L2$: $L2 = F(L1)$

Шаг 2. Формирование перечня вершин-кандидатов $L3$

Определить множество вершин уровня $L3$, возбужденных элементами C и $L2$:

$$L3' = F_d(C \cup L2) \text{ (функция } CreateDirectLinks(C \cup L2, L2))$$

Шаг 3. Формирование ассоциативных обратных связей слоя эффекторов

Сформировать множество возможных действий A'

$A' = F_b(L3')$ (функция *CreateBackLinks*($L3'$, A'))

Псевдокод функций *CreateDirectLinks* и *CreateBackLinks* приведен ниже. Здесь предполагается, что, в соответствии с (2), каждый элемент задается списком элементов-источников *connections*, уровнем *level* и своим состоянием активности *active*.

```
-- Определение вершин уровня secondary_level, возбуждаемых
-- элементами уровня primary_level_list
-- [primary_level_list] → secondary_level
function CreateDirectLinks(primary_level_list, secondary_level)
  for each e in Elements do
    if e.level ≠ secondary_level then continue
    oklist[i] ← False for each i in primary_level_list
    for each c of e.connections do
      for each i in primary_level_list do
        if c.active>0 and c.level=l then oklist[i] ← True
      endfor
    endfor
    ok ← True
    for each i in primary_level_list do
      if not oklist[i] then ok ← False
    endfor
    if ok then e.active ← ACTIVE_TIME_NUM
  endfor
end
```

```
-- Определение вершин уровня secondary_level,
-- которые являются источниками для вершин уровня primary_level
-- primary_level → secondary_level
function CreateBackLinks(primary_level, secondary_level)
  for each e in Elements do
    if e.level ≠ primary_level or e.active=0 then continue
    for each t of e.connections do
      if t.level=secondary_level then t.active←ACTIVE_TIME_NUM
    endfor
  endfor
end
```

Очевидно, что неоднозначность распознавания текущей ситуации зависит от умения "считать" (т.е. от количества элементов-счетчиков), а также от количества ассоциативных элементов.

Рассмотрим пример. При запоминании маршрута в моменты времени 927, 1333 и 1764 робот совершал действия (повороты), рис. 2. На рисунке эти моменты обозначены стрелками с цифрами 1, 2 и 3. На графиках отложено оцениваемое количество возможных действий робота в каждый момент времени – число активных вершин уровня A .

Значение, равное 1 – это однозначно выполняемое действие. Значение, равное 0, означает, что не возбуждена ни одна вершина уровня A (нераспознанная, незнакомая роботу ситуация). Значение, большее 1 – это ситуация неоднозначности.

Для рассматриваемого тестового маршрута номинальные значения числа элементов-счетчиков и ассоциативных элементов равны $|C_{nom}| = 18$ и $|T_{nom}| = 126$ соответственно. Этого хватает для запоминания маршрута. В зависимости от количества элементов неоднозначность проявляется так, как показано на рис. 4:

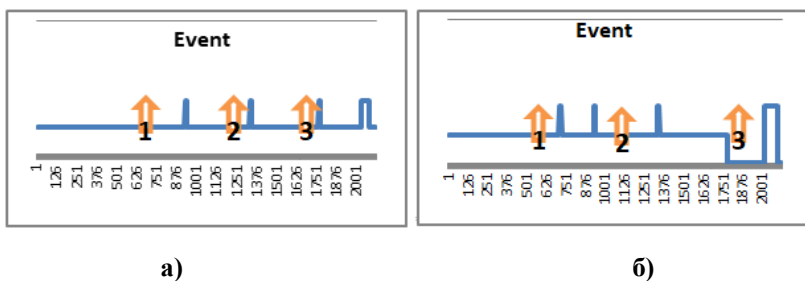


Рис. 4. Регистрируемые события. а) $|C| = |C_{nom}| = 18, |T| = |T_{nom}| = 126$, б) $|C| = 0.5|C_{nom}|, |T| = 0.5|T_{nom}|$

Ось абсцисс – время (такты), ось ординат – неопределенность выполнения действий (количество альтернатив). Нулевое значение обозначает полную неопределенность. Стрелки с цифрами 1, 2, 3 определяют моменты совершения действий (повороты) при прохождении маршрута.

Рис. 4,а иллюстрирует ситуацию с вполне приемлемым количеством элементов ассоциативного слоя и счетчиков. Здесь осталось некоторое количество свободных (незадействованных в ассоциациях) элементов, однако возникла неопределенность в районе такта 2000. Это было обусловлено недостаточностью счетчиков. Вместе с тем, наличие неопределенности – это принципиальный момент, который должен требовать специального разрешения. Так, для данного маршрута однозначные решения появляются при 162 элементах ассоциативного слоя и при 54 счетчиках. Однако простое увеличение числа элементов – это просто уход от проблемы.

На Рис. 4,б представлена совсем "плохая" конфигурация памяти, когда не хватает как ассоциативных элементов, так и элементов-счетчиков (50% от номинального количества). Здесь видно, что провалы (недостаточность информации, значение 0) начинаются в конце серии, когда элементов памяти уже не хватает.

Интересно, что во всех сериях экспериментов распределение ассоциативных элементов по уровням $L1$, $L2$, $L3$ определяется примерно как 1:2:7, т.е. большую часть составляют элементы уровня $L3$.

Разрешение неоднозначностей. Итак, предложенное устройство памяти позволяет определить моменты выбора того или иного действия. Непосредственное определение действия в случае неоднозначности основано на анализе последовательных кадров в моменты времени t и $t-1$ и определении действия a , которое приводит к смене кадров:

$$S(t) = a(S(t-1)) \quad (6)$$

Определение действия a может осуществляться на базе геометрических построений, например, исходя из преобразований координат видимых объектов при выполнении движений (поворот/вперед/назад). При этом в основе этого механизма могут лежать грубые, приблизительные оценки, в том числе – качественные, относительные, так, как это было предложено в работе [5]. В простейшем случае, если действия анимата ограничены лишь поворотами и движением прямо, а сам анимат оснащен "компасом", то задача определения действия становится примитивной. Основной вопрос при такой навигации – это определение момента поворота. Но эту задачу как раз и решает рассмотренный выше механизм воспроизведения маршрута.

Помехи и неточности восприятия. Основным неблагоприятным для навигации фактором являются, естественно, неточности измерений (восприятия среды), внешние помехи или динамичность среды в общем случае. Для данной модели это означает, что первичные возбуждения сенсорных элементов при воспроизведении маршрута не будут соответствовать тем, которые осуществлялись в процессе запоминания.

Реакция системы на наличие ошибок восприятия (несоответствие наблюдаемой сцены запомненной) зависит от периода остаточной активности вершин. Без остаточной активности маршрут воспроизводится однозначно лишь при отсутствии ошибок наблюдения. Увеличение периода активности приводит к тому, что кратковременные несоответствия не оказывают существенного влияния на запоминание событий. С другой стороны, очевидно, что причиной неоднозначности воспроизведения является именно остаточная активность вершин.

4. ЗАКЛЮЧЕНИЕ

Рассматриваемая в работе модель памяти не претендует, разумеется, на оптимальность с технической точки зрения. С решением реальных задач существующие методы справляются достаточно хорошо. Речь шла исключительно о демонстрации того, что реализация самых разных феноменов поведения возможна на основе единого механизма ассоциаций. Это особенно важно для работ в области создания множества базовых моделей для реализации феноменов социального взаимодействия искусственных агентов. Так, в рамках парадигмы моделей социального поведения было показано, что, например, феномены подражательного поведения и социального обучения могут быть рассмотрены с точки зрения ассоцианистского подхода: поведение первого лица выступает для второго в качестве различительного стимула, который запускает реакцию, ведущую к подкреплению [10]. Теперь же показано, что этот ассоцианистский подход позволяет описать и феномены памяти. Все это позволяет говорить о существовании единой базы, единого понятийного аппарата, реализующего все компоненты архитектуры социального агента.

Литература

1. Kümmerle R. и др. On measuring the accuracy of SLAM algorithms // *Auton. Robots*. 2009. № 27. С. 387–407.
2. Kuzmin M. Review. Classification and comparison of the existing SLAM methods for groups of robots // *22nd Conference of Open Innovations Association (FRUCT)*. , 2018. С. 115–120.
3. Cadena C. и др. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age // *IEEE Trans. Robot.* 2016. Т. 32. № 6. С. 1309–1332.
4. Евсеев Д.А. Алгоритм навигации мобильного робота на основе рекуррентных нейронных сетей // *Интеллектуальные системы и технологии: современное состояние и перспективы - сборник научных трудов (ISYT-2017)*. СПб.: Политехника-принт, 2017. С. 66–75.
5. Карпова И.П. К вопросу о представлении маршрута мобильного робота на основе визуальных ориентиров // *Мехатроника. Автоматизация. Управление*. 2017. Т. 18. № 2. С. 81–89.
6. Захаров А.А. Муравей. Семья. Колония. М.: Фитон XXI, 2018. 192 с.
7. Hastings M.H., Maywood E.S., Brancaccio M. The Mammalian Circadian Timing System and the Suprachiasmatic Nucleus as Its Pacemaker // *Biology (Basel)*. 2019. Т. 8. № 1. С. 13.
8. Mieda M. The Network Mechanism of the Central Circadian Pacemaker of the SCN: Do AVP Neurons Play a More Critical Role Than Expected? // *Front. Neurosci.* 2019. Т. 13. № February. С. 1–7.
9. MacDonald C.J. и др. Hippocampal «Time Cells» Bridge the Gap in Memory for Discontiguous Events // *Neuron*. 2011. Т. 71. № 4. С. 737–749.
10. Хегенхан Б., Олсон М. Теория научения. СПб.: Питер, 2004. Вып. 6. 473 с.

РЕАЛИЗАЦИЯ СИМПАТИИ В ЭМОЦИОНАЛЬНО-ПОТРЕБНОСТНОМ АГЕНТЕ С СЕМИОТИЧЕСКОЙ АРХИТЕКТУРОЙ*

Сорокоумов П.С., научный сотрудник

НИЦ «Курчатовский институт»

e-mail: petr.sorokoumov@gmail.com

Карпов В.Э., к.т.н., начальник лаборатории

НИЦ «Курчатовский институт»

e-mail: karpov_ve@mail.ru

1. ВВЕДЕНИЕ

Одной из актуальных проблем современной робототехники является создание эффективных коллаборативных систем, в которых предполагается не только совместное существование человека и робота в единой среде, но и их непосредственное взаимодействие. Это может иметь формы как постоянной работы в устойчивых группах, когда робот находится под более или менее прямым контролем человека, так и для эпизодического, спонтанного взаимодействия. При этом предполагается необходимость снижения риска взаимного непонимания, опасного как для людей, так и для роботов. Исходя из этого, для высокоуровневого управления интеллектуальными роботами, общающимися с людьми, желательно использовать достаточно простые, предсказуемые, легко воспринимаемые человеком и объяснимые (верифицируемые) модели. Обязательным их элементом должна быть способность робота выделять людей из других объектов среды, относиться к ним особенно внимательно.

В данной работе предлагается модель интеллектуального агента, позволяющая менять поведение путём логического вывода в изменяющейся обстановке, учитывая при этом как собственные потребности агента, так и поведение окружающих его людей. Делается это путём сочетания семиотической картины мира и эмоционально-потребностной модели принятия решений, дополненной механизмом эмпатии к другим агентам. При этом величина эмпатии определяется

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

степенью схожести агента-робота и наблюдаемого конспецифика (оператора). Данная система позволяет естественным образом программировать такие феномены группового поведения, как быстрое социальное обучение или частично подражательное поведение (контагиозность).

2. СЕМИОТИЧЕСКАЯ СИСТЕМА УПРАВЛЕНИЯ

Семиотическое представление данных обеспечивает совместное использование фактов и логических соотношений между ними. Один из применяемых на практике подходов – это модель школы Г.С.Осипова, в которой знак состоит из четырёх компонентов – имени n , образа p , значения t и личностного смысла a . Имя n при этом – обозначение знака, взятое из культурного контекста; с формально-логической точки зрения оно выбирается произвольно.

Под образом p понимается набор всех предикатов, которые верны для означаемого объекта. Например, для знака «дверь1» в образ могут входить предикаты *тип (дверь1, дверь)* (т.е. «дверь1 является дверью»), *цвет (дверь1, красный)* и *номер (дверь1, 18)*.

Значение t и личностный смысл a представляют собой множества МУСИН-правила вывода, то есть четверок из наборов предикатов

$$R = (C^+, C, A, D),$$

где C^+ – предикаты, которые должны иметься в модели для срабатывания правила, C – предикаты, которые не должны присутствовать в модели для срабатывания правила, A – предикаты, добавляемые в модель после срабатывания правила, D – предикаты, удаляемые из модели после срабатывания правила. При этом в правилах, в отличие от образа, могут присутствовать переменные – значения, которые могут принимать любые из имеющихся в базе атомарных величин, если это приведёт к срабатыванию правила, т.е. совпадению предикатов из C^+ с имеющимися в базе и к несовпадению с ними же предикатов из C . Ниже названия переменных везде начинаются с прописных букв. Например, правило может иметь вид:

ЕСЛИ тип (X, стол) ТО тип (X, мебель),

т.е. «если X – это стол, то X – это мебель», или

*ЕСЛИ тип(X, команда) И субъект(X, робот) И задача (робот, Y), ТО
ДОБАВИТЬ [задача (робот, X), остановить(робот, Y)] И УБРАТЬ
[задача(робот, Y)],*

что означает:

«Если робот, выполняющий команду Y, получил команду X, то необходимо начать выполнение команды X и остановить выполнение команды Y».

Правила из значения и личностного смысла в оригинальной работе Осипова и др. различаются тем, что в личностном смысле собраны закономерности, полученные из опыта взаимодействия агента с означаемыми сущностями, а в значении – результаты обобщения зависимостей между образами. Это различие основано на описанной процедуре итеративного формирования знаков [1]. Однако если интерпретировать правила значения как описания зависимостей образов, т.е. отношений, которые выполняются всегда, а правила личностного смысла – как описания действий агента в системе, т.е. отношений, которые выполняются при выполнении определённой операции, то можно построить семиотическую систему принятия решений, в которой на каждой итерации обработки агент выбирает для выполнения одно из действий, описанных в его личностном смысле. Личностные смыслы других агентов могут использоваться в такой модели для предсказания или объяснения их поведения. Таким образом, последовательность обработки данных семиотической системой управления включает следующие шаги, выполняемые циклически:

1. Обновить образы знаков системы по данным из внешних источников (например, сенсоров);
2. Выполнить все допустимые выводы из правил-значений и правил-личностных смыслов всех знаков, кроме текущего агента;
3. Получить множество всех доступных для выполнения правил-личностных смыслов текущего агента;
4. Если на шаге 3 получено пустое множество, ничего не делать; если оно непустое, выбрать какое-либо из правил в качестве действия агента на текущем шаге в соответствии с некоторой политикой выбора;
5. Выполнить выбранное правило и все допустимые дополнительные выводы, как на шаге 2.

Подобная система позволяет строить пригодные для практического использования системы управления; её достоинством является интерпретируемость и расширяемость. К ограничениям такой интерпретации знаковой системы можно отнести отсутствие процедуры порождения новых знаков, весьма существенной для концепции [1], а также неясный механизм приоритизации действий. В

целом по вычислительным возможностям эта система примерно совпадает с современными когнитивными архитектурами, например *Soar* [2].

Для определения приоритетов действий разумным представляется дополнить описанный механизм алгоритмом, позволяющим обосновывать предпочтение одного действия другим. Для этого подходит эмоционально-потребностная архитектура.

3. БАЗОВАЯ ЭМОЦИОНАЛЬНО-ПОТРЕБНОСТНАЯ АРХИТЕКТУРА

Для построения архитектуры интеллектуального агента, способной воспроизвести феномены эмоций человека, можно использовать модель П.В. Симонова [3], в которой эмоции связаны с оценкой возможности удовлетворения актуальных потребностей, исходя из анализа соотношения имеющихся и необходимых для удовлетворения этих потребностей ресурсов. Пусть агент имеет некоторое множество потребностей $N = \{N_i\}$, $i=1..r$. При этом $N_i \in [0,1]$. Чем выше эта величина, тем сильнее потребность. Например, агент может иметь потребности в питании, безопасности, общении с другими агентами, исследовании мира и т.п.

Пусть далее имеется множество доступных агенту действий $A=\{A_j\}$, $j=1..s$. Здесь предполагается, что действия являются высокоуровневыми (в общем случае они могут рассматриваться как поведенческие процедуры). Кроме того, будем считать, что каждое из этих действий направлено на удовлетворение потребностей. Степень удовлетворения потребностей опишем в матричном виде:

$$AN=\{AN_{ij}, i=1..r, j=1..s, AN_{ij} \in [0,1]$$

Величина AN_{ij} определяет вклад действия A_j в удовлетворение потребности N_i .

Выше считается, что все действия выполняются успешно, и степень удовлетворения потребностей ими известна заранее, что является излишним упрощением модели. Его можно попытаться преодолеть, если считать соответствующие параметры случайными величинами, однако обсуждение этой модификации выходит за рамки данной работы. В то же время можно использовать это свойство для «обещания» агенту перспективности удовлетворения потребности в будущем. Например, если для утоления голода ему нужно приблизиться к пище, прежде чем съесть её, то можно указать для действия «подойти к пище» некоторую ненулевую степень утоления

голода; непосредственно от приближения агент не насытится, но на следующем шаге он сможет успешно удовлетворить эту потребность.

Выбор действий в эмоционально-потребностной модели осуществляется в соответствии с множеством правил, которые позволяют по текущим потребностям агента и данным о его окружении сформировать множество принципиально осуществимых в данный момент действий. Из этого множества агент выполняет то действие, которое способно в максимальной степени удовлетворить его потребности. В качестве меры этого удовлетворения используется уровень эмоций агента E , рассчитываемый как разница между суммарным снижением потребностей в результате действия и суммой оставшихся после него потребностей:

$$E_j = \sum dN_j - \sum N_j^{new} ,$$

где

$$N_j^{new} = \max(0, N - AN_j),$$

$$dN_j = N - N_j^{new}.$$

Например, если в текущий момент четыре потребности агента имеют величины $N=[1; 0,1; 0,5; 1]$, а некоторое действие A_0 способно удовлетворить их на величины $AN_0=[1; 0,3; 0,3; 0,7]$, то после этого действия потребности станут равными $N_0^{new} = [0; 0; 0,2; 0,3]$, и их снижение окажется равным $dN_0 = [1; 0,1; 0,3; 0,7]$. Тогда общий эмоциональный уровень после действия A_0 рассчитывается как сумма элементов вектора $(dN_0 - N_0^{new})$, т.е. $2,1 - 0,5 = 1,6$.

Другими словами, когда агент выполняет действия, которые позволяют удовлетворить наиболее актуальные потребности в максимальной степени, его эмоциональный уровень растёт; когда актуальные потребности остаются неудовлетворёнными, этот уровень падает. По изменению величины E можно судить об успешности агента: постоянно отрицательная величина E говорит о невозможности удовлетворения актуальных потребностей, положительная – о наличии актуальных потребностей, которые в текущий момент успешно удовлетворяются, околонулевая – о благополучном существовании, в котором ни одной из потребностей не удаётся достичь значительных величин без её удовлетворения.

Видно, что эмоционально-потребностная архитектура даёт возможность обоснованно выбирать действия агента, то есть существенно дополняет возможности базовой семиотической архитектуры. В некотором такие рассуждения созвучны взглядам на

эмоции как на фактор, определяющий целеполагание и мотивацию агента, например рассуждениям о роли и функции эмоций Е.П. Ильина [4]. Исходя из этого, представляется целесообразным объединить семиотическую архитектуру с эмоционально-потребностным механизмом. Сделать это проще всего через встраивание семиотической процедуры поиска допустимых действий в эмоционально-потребностную модель. При этом общий алгоритм работы агента можно описать следующей последовательностью шагов:

1. Рассчитать текущие потребности N агента, исходя из его параметров состояния и сенсорных данных.

2. Пополнить семиотическую картину мира сенсорными данными и, если это требуется в правилах, сведениями о потребностях. Так как семиотическая модель в имеющейся реализации не работает напрямую с числовыми характеристиками потребностей, следует представлять их лингвистическими переменными.

3. Получить множество возможных действий A , как описано выше.

4. Выбрать действие, приводящее к наибольшему уровню эмоций E .

Такой агент естественным образом объединяет логический вывод и удовлетворение своих потребностей, что является важным достоинством. Ещё более интересное следствие из этого подхода – возможность реализации феномена симпатии между агентами, основываясь на их сходстве.

4. РЕАЛИЗАЦИЯ СИМПАТИИ В СЕМИОТИКЕ ПУТЁМ ЛОГИЧЕСКОГО ВЫВОДА О БЛИЗОСТИ СЕБЯ И АГЕНТА

Взаимодействие агентов в эмоционально-потребностной архитектуре реализуется добавлением соответствующих потребностей. Наиболее типичные их варианты для коллаборативных систем – потребность в дружеском взаимодействии и потребность в избегании враждебного или агрессивного взаимодействия (положительная и отрицательная формулировки "золотого правила" морали, [5]). К первому варианту можно отнести, например, желание делиться информацией или ресурсами; ко второму – избегание нападения. В наиболее простом виде можно реализовать такие взаимодействия в семиотической системе с помощью правил-значений, которые будут явным образом выводить признаки дружелюбности или враждебности из априорно известных признаков агентов («ЕСЛИ поза агента угрожающая И он смотрит на меня ТО он враждебен»).

Однако это решение не способно автоматически адаптироваться к изменениям как окружения, как и самого интеллектуального агента: при смене решаемой задачи понадобится перерабатывать систему правил. Более перспективным представляется автоматически выделять такие признаки, руководствуясь сходством образа текущего агента с образом других агентов. Выявление степени сходства агента с наблюдаемым объектом – это фундаментальное свойство, определяющее суть механизма эмоциональной отзывчивости, или эмпатии. В биологии это называется симпатической индукцией, и на этом механизме построено множество феноменов социального взаимодействия: когезия (стремление держаться вместе), контагиозное (заразное, т.е. «как все, так и я») поведение, феномены альтруизма/эгоизма и др.

Определения такого сходства в семиотической картине мира можно устанавливать разными способами. Наиболее простой вариант – создание специального стороннего модуля для прямого сравнения образов всех агентов с образом текущего (т.е. его «субъективным Я») и переноса их свойств на него. Более универсальный метод – создание специальных правил для такого анализа, обеспечивающих агрегацию результатов частных сравнений, т.е. их численную обработку. В простейшем случае можно представить эти правила для сходства и несходства как:

ЕСЛИ тип (X, агент) И тип (Y, агент) И Свойство (X, V) И Свойство (Y, V)

ТО совпадение(X,Y,Свойство) И совпадение (Y,X,Свойство)

ЕСЛИ тип (X, агент) И тип (Y, агент) И Свойство(X, V) И ОТСУТСТВУЕТ Свойство (Y, V)

ТО несовпадение(X,Y,Свойство) И несовпадение (Y,X,Свойство),

а правила для агрегации как:

*ЕСЛИ $0.2 * \text{number}[\text{несовпадение}(X, Y, _)] < \text{number}[\text{совпадение}(X, Y, _)] + \text{number}[\text{несовпадение}(X, Y, _)]$*

ТО подобен (X,Y) И подобен(Y,X)

При этом оказалось, что в семиотическую модель необходимо добавить специфическую операцию агрегирования *number*, пригодную для вставки в условие правила, и некоторые коэффициенты правила, минимального сходства и максимального различия (в примере он равен 0,2). Это, конечно, усложняет представление модели и потенциально способно снизить её быстроедействие, однако такое

решение оказалось работоспособным. Общая архитектура полученной системы показана на Рис. 3.

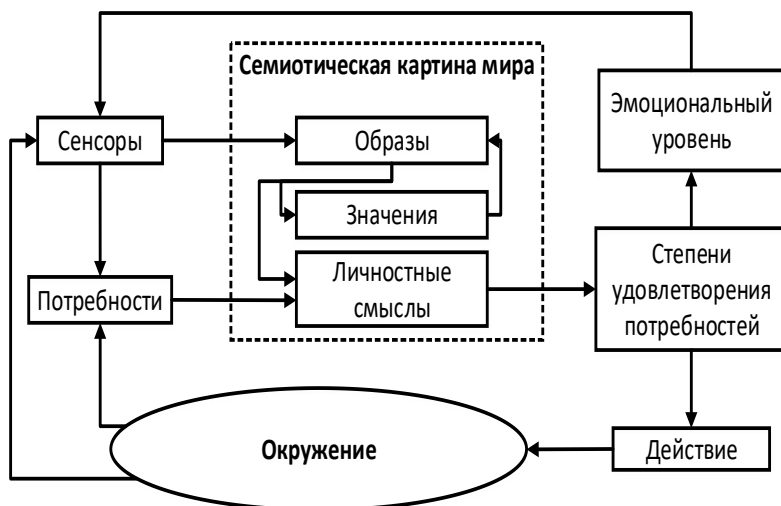


Рис. 3. Общая структура эмоционально-потребностной архитектуры с принятием решений на основе семиотического вывода

5. РЕЗУЛЬТАТЫ И ВЫВОДЫ

Описанная архитектура была реализована как встраиваемый программный модуль к описанной в [6] системе эмоционально-потребностного моделирования. Если в упомянутой работе явления симпатии/антипатии реализовывались как феноменологические механизмы, то теперь была рассмотрена реализация этих механизмов.

Тестирование показало, что полученная система работоспособна: агенты корректно выбирают действия, меняют своё состояние и удовлетворяют потребности. Возможна оказалась организация обучения агентов с подкреплением с использованием уровня эмоций как показателя успешности его функционирования.

Созданную архитектуру планируется использовать для реализации робота-помощника для человека, способного действовать в паре и интерпретировать как команды с пульта операторского управления (джойстика или аналогичного устройства), так и голосовые

указания из ограниченного подмножества естественного языка. Кроме того, благодаря дополнительной обработке сенсорной информации робот будет получать указания и без явного использования команд: в некоторых ситуациях он должен автоматически реагировать на действия оператора, например следовать за ним по маршруту при совместном патрулировании или сходить с маршрута при выявлении нарушителя. Оценка ситуации роботом и принятие решения формализуема на основе эмоционально-потребностной архитектуры с использованием моральных норм. Благодаря встроенной эмпатии агент будет способен распознавать «язык поз», то есть классифицировать фигуры людей с отнесением каждой из них к классу «нейтральная», «агрессивная», «подчиняющаяся» и реагированием на них в соответствии с ситуацией.

Литература

1. Осипов Г.С., Панов А.И., Чудова Н.В. Управление поведением как функция сознания. I. Картина мира и целеполагание // Известия Российской Академии Наук. Теория и системы управления. 2014. № 4. С. 49–62.
2. Milnes B.G. и др. A Specification of the Soar Cognitive Architecture in Z // Tech. Rep. С. 1992. № August 1992.
3. Simonov V.P. Thwarted action and need – informational theories of emotions // Int. J. Comp. Psychol. 1991. Т. 5. № 2. С. 103–107.
4. Ильин Е.П. Эмоции и чувства. СПб: Изд-во: Питер, 2001. 752 с.
5. Апресян Р.Г., Артемьева О.В., Прокофьев А.В. Феномен моральной императивности. Критические очерки. М.: ИФ РАН, 2018. 196 с.
6. Карпов В.Э., Сорокоумов П.С. К вопросу о моральных аспектах адаптивного поведения искусственных агентов // Искусственные общества. 2021. Т. 16. № 2.

ИССЛЕДОВАНИЕ АЛГОРИТМОВ ПЛАНИРОВАНИЯ МАРШРУТА МОБИЛЬНОГО РОБОТА С УЧЕТОМ ОГРАНИЧЕНИЙ НА ОРИЕНТАЦИЮ

*Кушнарев В.Н., студент
МГТУ им. Н.Э. Баумана
e-mail: grand_yarl@mail.ru*

1. ВВЕДЕНИЕ

При решении задачи автоматического планирования маршрута робота необходимо учитывать все кинетические и динамические ограничения, накладываемые на мобильного робота при движении в заданной среде. Для задачи, связанной с учетом допустимого положения робота в среде, разработано множество методов и подходов, позволяющих учитывать совершенно различные виды препятствий. Не менее важным является решение задач, связанных с учетом ограничений на ориентацию мобильного робота в пространстве, т.к. в некоторых ситуациях от ориентации зависит устойчивость робота и его динамические характеристики.

В робототехнике методы планирования маршрута условно делятся на 2 основных вида: глобальные и локальные. Глобальные методы планирования используют заранее заданную карту препятствий для построения глобального плана маршрута, обходящего известные препятствия. Данный алгоритм может рассчитывать длинные маршруты через сложные конфигурации, что требует значительных затрат по времени и памяти. Локальные методы планирования используют глобальный маршрут, разработанный заранее, и данные с датчиков для корректировки маршрута во время движения, учитывая препятствия лишь в некоторой ограниченной области [1]. Локальный маршрут строится во время движения робота, поэтому вычисления должны быть быстрыми и требовать минимального количества памяти.

Однако часто применяемые алгоритмы планирования не учитывают другие ограничения на допустимые или желательные положения робота, в частности, ориентацию. Ориентация может быть важна как для обеспечения устойчивости робота при движении по наклонным поверхностям, так и для оптимизации выполнения других задач. К примеру, при следовании за целью следует держать её в поле

зрения камеры, установленной на роботе. При определении ориентации робот использует различные визуальные ориентиры. С помощью этих ориентиров можно задать хорошие и нежелательные положения для робота.

Наиболее часто используемым является метод планирования маршрута на основе клеточного разбиения. Для поиска пути обычно используются алгоритмы эвристического поиска, одним из которых является алгоритм A^* [2]. В рамках поставленной задачи возможен учет угловых ограничений на ориентацию за счет определения дополнительной оценки допустимости перехода из одной клетки поля в другую.

Значительное распространение получили методы построения маршрутной карты на основе графов видимости [3] и диаграмм Вороного [4]. В отличие от клеточного разбиения эти методы отличаются меньшим размером составляемого графа, что увеличивает быстродействие алгоритма поиска. При составлении графа необходимо проверять условие возможности перехода из одного узла в другой в областях с ограничениями на ориентацию, что может привести к получению несвязного графа и отсутствию решения.

Недостатком точных алгоритмов нахождения маршрута является быстрый рост вычислительной сложности задачи при усложнении конфигурации пространства. Такого недостатка лишены приближенные методы построения маршрутной карты на основе выборки, яркими представителями которых является метод вероятностных маршрутных сетей [5] и алгоритм быстро растущих деревьев [6]. Эти алгоритмы легко адаптировать под сформулированную задачу, необходимо лишь добавить дополнительное условие для бесконфликтного перехода из одной вершины в другую при составлении графа.

Отдельным классом выделяются методы построения маршрута на основе искусственных потенциальных полей. Основной идеей этих алгоритмов является проведение физической аналогии движения робота с движением заряженной частицы в электростатическом поле. Конечная точка генерирует притягивающую силу, препятствия генерируют отталкивающие силы. В результате этих сил формируется маршрут робота [7].

В данной работе исследуются методы построения маршрута с учетом угловых ограничений на основе дискретизации угла в качестве глобального метода планирования и на основе дополнительного потенциального поля в качестве локального, предлагаются соответствующие модификации алгоритмов. Для полного решения

поставленной проблемы возможно совместное использование предложенных алгоритмов, которое позволит избавиться от некоторых недостатков, присущих каждому методу по отдельности.

2. МЕТОД НА ОСНОВЕ ДИСКРЕТИЗАЦИИ УГЛА

В качестве структуры данных, описывающих оценки допустимости ориентации в различных точках пространства, используется структура, представляющая собой двумерную карту, содержащую массивы значений оценки допустимости угла для каждой клетки, соответствующего его номеру в массиве с учетом дискретизации полного оборота.

Таким образом, для каждой клетки, где вводится ограничение на расположение, хранится массив из n чисел (рис. 1), которые означают цену перехода из одной клетки в соседнюю (например, от 0 до 1, где 0 – допустимый переход, 1 – переход невозможен).

| | | |
|------------|-------------|------------|
| 0.1 | 0 | 0.1 |
| 1 | i, j | 1 |
| 1 | 1 | 1 |

Рис. 1а – Структура массива для дискретизации угла с 8 соседними клетками

| | | | | |
|------------|----------|-------------|----------|------------|
| 0.1 | 0 | 0 | 0 | 0.1 |
| 1 | | | | 1 |
| 1 | | i, j | | 1 |
| 1 | | | | 1 |
| 1 | 1 | 1 | 1 | 1 |

Рис. 1б – Структура массива для дискретизации угла с 16 соседними клетками

После определения цены перехода данная оценка включается в алгоритм A^* . В таком случае стоимость перехода в клетку высчитывается следующим образом:

$$g(x_{i+1}) = dist(start, x_i) + length(x_i, x_{i+1}) + k \cdot restrict(x_i, x_{i+1}), \quad (1)$$

где $dist(start, x_i)$ – минимальное расстояние от начала до текущей точки, $length(x_i, x_{i+1})$ – цена перехода из текущей точки в соседнюю,

$restrict(x_i, x_{i+1})$ – ограничение на ориентацию (описанное выше), k – коэффициент влияния оценки ориентации.

Для соблюдения безопасности маршрута необходимо ввести условие невозможного перехода, т.е. если $restrict(x_i, x_{i+1}) = 1$, то переход считается невозможным и робот не может следовать в том направлении.

Графическое отражение расчета полученного алгоритма приведено на рисунке 2. Красные зоны обозначают препятствия с их зонами влияния, синие стрелки показывают приоритетное направление для зоны с ограничениями на ориентацию. Результаты представляют собой маршруты: ($A^* 8$) рассчитывался без учета ограничений на ориентацию, ($A^* \text{ angle } 8$) маршрут был рассчитан со всеми условиями. На рисунке видно, что с введением дополнительной оценки маршрут полностью перестраивается для выполнения ограничений на ориентацию в заданной области.

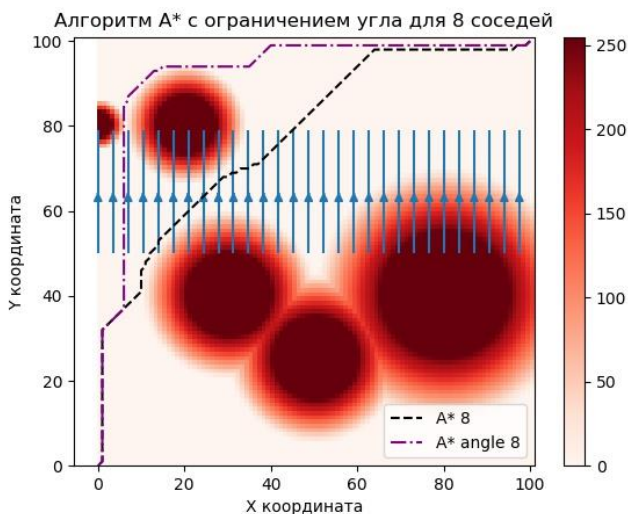


Рис. 2 – Результат работы алгоритма A^* с учетом угла для 8 соседей

Недостатком полученного результата являются большие интервалы дискретизации угла. Т.к. у клетки всего 8 соседей, то мы можем учесть лишь углы, кратные 45° . Если же допустимый угол будет меньше 45° , то робот сможет ехать только в прямом направлении. Для решения этой проблемы составим матрицу

смежности не для смежных клеток, а для более отдаленных на одинаковое расстояние (рис. 1б).

Графическое отражение расчета по алгоритму с большим количеством соседей представлен на рисунке 3, где маршруты (A* 16) и (A* angle 16) – это новый алгоритм A* с увеличенным количеством соседей без учета угла и с его учетом соответственно. (A* angle 8) показывает результат расчета той же сцены, но с использованием алгоритма, учитывающего переход только для 8 соседей.

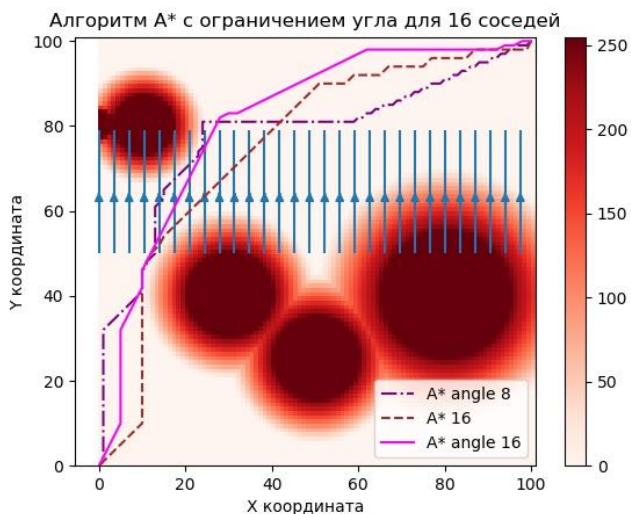


Рис. 3 – Результат работы алгоритма A* с учетом угла для 16 соседей

Для сравнительного анализа производительности алгоритмов были проведены 100 экспериментов по нахождению маршрута в случайных конфигурациях пространства. Полученные решения обеспечивали соблюдение различных ограничений ориентации на случайных областях пространства. Среднее количество операций для 8 соседей составило 5980,91; для 16 соседей – 4532,74. Таким образом эксперимент показал, что в большинстве случаев алгоритм, учитывающий 16 соседних клеток, работает быстрее, чем алгоритм, учитывающий 8. Особенно явно это проявляется в случаях простой конфигурации пространства, когда количество препятствий и сложность маршрута невелика.

Преимуществами данного алгоритма являются простота реализации и интеграции в существующие системы планирования маршрута. Данная модификация алгоритма A^* подходит для глобального планирования маршрута с учетом ограничений на ориентацию.

Недостатками является высокая вычислительная сложность задачи и большие затраты памяти, необходимой для хранения ограничений по ориентации, при усложнении сцены вокруг робота.

3. МЕТОД НА ОСНОВЕ ДОПОЛНИТЕЛЬНОГО ПОТЕНЦИАЛЬНОГО ПОЛЯ

Для локального планирования маршрута метод дискретизации угла не подойдет вследствие низкого быстродействия и высокого потребления памяти. В качестве локального метода планирования можно использовать метод на основе искусственных потенциальных полей

Навигационная функция, как правило, задается суммой притягивающих и отталкивающих потенциалов [8, с. 2]:

$$U(x, y) = U_{\text{прит}}(x, y) + U_{\text{отт}}(x, y); \quad (2)$$

Притягивающий потенциал задается следующими уравнениями [3, с. 4]:

$$U_{\text{прит}}(x, y) = \frac{1}{2} \zeta \left(d \left((x, y), (x_{\text{цель}}, y_{\text{цель}}) \right) \right)^2 \text{ при } d < D';$$
$$U_{\text{прит}}(x, y) = D' \zeta d \left((x, y), (x_{\text{цель}}, y_{\text{цель}}) \right) - \frac{1}{2} \zeta D'^2 \text{ при } d \geq D', \quad (3)$$

где d – функция, определяющая расстояние между точками, ζ – масштабирующий коэффициент, D' является константным параметром, который определяет пороговое значение расстояния до цели, при достижении которого скорость движения объекта начинает снижаться. Это предотвращает дрожание робота около цели.

Отталкивающий потенциал считается как сумма отталкивающих потенциалов от каждого препятствия. Для препятствия отталкивающий потенциал задается следующим образом [8, с. 3]:

$$U_{\text{отт}_i}(x, y) = \frac{1}{2} n \left(\frac{1}{d_{\min((x, y), o)}} - \frac{1}{D''} \right)^2 \text{ при } d_{\min}((x, y), o) < D''; \quad (4)$$

$$U_{\text{отт}_i}(x, y) = 0 \text{ при } d_{\min} \geq D'',$$

где минимальное расстояние от заданной точки до препятствия задается функцией $d_{\min}((x, y), o)$, константный параметр D'' определяет предельное расстояние от заданной точки до удаленных препятствий сцены, вклад от которых учитывается при подсчете потенциала.

Таким образом получаем статическое потенциальное поле для сцены, в которой происходит расчет маршрута (рис. 4).

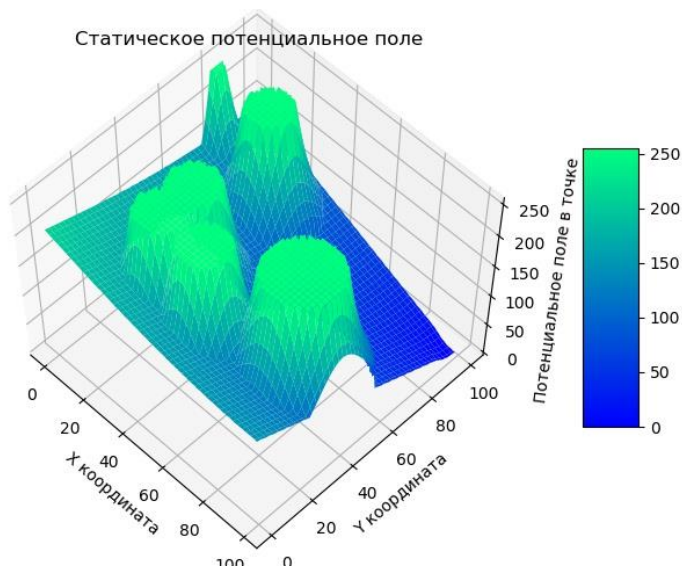


Рис. 4 – Пример построения статического поля

Помимо отталкивающего и притягивающего потенциальных полей для учета угла вводится новое динамическое поле, сцепленное с роботом в области с ограничением ориентации. Это поле выступает в качестве «хомута», направляя робота в приоритетном направлении. Для данной симуляции в качестве такого поля выступает уравнение конуса (рис. 5), которое описывается:

$$z(x, y) = -k * tg \alpha * \sqrt{\left(\frac{y-y_0+1}{tg \alpha}\right)^2 - (x-x_0)^2},$$

$$y - y_0 > \pm tg \alpha * (x - x_0) - 1;$$

$$z(x, y) = c,$$
(5)

$y - y_0 < tg \alpha * (x - x_0) - 1 \vee y - y_0 < -tg \alpha * (x - x_0) - 1;$
 где x_0 и y_0 – текущее положение робота, α – допустимый угол подъема робота, c – значение потенциального поля для препятствий.

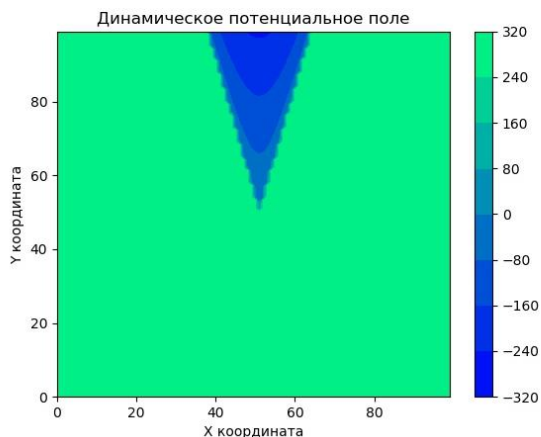


Рис. 5 – Конфигурация динамического потенциального поля

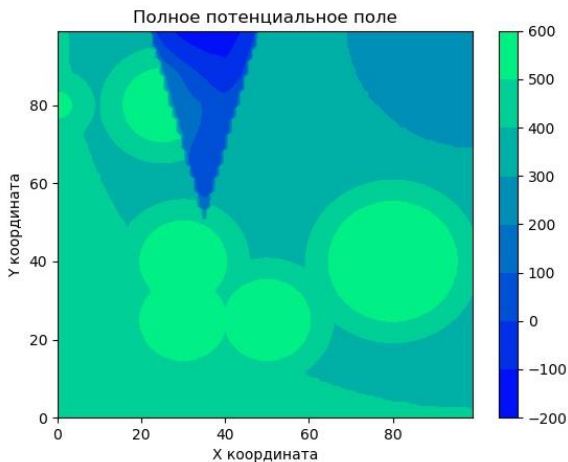


Рис. 6 – Сумма статического и динамического потенциальных полей для точки

Для произвольной точки с наличием ограничений на ориентацию вычисляется сумма статического и динамического потенциальных полей. Из рисунка 6 видно, что происходит одновременный учет как угла, так и препятствий. Резкое возрастание поля через угол α обеспечивает его соблюдение. Подбор значимости оптимальности маршрута и соблюдения угла осуществляется с помощью параметра k . При увеличении коэффициента влияние динамического потенциального поля возрастает, что увеличивает безопасность маршрута, но при этом может его удлинить.

Результат расчет маршрута методом градиентного спуска изображен на рисунке 7.

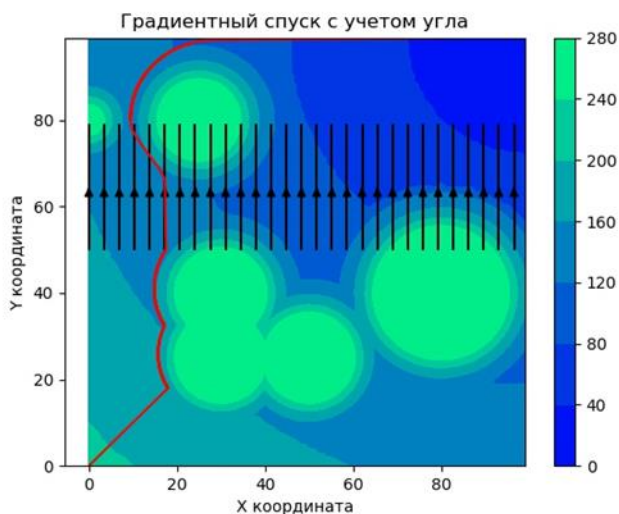


Рис. 7 – Результат градиентного спуска для поля с препятствиями при $k = 255 / 10$

При увеличении степени влияния динамического поля получим, что робот может сильнее приближаться к препятствиям ради соблюдения угла (рис. 8). Таким образом, регулировка коэффициента k дает возможность подстраивать решение под различные задачи.

Для исследования производительности метода были также проведены 100 экспериментов для различных конфигураций пространства. При этом также изменялось динамическое потенциальное поле – изменялся не только допустимый угол, но само поле поворачивалось на различный угол. При достаточно большом

значении k все ограничения ориентации соблюдались. Для шага 0.1 на поле 100x100 среднее количество операций составило 456.59, что свидетельствует о высокой скорости вычисления маршрута.

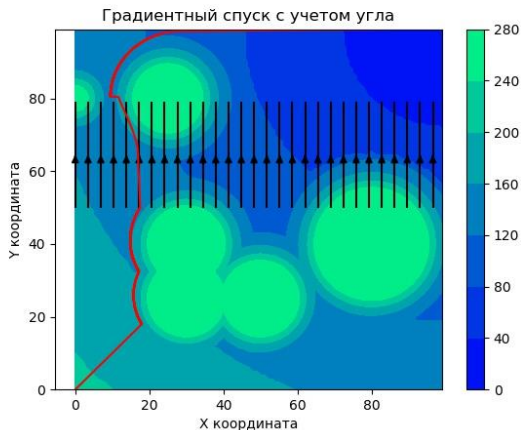


Рис. 8 – Результат градиентного спуска для поля с препятствиями при $k = 255 / 5$

Преимуществом метода является малая вычислительная сложность и малые затраты памяти, т.к. для области можно хранить только необходимый угол.

Недостатками метода является возможность «застревания» робота в точках локального минимума потенциала, колебания траектории вблизи препятствия [9] и проблема соблюдения угла вблизи препятствий при малых k .

4. ЗАКЛЮЧЕНИЕ

В представленной работе была рассмотрена проблема планирования движения мобильного робота с учетом ограничений на допустимый угол перемещения. Были рассмотрены несколько методов и предложены модификации алгоритмов, позволяющие учитывать ограничение на угол движения.

Метод, основанный на дискретизации угла, показал свою работоспособность в качестве метода глобального планирования, результаты его работы являются удовлетворительными. Введение матрицы смежности для большего числа равноудаленных от робота клеток приводит к повышению точности и быстродействию алгоритма, но увеличивает требуемую для хранения ограничений память.

Метод, основанный на применении динамического потенциального поля может использоваться в качестве метода локального планирования, однако имеет ряд недостатков. Необходимо разработать алгоритм для подбора оптимального параметра динамического потенциального поля для более корректного объезда препятствий с соблюдением допустимого угла. Для решения проблем колебания траектории около препятствия необходимо применение аппроксимации траектории, что позволит сформировать корректные управляющие воздействия.

Совмещение предложенных алгоритмов может избавить метод динамического потенциального поля от недостатков некорректного объезда препятствий и шанса попадания в локальный минимум и обеспечить соблюдение маршрутом заданных ограничений на всех этапах работы мобильного робота.

Литература

1. Cheng C. и др. Path planning and obstacle avoidance for AUV: A review // *Ocean Eng.* 2021. Vol. 235.
2. Peter E.Hart, Nils J. Nilsson, Bertram Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. 1968
3. De Berg M., Cheong O., Van Kreveld M., Overmars M. *Computational Geometry Algorithms and Applications*. 3rd Edition. Springer-Verlag, 2008. 386 p.
4. Franz Aurenhammer. Voronoi Diagrams — A Survey of a Fundamental Geometric Data Structure *ACM Computing Surveys*. Vol. 23, No. 3. September 1991.
5. Kavradi L.E., Svestka P., Latombe J.C., Overmars M.H. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robot. Autom.* Vol. 12, № 4. 1996. Pp. 566–580.
6. LaValle S.M., Kuffner J.J. Rapidly-exploring random trees: Progress and prospects. 2000 Workshop on the Algorithmic Foundations of Robotics. 2000. Pp. 293–308.
7. Лавренов Р.О. Математическое и программное обеспечение решения задачи многокритериального поиска пути мобильного объекта. Дис. ... канд. тех. наук: 05.13.18. Казань, 2020.
8. Филимонов А.Б., Филимонов Н.Б., Барашков А.А. Вопросы построения потенциальных полей в задачах локальной навигации мобильных роботов // *Автометрия*. 2019. №4. С. 65-70.
9. Koren Y., Borenstein J. Potential field methods and their inherent limitations for mobile robot navigation // *Proc. - IEEE Int. Conf. Robot. Autom.* 1991. Vol. 2. P. 1398–1404.

ИЕРАРХИЧЕСКАЯ ПОСТАНОВКА ЗАДАЧИ ОБЪЕКТНО-ЦЕНТРИЧНОГО ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ*

*Панов А.И., к.ф.-м.н., доцент
ФИЦ ИУ РАН, МФТИ, AIRI
panov.ai@mipt.ru*

1. ВВЕДЕНИЕ

В настоящее время обучение с подкреплением без использования модели среды показывает впечатляющие результаты для многих задач управления поведением когнитивных агентов и их групп [1-5]. Однако, большинство из используемых в этой области методов требуют чрезвычайно большого количества эпизодов взаимодействия агента со средой. Эта так называемая проблема эффективности выборов является одним из ключевых препятствий на пути развития новых методов и их внедрения в практику. Среди подходов к решению этой проблемы необходимо отметить имитационное обучение (с использованием заранее подготовленных демонстраций) [6,7] и обучение на основе модели [8,9]. Последний подход перспективен также и по той причине, что допускает возможность использования эффективных методов планирования и поиска по графу марковского процесса принятия решений [10,11].

Построение модели с использованием аппроксиматоров является вычислительно затратной процедурой и эффективные методы ее решения, не использующие заранее подготовленные и набранные агентом данные, только разрабатываются. Здесь перспективным является подход по использованию объектного представления состояний среды [12,13], который позволяет декомпозировать модель среды на частные объектные модели. Настоящая работа посвящена именно объектной формулировке задачи обучения с подкреплением на основе модели. Используются недавние результаты по свойствам эквивалентности моделей по функции полезности [14] для уточнения оптимизационной постановки задачи обучения. Основным

* Работа выполнена при финансовой поддержке Российского научного фонда, проект № 20-71-10116.

результатом работы является сведение оптимизационной задачи к иерархическому случаю и использование подхода умений [15], для которого существуют эффективные реализации процесса обучения [16,17]. В заключение работы введенные понятия и модели иллюстрируются в игровой среде Crafter [18].

2. ПОСТАНОВКА ЗАДАЧИ

Определим марковский процесс принятия решений как пятерку $\langle S, A, T, R, \gamma \rangle$, где S – это дискретное множество состояний среды, A – множество доступных действий (в т.ч. непрерывных), $T: S \times A \rightarrow \Pi(S)$ – функция переходов состояний среды (здесь и далее мы будем записывать множество вероятностных распределений на некотором множестве X как $\Pi(X)$), $R: S \times A \rightarrow \mathbb{R}$ – функция вознаграждений, γ – дисконтирующий множитель. Агенту, действующему в среде не известны функции T и R и в случае обучения с использованием модели он поддерживает аппроксимации этих функций \tilde{T} и \tilde{R} , которые в совокупности и образуют модель среды. Будем называть стратегией агента отображение $\pi: S \rightarrow \Pi(A)$ из состояния в некоторое распределение на множестве действий, из которого затем выбирается конкретное действие для применения в среде. Наконец, функцией полезности $V^\pi(s)$ будем называть ожидаемое суммарное вознаграждение (отдачу), которое получит агент, применяя некоторую стратегию π до завершения эпизода взаимодействия со средой: $V^\pi(s) = E_\pi[\sum_{i=1}^{\infty} \gamma^i R(s_{t+i}, a_{t+i}) | s_t = s]$. Целью агента является нахождение такой стратегии, которая будет максимизировать отдачу в среде.

Формально задачу обучения с подкреплением на основе модели можно сформулировать как следующую оптимизационную задачу:

$$\begin{cases} \operatorname{argmax}_\pi E[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)], \\ \operatorname{argmin}_R \mathcal{L}_R(R, \tilde{R}), \\ \operatorname{argmin}_T \mathcal{L}_T(T, \tilde{T}), \end{cases} \quad (1)$$

где в качестве функций потерь для определения компонентов модели \mathcal{L}_R и \mathcal{L}_T обычно выступают среднеквадратичная ошибка и оценка максимума правдоподобия соответственно:

$$\begin{aligned} \mathcal{L}_R(R, \tilde{R}) &= E_{(s,a) \sim \mathcal{D}} \left[\left(R(s, a) - \tilde{R}(s, a) \right)^2 \right], \\ \mathcal{L}_T(T, \tilde{T}) &= E_{(s,a) \sim \mathcal{D}} \left[D_{KL}(T(\cdot | s, a) || \tilde{T}(\cdot | s, a)) \right], \end{aligned}$$

где \mathcal{D} – память прецедентов или траекторий, по которым проводится оценка соответствующих функций потерь.

Оценка текущей стратегии агента, т.е. определение функции V^π полезности для всех состояний среды, осуществляется на основе нахождения решения следующего уравнения: $B_\pi[V^\pi] = V^\pi$, где

$$B_\pi[V](s) = E_{a \sim \pi, s' \sim T}[R(s, a) + \gamma V(s')] \quad (2)$$

называется оператором Беллмана.

Введем объектное представление состояния среды. Пусть имеется конечное множество классов объектов $C = \{c_1, c_2, \dots, c_k\}$ и множество признаков $F^c = \{f_1^c, f_2^c, \dots, f_{n_c}^c\}$, с помощью которых описывается каждый объект класса c . Объект $e \in E$ класса $c(e) \in C$ характеризуется вектором $e = \{d_1^c, d_2^c, \dots, d_{n_c}^c\}$, где $d_i^c \in D(f_i^c)$ – значение признака f_i^c из его области определения. Пусть имеется некоторая функция выделения множества объектов из первоначальной информации о состоянии $\Phi: S \rightarrow 2^E$. Данную функцию может реализовывать некоторая дополнительная сенсорная модель, обучаемая в ходе взаимодействия агента со средой, либо сформированная (предобученная) заранее. Состояние, таким образом, будет задаваться набором объектов: $s = \bigcup_{i=1}^n e_i$. В том случае, когда всю информацию об объектах удастся закодировать в признаках, в том числе информацию об их взаимодействии с другими объектами, модель среды может быть декомпозирована пообъектно: $\tilde{T} = \{\tilde{T}_{c_i} | c_i \in C\}$, $\tilde{R} = \{\tilde{R}_{c_i} | c_i \in C\}$. Такая декомпозиция естественным образом продолжается на множество действий и функцию полезности, т.е. в том случае, когда выполняется условие пообъектного разделения условий и эффектов каждого действия, множество действия разбивается на подмножества для каждого класса объектов: $A = \{A_{c_i} | c_i \in C\}$. Классовое действие $a_{c_i} \in A_{c_i}$ меняет свойства только одного объекта $e \in c_i$, $e \in s$. Аналогично можно проследить вознаграждения, получаемые для каждого класса объектов отдельно, и таким образом ввести объектную параметризацию функции полезности: $V^\pi(e)$, $e \in s$.

3. ПРИНЦИП ЭКВИВАЛЕНТНОСТИ МОДЕЛИ ПО ПОЛЕЗНОСТИ

Для формулировки полезных свойств объектного представления модели воспользуемся принципом эквивалентности модели по полезности [14]. Для начала отметим, что любая модель $M = \langle T, R \rangle$

индуцирует оператор Беллмана в соответствии с выражением (2), где функциями переходов и вознаграждения служат элементы модели M .

Определение 1. Пусть \mathcal{P} – множество стратегий, а \mathcal{V} – множество некоторых функций (полезности). Будем говорить, что две модели $M = \langle T, R \rangle$ и $\tilde{M} = \langle \tilde{T}, \tilde{R} \rangle$ эквивалентны по полезности относительно \mathcal{P} и \mathcal{V} тогда и только тогда, когда

$$\mathcal{B}_\pi[V] = \tilde{\mathcal{B}}_\pi[V], \forall \pi \in \mathcal{P}, V \in \mathcal{V},$$

где \mathcal{B}_π и $\tilde{\mathcal{B}}_\pi$ – операторы Беллмана, индуцированные моделью M и \tilde{M} соответственно.

Данное определение приводит к заданию функциональных классов моделей, в которых элементы являются неразличимыми с точки зрения структуры функции полезности, задаваемой уравнением Беллмана.

Определение 2. Пусть \mathcal{P} – множество стратегий, \mathcal{V} – множество некоторых функций (полезности), а \mathcal{M} – пространство моделей. При заданной некоторой модели m будем называть $\mathcal{M}_m(\mathcal{P}, \mathcal{V})$ пространством эквивалентных по полезности моделей, каждая из которых эквивалентна по полезности модели m относительно \mathcal{P} и \mathcal{V} .

В обучении с подкреплением на основе модели целью агента является в том числе и поиск оптимальной модели $m^* = \langle T, R \rangle$ с истинными функциями переходов и вознаграждения. Если мы предполагаем использованием модели именно для более быстрой оценки стратегии, то агенту достаточно найти любую модель, которая эквивалентна по полезности оптимальной модели m^* , т. е. найти $m \in \mathcal{M}_{m^*}(\mathcal{P}, \mathcal{V}) \equiv \mathcal{M}(\mathcal{P}, \mathcal{V})$. Для этого оптимизационную задачу, сформулированную в (1) можно переписать, объединив минимизирующие функционалы по функциям вознаграждения и переходов в один функционал для уравнения Беллмана:

$$\operatorname{argmin}_{\tilde{m}} \sum_{\pi \in \mathcal{P}} \sum_{V \in \mathcal{V}} \|\mathcal{B}_\pi[V] - \tilde{\mathcal{B}}_\pi[V]\|.$$

Соответствующая функция потерь уже явно зависит от выбора стратегии и соответствующей функции полезности, что позволяет, во-первых, сократить пространство поиска, во-вторых, использовать свойство функциональной эквивалентности по полезности моделей. Основной трудностью в данном случае является определение пространств \mathcal{P} и \mathcal{V} по которым проходит сравнение двух операторов Беллмана. В работе [14] предлагается несколько вариантов их

определения, в том числе поточечной линейной оболочки для \mathcal{P} и параметризованного пространства аппроксиматоров для \mathcal{V} .

4. ОБЪЕКТНЫЕ МОДЕЛИ И ПРИНЦИП ЭКВИВАЛЕНТНОСТИ

Теперь воспользуемся объектной декомпозицией модели, чтобы переформулировать оптимизационную задачу и сравнить решения, получаемые при использовании объектного представления и без него. Т. к. ранее мы вводили объектную параметризацию всех функций, участвующих в определении оператора Беллмана, то его объектная версия будет выглядеть следующим образом:

$$V_{\pi}[V](e) = E_{a(e) \sim \pi, s' \sim \tau} [R(e, a) + \gamma V(e') | e' \in s'].$$

Особенностью данного оператора является то, что выполнение действия для агента становится двухэтапным: вначале агент должен определить объект $e \in S$, с которым он будет взаимодействовать, а затем уже выбрать действие из множества доступных действий A_c , определенных для класса данного объекта $c(e)$. Таким образом, мы получаем иерархический вариант обучения с подкреплением, для которого обычно используется формулировка абстрактных действий в виде умений [19]. Пусть $\kappa(c) = \langle I_c, \pi_c, \beta_c \rangle$ – длящееся во времени действие, или умение, для которого $I_c \subseteq S$ – инициирующее множество состояний, такое, что $\forall s \in S, e \in s, c(e) = c$, π_c – стратегия, реализующая данное умение, такая, что $\forall a \sim \pi_c, a \in A_c, c(e) = c$, $\beta_c: S \rightarrow \{0,1\}$ – это терминальная функция, завершающая данное умение. Каждое умение $\kappa(c)$ задает типичную стратегию действия агента с любым объектом данного класса c . Умение может быть и одношаговым, когда $\beta_c(s_{t+1}) = 1, \forall s_t \in I_c$.

Иерархическая формулировка объектного обучения с подкреплением с использованием умений приводит к следующей реализации двухуровневой стратегией. На верхнем уровне агент на каждом шаге определяет объект определенного класса, с которым он будет взаимодействовать – это стратегия на умениях $\pi_{\kappa}: S \rightarrow C$. Второй уровень представляет собой стратегию π_c , которая задает последовательность действий с объектом класса c , реализуемую умением $\kappa(c)$. В простейшем случае можно считать, что для каждого класса объектов существует только одно умение.

Для иерархического случая уравнение Беллмана для стратегии на множестве умений K практически не изменится за исключением учета длительности самого умения, равной τ :

$$\mathcal{B}_{\pi_{\kappa}}[V](s) = E_{\kappa(c) \sim \pi_{\kappa}, s' \sim T} [R(s, \kappa(c)) + \gamma^t V(s') | s \in I_c(\kappa)].$$

Определение 3. Будем называть частной моделью модель $m(c) = \langle T_c, R_c \rangle$, которая характеризует часть марковского процесса принятия решений, задающего функции переходов и вознаграждений для конкретного класса объектов c .

Оптимизационная задача для объектно-центричного обучения с подкреплением на основе модели будет включать в себя помимо стандартной максимизации ожидаемой отдачи $k + 1$ подзадач минимизации, отвечающих за поиск оператора Беллмана в рамках эквивалентных по полезности k частичных моделей и одной общей (для умений):

$$\left\{ \begin{array}{l} \operatorname{argmax}_{\pi_{\kappa}} E[\sum_{t=0}^{\infty} \gamma^t R(s_t, \kappa_t)], \\ \operatorname{argmin}_{\tilde{m}} \sum_{\pi_{\kappa} \in \mathcal{P}_{\kappa}} \sum_{V \in \mathcal{V}_{\kappa}} \|\mathcal{B}_{\pi_{\kappa}}[V] - \tilde{\mathcal{B}}_{\pi_{\kappa}}[V]\|, \\ \operatorname{argmin}_{\tilde{m}_{c_1}} \sum_{\pi_{c_1} \in \mathcal{P}_{c_1}} \sum_{V \in \mathcal{V}_{c_1}} \|\mathcal{B}_{\pi_{c_1}}[V] - \tilde{\mathcal{B}}_{\pi_{c_1}}[V]\|, \\ \dots \\ \operatorname{argmin}_{\tilde{m}_{c_k}} \sum_{\pi \in \mathcal{P}_{c_k}} \sum_{V \in \mathcal{V}_{c_k}} \|\mathcal{B}_{\pi_{c_k}}[V] - \tilde{\mathcal{B}}_{\pi_{c_k}}[V]\|. \end{array} \right. \quad (3)$$

Утверждение 1. $k + 1$ подзадачи минимизации ошибки в определении оператора Беллмана могут решаться параллельно с использованием одной и той же памяти прецедентов или траекторий \mathcal{D} , для каждого состояний которой применена функция выделения объектов Φ .

Замечание 1 (локальная организация [20]). Полученный результат справедлив только в том случае, когда существует локальность эффектов каждого действия. Это означает, что изменение свойств одних объектов не сказываются на изменении свойств других. Это достаточно частный, но важный случай сред, пример которых будет рассмотрен ниже.

В качестве практической реализации подсчета близости модели к оптимальной эффективно использовать эмпирическую версию расстояния между предсказаниями полезности для всех состояний, которые встречаются для некоторой сгенерированной выборки:

$$\mathcal{L}(m^*, \tilde{m}) = \sum_{\pi \in \mathcal{P}} \sum_{V \in \mathcal{V}} \sum_{s \sim \mathcal{D}} \left[\frac{\sum_{\mathcal{D}, s_i = s} R(s_i) + \gamma V(s'_i)}{N_{\mathcal{D}}(s_i)} - \tilde{\mathcal{B}}_{\pi}[V](s) \right],$$

где $N_D(s_i)$ – количество раз, сколько состояние s_i встретилось в выборке \mathcal{D} .

Итоговый алгоритм обучения когнитивного агента с использованием объектного представления будет выглядеть следующим образом:

1. Инициализация верхнеуровневой стратегии агента π , объектных умений π_{c_i} , общей модели \tilde{m} и частных моделей \tilde{m}_{c_i} .
2. Генерация дополнительного опыта агента \mathcal{D} , полученного в среде с использованием стратегий π и π_{c_i} .
3. Обновление общей и частных моделей с использованием оптимизационной задачи (3).
4. Решение задачи планирования (определения субоптимальных функций полезности для умений V^i и общей стратегии V^k) по общей и частными моделям.
5. Использование жадных по полезностям V^i и V^k стратегий в качестве новых высокоуровневой стратегии π и объектных умений π_{c_i} . Переход к шагу 2.

5. МОДЕЛЬНЫЙ ПРИМЕР

Рассмотрим известную в области исследования методов обучения с подкреплением на основе модели среду Crafter [18]. Это клеточная среда с богатым набором подзадач (см. рис. 1), каждая из которых представляет собой конструирование некоторого объекта в среде. Наблюдение агента o представляет собой RGB изображение размером 64×64 , часть из которого отведена под представление некоторой окрестности среды вокруг агента (вся информация сгруппирована в клетки размеров 9×7), а часть – под представление собранных агентом ресурсов («инвентарь») и его собственных свойств (насыщение, здоровье и т. д.). Агенту доступны действия по перемещению и взаимодействию с имеющимися объектами (конструирование предметов, уничтожение противников и т. д.). Данная среда хорошо подходит для объектного представления: функция Φ выделения объектов может работать поклеточно: каждой клетке сопоставляется один из 19 типов объектов, каждый из которых может характеризоваться плотным векторным представлением.

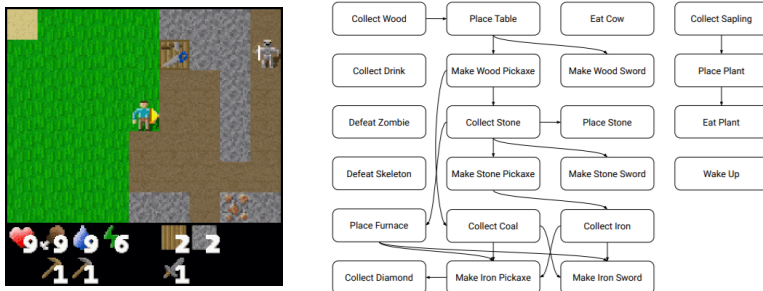


Рис. 1. Среда Crafter (слева) и список подзадач, которые выделяются в этой среде (справа).

Частными моделями для данной среды будут являться модели изменения признаков каждого объекта, который детектируется в соответствующих клетках (за исключением типов, отвечающих за подстилающую поверхность – траву, камни и т. п.). Умениями агента будут являться те действия, которые меняют характеристики данных объектов, соответствующие им оптимальные стратегии могут задаваться заранее (например, рубка дерева или уничтожение врагов).

6. ЗАКЛЮЧЕНИЕ

Перспективным подходом к повышению эффективности методов обучения с подкреплением является использование модели среды. В данной работе обсуждаются особенности объектного представления состояний. С использованием понятия эквивалентности моделей по полезности предложена иерархическая формулировка оптимизационной задачи объектно-центричного обучения с подкреплением на основе модели, сформулирован алгоритм обучения агента и используемые понятия продемонстрированы на среде Crafter.

Литература

1. Mnih V. et al. Human-level control through deep reinforcement learning // Nature. Nature Publishing Group, 2015. Vol. 518, № 7540. P. 529–533.
2. Rashid T. et al. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement Learning // 35th International Conference on Machine Learning, ICML 2018. 2018. Vol. 10. P. 6846–6859.
3. Staroverov A. et al. Real-Time Object Navigation with Deep Neural Networks and Hierarchical Reinforcement Learning // IEEE Access. 2020. Vol. 8. P. 195608–195621.

4. Skrynnik A. et al. Forgetful experience replay in hierarchical reinforcement learning from expert demonstrations // Knowledge-Based Systems. Elsevier B.V., 2021. Vol. 218. P. 106844.
5. Ugadiarov L., Skrynnik A., Panov A.I. Long-Term Exploration in Persistent MDPs // Advances in Soft Computing. MICAI 2021. Part I. Lecture Notes in Computer Science / ed. Baturshin I., Gelbukh A., Sidorov G. Springer, 2021. Vol. 13067. P. 108–120.
6. Reddy S., Dragan A.D., Levine S. SQIL: imitation learning via reinforcement learning with sparse rewards // ICLR 2020. 2020. P. 1–14.
7. Skrynnik A. et al. Hierarchical Deep Q-Network from imperfect demonstrations in Minecraft // Cognitive Systems Research. 2021. Vol. 65. P. 74–78.
8. Moerland T.M., Broekens J., Jonker C.M. Model-based Reinforcement Learning: A Survey. 2020. P. 421–429.
9. Zholus A., Panov A.I. Case-based Task Generalization in Model-based Reinforcement Learning // Artificial General Intelligence. AGI 2021. Lecture Notes in Computer Science / ed. Goertzel B., Iklé M., Potapov A. Springer International Publishing, 2022. Vol. 13154. P. 344–354.
10. Silver D., Gelly S. Monte-Carlo tree search and rapid action value estimation in computer Go // Artificial Intelligence. 2011. Vol. 175, № 11. P. 1856–1875.
11. Schrittwieser J. et al. Mastering Atari, Go, chess and shogi by planning with a learned model // Nature. Springer US, 2020. Vol. 588, № 7839. P. 604–609.
12. Diuk C., Cohen A., Littman M.L. An object-oriented representation for efficient reinforcement learning // Proceedings of the 25th International Conference on Machine Learning. 2008. P. 240–247.
13. Keramati R. et al. Strategic Object Oriented Reinforcement Learning. 2018. 1–10 p.
14. Grimm C. et al. The value equivalence principle for model-based reinforcement learning // Advances in Neural Information Processing Systems. Curran Associates, Inc., 2020. Vol. 33. P. 5541–5552.
15. Stolle M., Precup D. Learning Options in Reinforcement Learning // Abstraction, Reformulation, and Approximation. SARA 2002. Springer, 2002. P. 212–223.
16. Bacon P.-L., Harb J., Precup D. The Option-Critic Architecture // Proceedings of the AAAI Conference on Artificial Intelligence. 2017. Vol. 31.
17. Klissarov M., Precup D. Flexible Option Learning // NeurIPS. 2021.
18. Hafner D. Benchmarking the Spectrum of Agent Capabilities // Deep RL Workshop NeurIPS. 2021.
19. Sutton R.S., Precup D., Singh S. Between MDPs and Semi-MDPs: A Framework for Temporal Abstraction in Reinforcement Learning // Artificial Intelligence. 1999. Vol. 112. P. 181–211.
20. Стефанюк В.Л. Локальная организация интеллектуальных систем. М.: Физматлит, 2004. 328 с.

СЕКЦИЯ 8

**ИНТЕЛЛЕКТУАЛЬНОЕ
ПРОИЗВОДСТВО,
ИНТЕЛЛЕКТУАЛЬНОЕ
УПРАВЛЕНИЕ ЖИЗНЕННЫМ
ЦИКЛОМ**

УДК 004.8; 621.791

**КОНЦЕПЦИЯ СОЗДАНИЯ ИНФОРМАЦИОННЫХ
СИСТЕМ АВТОМАТИЧЕСКОГО КОНТРОЛЯ
РАБОТЫ СВАРЩИКОВ**

***Болотов С.В.**, к.т.н., доцент,
Белорусско-Российский университет,
e-mail: s.v.bolotov@mail.ru*

***Захарченков К.В.**, к.т.н.,
Белорусско-Российский университет,
e-mail: zaharchenkovkv@mail.ru*

***Фурманов В.А.**,
ИООО «GoDel Technologies»,
e-mail: fupkaa@gmail.com*

***Макаров Е.В.**,
ИООО «iTechArt»,
e-mail: jastapanda@gmail.com*

***Бобков Н.К.**,
Белорусско-Российский университет*

1. ВВЕДЕНИЕ

Информационные системы контроля сварочных работ широко используются на сварочных предприятиях, что является следствием перехода к Индустрии 4.0 [1,2]. Производители сварочного

оборудования ESAB, Kemppi, Fronius и др. предлагают собственные системы сбора данных и анализа сварочных процессов.

Информационные системы Fronius WeldCube, Kemppi WeldEye используются только на собственном сварочном оборудовании, отсутствует оценка работы сварщика [3-5]. Информационная система ESAB WELDQAS работает с любым сварочным оборудованием, позволяет оценивать качество сварки от 1 (отлично) до 7 (удовлетворительно), формировать протокол сварки. Недостатками информационной системы WELDQAS являются: исполнение в виде компьютерной программы, что не позволяет вести удалённо в реальном времени контроль сварочных процессов, отсутствие идентификации сварщиков и оборудования, закрытый алгоритм оценки качества сварки [6].

Развитием информационных систем контроля сварочных работ является использование интеллектуальных систем активного мониторинга и контроля и качества сварки [7,8], использование сварочных комплексов на основе человеко-кибер-физических систем (HCPS) [9]. Такие системы не нашли массового применения в производстве, однако интеллектуальные алгоритмы обработки данных, обнаружения дефектов сварки имеют большие перспективы [10,11].

Анализ существующих подходов к созданию информационных систем автоматического контроля сварочных процессов показал, что оценка работы сварщиков либо отсутствует, либо имеет субъективный характер; отсутствуют подходы, обеспечивающие комплексную автоматизацию оценки работы сварщиков, позволяющие исключить влияние человеческого фактора на результаты оценки.

Разработанная концепция предполагает переход от субъективной ручной оценки работы сварщиков руководителями сварочных работ и контролерами к комплексной автоматической оценке работы сварщиков на основе анализа параметров сварочных процессов, получаемых с регистраторов с последующей обработкой полученных данных для формирования оценки результатов сварки каждого сварочного прохода.

Основное достоинство предложенной концепции заключается в реализации постоянного автоматического контроля работы сварщиков с возможностью сохранения на сервере всех параметров режима, оценки качества работы и составления паспорта для каждого сварного соединения.

Предложенная концепция обеспечивает решение научной задачи повышения эффективности контроля сварочных работ с использованием регистраторов параметров сварочных процессов и авторского алгоритма формирования оценки работы сварщика на

основе усредненных значений параметров сварочных процессов, полученных с регистраторов.

2. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ КОНЦЕПЦИИ СОЗДАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ АВТОМАТИЧЕСКОГО КОНТРОЛЯ РАБОТЫ СВАРЩИКОВ

Концепция создания информационных систем автоматического контроля работы сварщиков основана на развитии методов объективной экспериментальной оценки процессов дуговой сварки [12,13].

Предложено анализ сварочных процессов осуществлять в двух частотных диапазонах [14,15]:

- 1) среднечастотном (до 10 кГц), который характеризует процессы переноса капель электродного металла в сварочную ванну;
- 2) низкочастотном (до 10 Гц), который характеризует процесс формирования сварочной ванны (сварного шва).

К методам изучения процессов переноса электродного металла относят: осциллографирование параметров режима сварки и высокоскоростную съёмку, которые реализуются на специальном информационно-измерительном комплексе [16]. Контролю подвергаются следующие параметры режима сварки: сварочный ток, напряжение на дуге, скорость сварки, скорость подачи проволоки, расход защитного газа и др. Стабильность процесса переноса электродного металла предложено оценивать по коэффициенту вариации энергии дуги на стадии формирования капли расплавленного металла [17]. Стабильность процесса обеспечивают определённым сочетанием параметров режима сварки для заданного материала свариваемых деталей, сварочных материалов.

Процесс формирования сварочной ванны оказывает влияние на геометрические размеры сварного шва: глубину проплавления, ширину шва, высоту усиления. Отклонения тока от заданного или среднего значения сказываются на глубине проплавления. Отклонения сварочного напряжения вызывают пропорциональные изменения ширины шва.

Обеспечить качество сварки возможно только при обеспечении стабильности процесса сварки и формировании сварного соединения требуемых по соображениям прочности геометрических размеров. Таким образом, информационная система должна использоваться как на этапе разработки инструкции на технологический процесс сварки для оптимизации параметров режима, так и в процессе сварки для контроля за их соблюдением.

В основе предложенной концепции лежат следующие принципы:

1) комплексный подход к автоматическому формированию оценки работы сварщика на основе параметров сварочных процессов;

2) комплексная обработка данных, получаемых с датчиков, на регистраторе в сочетании с последующей обработкой данных, поступивших с регистратора, на сервере в процессе формирования оценки работы сварщика;

3) визуальное представление параметров процесса работы сварщика в виде графических зависимостей, демонстрирующих изменение параметров процесса сварки в процессе проведения работ.

Повышение качества сварочных работ, в соответствии с предложенной концепцией, достигается за счет объективной оценки работы каждого сварщика с возможностью последующего детального анализа результатов работы сварщика в течение заданного периода, реализации возможности формирования рейтинга сварщиков и выявления сварщиков, у которых чаще других параметры сварочных процессов выходят за пределы допустимых значений, заданных в технологических инструкциях.

Отличительными особенностями концепции являются:

1) комплексный подход к созданию информационных систем контроля работы сварщиков, отличающийся:

– автоматической регистрацией параметров сварочных процессов с использованием специальных аппаратных средств (регистраторов), обеспечивающих получение параметров сварочных процессов с частотой от 10 кГц;

– комплексной обработкой информации о параметрах сварочных процессов с использованием аппаратного и программного обеспечения, что позволяет полностью автоматизировать оценку сварочных работ, исключив участие человека и повысив достоверность оценки качества сварочных работ;

2) использование алгоритмов оценки работы сварщиков, адаптированных к условиям производства сварочных работ.

Контроль работы сварщиков предполагает оценку качества сварочных работ на основе оценки параметров сварочных процессов. В соответствии с предложенной концепцией, контроль работы сварщиков обеспечивается путем автоматического выбора и поддержания оптимальных параметров режима сварки в зависимости

от технологии, требуемых характеристик сварных соединений, основных и сварочных материалов.

Автоматический контроль работы сварщиков с использованием предложного подхода предполагает реализацию аппаратной и программной части информационной системы (рис.1). Аппаратная часть системы обеспечивает получение параметров сварочных процессов со сварочных аппаратов и передачу на сервер. Программная часть обеспечивает интеллектуальный анализ данных [17]. Автоматический контроль работы сварщиков может быть реализован на этапе сварки корневого, заполняющего и облицовочного слоя в разных пространственных положениях.

Для решения задачи автоматического контроля работы сварщиков разработаны алгоритмы обработки информации, обеспечивающие возможность реализации основных принципов и особенностей предложенной концепции. Разработанные алгоритмы реализованы в автоматизированной системе регистрации сварочных процессов [18].



Рис. 1. Структурная схема концепции создания информационных систем контроля работы сварщиков

3. ФОРМАЛИЗАЦИЯ ПОСТАНОВКИ ЗАДАЧИ СОЗДАНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ АВТОМАТИЧЕСКОГО КОНТРОЛЯ РАБОТЫ СВАРЩИКОВ

Информационная система автоматического контроля сварочных работ $ISWk = \langle WP, Wemp, WT, Weq, WPS \rangle$ характеризуется множеством сварочных процессов WP , множеством сварщиков WE , множеством заданий на выполнение сварочных работ WT , множеством единиц оборудования Weq и множеством технологических инструкций WPS .

Сварочный процесс WP определяется как множество технологических операций WPO , которые необходимо выполнить для получения качественного сварного соединения.

Каждый сварщик $Wemp = \langle WPOa, WPw \rangle$ характеризуется множеством областей распространения квалификации, по которым он аттестован $WPOa$, и множеством параметров сварочных процессов, которые автоматически получают с регистраторов WPw .

Каждое задание на выполнение сварочных работ $WT = \langle Tech_{WT}, T_{WT} \rangle$ характеризуется используемой технологией $Tech_{WT}$ и сроком выполнения T_{WT} .

Каждая единица оборудования $Weq = \langle WPO_{Weq}, PMin_{Weq}, PMax_{Weq} \rangle$ характеризуются множеством способов сварки, для выполнения которых оно может использоваться WPO_{Weq} , множеством диапазонов (максимальных $PMax_{Weq}$ и минимальных $PMin_{Weq}$ допустимых значений) каждого параметра режима сварки.

Каждая технологическая инструкция $WPS = \langle WPO_{WPS}, PMin_{WPS}, PMax_{WPS}, PMid_{WPS} \rangle$ содержит множество технологических операций, которые могут выполняться в соответствии с данной инструкцией WPO_{WPS} , и множество максимальных $PMax_{WPS}$, минимальных $PMin_{WPS}$ и средних $PMid_{WPS}$ значений параметров сварочных процессов ($PMid_{WPS} = (PMax_{WPS} + PMin_{WPS})/2$).

Целевая функция задачи создания информационной системы автоматического контроля сварочных процессов представляет собой сумму оценок сварщиков за выполнение всех работ:

$$\sum_{i=1}^W WMw_i(WPO_i, WT_i, Weq_i, WPS_i) \rightarrow \max ,$$

где i – номер оценки сварщика за выполнение задания WT_i технологической операции WPO_i на оборудовании Weq_i в соответствии с технологической инструкцией WPS_i ; W – количество оценок, полученных сварщиками за выполнение всех работ.

4. АЛГОРИТМ АВТОМАТИЧЕСКОЙ ОЦЕНКИ РАБОТЫ СВАРЩИКОВ

Автоматическая оценка работы сварщиков получается на основании значений параметров сварочных процессов, передаваемых с блока датчиков на регистратор.

Параметры сварочных процессов WP_w можно разделить на следующие категории:

- критические параметры WP_{wk} , при выходе которых за пределы допустимых значений сварочный процесс не должен запускаться;

- пороговые параметры WP_{wt} , при выходе которых за пределы допустимых значений, сварщик и руководитель сварочных работ должны получать соответствующие уведомления;

- рабочие параметры сварочных процессов WP_{ww} , которые оказывают влияние только на оценку работы сварщика.

Оценка автоматической работы сварщиков выполняется по следующему алгоритму:

Этап 1. Оценка при запуске сварочного процесса. Если критические параметры сварочного процесса, при выходе которых за пределы допустимых значений процесс не должен выполняться, оказываются выше максимального $WP_{wk_j} > P_{Omax_{WPS_{ij}}}$, где j – номер параметра сварочного процесса) или ниже минимального значения ($WP_{wk_j} < P_{Omin_{WPS_{ij}}}$), указанного в технологической инструкции, сварочный процесс не запускается и на регистраторе появляется соответствующее сообщение об ошибке.

Этап 2. Оценка работы сварщика во время сварки.

Шаг 2.1. Если контролируемый параметр сварочного процесса выходит за пределы допустимых значений ($WP_{wt_j} < P_{Omin_{WPS_{ij}}}$ или $WP_{wt_j} > P_{Omax_{WPS_{ij}}}$), указанных в соответствующей технологической инструкции, сварщику выдается звуковой сигнал. Частота звукового сигнала зависит от контролируемого параметра, выходящего за пределы допустимых значений.

Шаг 2.2. Полученные с датчиков значения параметров сварочного процесса WP_w усредняются на регистраторе.

Шаг 2.3. Усредненные значения параметров $\overline{WP_w}$ передаются на сервер.

Этап 3. Формирование оценки работы сварщика выполняется следующим образом:

Шаг 3.1. Для каждого измеримого параметра сварочного процесса $WP_{w_{ij}}$, указанного в технологической инструкции,

определяется весовой коэффициент δWPw_j в диапазоне от 0 до 1. Сумма всех весовых коэффициентов должна быть равна 1. Весовые коэффициенты определяют на стадии разработки технологического процесса сварки и хранятся в базе данных.

Шаг 3.2. Оценка усредненного k -го значения каждого параметра сварочного процесса выставляется следующим образом:

$$WMw_{ijk} \begin{cases} \frac{(POMax_{WPSij} - \overline{WPw_{ijk}})}{(POMax_{WPSij} - POMid_{WPSij})}, & \text{если } POMid_{WPSij} < \overline{WPw_{ijk}} \\ \frac{(\overline{WPw_{ijk}} - POMin_{WPSij})}{(POMid_{WPSij} - POMin_{WPSij})}, & \text{если } POMid_{WPSij} \geq \overline{WPw_{ijk}} \\ 0, & \text{если } POMin_{WPSij} > \overline{WPw_{ijk}} \text{ или } POMax_{WPSij} < \overline{WPw_{ijk}} \end{cases},$$

Оценка сварщика за каждый шов рассчитывается по формуле:

$$WMw_i = \sum_{j=1}^P (\delta WPw_{ij} \cdot (\frac{\sum_{k=1}^{Pv} \overline{WPw_{ijk}}}{WP})),$$

где Pv – количество усредненных значений j -го параметра сварочного процесса, переданных с регистратора на сервер; P – количество параметров сварочных процессов, допустимые диапазоны значений которых указаны в технологической инструкции WPS_i .

Этап 4. Запись оценки сварщика для каждого прохода в базу данных для последующего анализа качества работы.

Предложенный алгоритм обеспечивает возможность полной автоматизации оценки работы сварщика, полностью исключая человеческий фактор и повышая объективность оценки работ.

5. ТЕХНОЛОГИЯ КОМПЛЕКСНОГО АВТОМАТИЧЕСКОГО КОНТРОЛЯ РАБОТЫ СВАРЩИКОВ

Технология автоматического контроля сварочных работ, разработанная в соответствии с предложенной концепцией, включает следующие основные этапы:

Этап 1. Ввод с помощью Web-приложения сменных заданий руководителями сварочных работ, включающих: наименование объекта, участок/цех, изделие, маркировку сварных соединений, прошедший аттестацию технологический процесс сварки с допусками на контролируемые параметры режима, основные и сварочные материалы.

Этап 2. Радиочастотная идентификация сварщика, сварочного оборудования на регистраторе сварочных процессов. Считывание сменных заданий с сервера, соответствующих области

распространения квалификации сварщика, с использованием GSM-канала.

Этап 3. Запуск на регистраторе процесса контроля сварочных работ.

Этап 4. Передача данных с блока датчиков (сварочный ток, напряжение на дуге, скорость подачи проволоки, расход защитного газа, температура в зоне сварного соединения) на регистратор и пульт сварщика.

Этап 5. Формирование паспорта каждого сварного соединения и отчета о работе сварщика.

Отличительными особенностями предложенной технологии является объективная оценка качества сварочных работ; возможность постоянного контроля работы сварщиков в реальном времени; документальное подтверждение качества сварки каждого соединения.

6. АПРОБАЦИЯ ПРЕДЛОЖЕННОГО ПОДХОДА ПРИ СОЗДАНИИ ИНФОРМАЦИОННОЙ СИСТЕМЫ АВТОМАТИЧЕСКОГО КОНТРОЛЯ РАБОТЫ СВАРЩИКОВ

Информационная система автоматического контроля работы сварщиков состоит из аппаратной части – регистратора сварочных процессов и программного обеспечения [18,19].

Регистратор подключается к сварочному источнику ручной дуговой, полуавтоматической или автоматической дуговой сварки любого производителя и конструктивно содержит три отдельных блока: основной блок регистрации и блок датчиков сварочного тока и напряжения на дуге, которые располагаются у сварочного аппарата и блок датчиков, который располагается в месте проведения сварочных работ и предназначен для переключения сварочных слоёв, контроля межслойной температуры и оповещения звуковым сигналом сварщика о приближении к границам рекомендуемых технологической инструкцией параметров [18].

Информационная система прошла апробацию в Центре сертификации и испытаний Белорусско-Российского университета при аттестации сварщиков по требованиям СТБ EN 287-1-2009 для выполнения сварочных работ на опасных производственных объектах. Проверка соответствия параметров режима, заданным инструкцией на технологический процесс сварки с записью на сервер, осуществлялась при проведении квалификационных испытаний контрольных сварных соединений при аттестации всех сварщиков на протяжении 2021 г. Контрольные образцы подвергались визуальному контролю и механическим испытаниям в контрольно-измерительной лаборатории для подтверждения соответствия требованиям технических

нормативно-правовых актов. В результате применения предложенной концепции количество сварщиков, прошедших первичную аттестацию, увеличилось с 55% в 2020 г. до 78% в 2021г., а повторную аттестацию – до 95%.

Система автоматизированного контроля работы сварщиков и качества сварочных работ установлена и прошла приёмо-сдаточные испытания в РУП «Производственное объединение «Белоруснефть» при сварке магистральных трубопроводов диаметром от 89 мм до 720 мм из стали 20. Белорусско-Российским университетом разработаны инструкции на 9 технологических процессов сварки, определены пороговые значения сварочного тока, напряжения на дуге для обеспечения стабильного процесса и требуемых размеров сварного шва при сварке корневого, заполняющего облицовочного проходов. Контроль сварочных работ осуществлялся с помощью трёх регистраторов сварочных процессов. На каждый сварной стык формировался паспорт с осциллограммами параметров режима сварки и оценкой работы сварщика по каждому сварочному проходу. На рис.2 представлен фрагмент паспорта сварного стыка при ручной дуговой сварке трубы 273x8 из стали 20 с использованием сварочных электродов LB-52U. Сварочный ток в соответствии с технологической инструкцией должен находиться в диапазоне от 50 до 80 А. В таблице представлены максимальные, минимальные и средние значения контролируемых параметров, а также сформированная системой средняя оценка за сварку прохода.



Рис.2. Фрагмент паспорта сварного соединения

В результате приёмо-сдаточных испытаний всех сварных стыков выходов за границы рекомендуемых технологической

инструкцией значений не зафиксировано, что отражено в паспортах сварных стыков и подтверждено отсутствием дефектов сварных соединений при 100 % рентгенографическом контроле.

Отсутствие нарушений параметров режима сварщиками объясняется информированием регистратором сварочных процессов о требуемых технологической инструкцией значениях при сварке каждого прохода и звуковым оповещении приближения к граничным значениям в процессе сварки, что позволяет своевременно корректировать технологический процесс.

В результате применения предложенной концепции, алгоритма автоматической оценки работы сварщиков и технологии комплексного автоматического контроля при производственных испытаниях, удалось повысить среднюю оценку сварщиков с 6,2 в начале испытаний до 7,2 в конце испытаний, за счёт объективной оценки каждого сварщика и проведенного руководителем сварочных работ анализа их работы с выработкой корректирующих воздействий.

7. ЗАКЛЮЧЕНИЕ

Предложена новая концепция создания информационных систем автоматического контроля работы сварщиков, основанная на комплексном применении новых аппаратных и программных средств автоматической регистрации сварочных процессов и последующей автоматической обработки информации.

Апробация предложенного подхода проведена при создании автоматизированной системы регистрации сварочных процессов для аттестации сварщиков в Центре сертификации и испытаний Белорусско-Российского университета. В результате применения предложенной концепции количество сварщиков, прошедших аттестацию увеличилось с 55% до 78%.

Применение предложенной концепции при реализации системы автоматизированного контроля работы сварщиков и качества сварочных работ в РУП «Производственное объединение «Белоруснефть» позволила повысить среднюю оценку сварщиков с 6,2 до 7,2 при обеспечении качества сварных соединений магистральных трубопроводов.

Литература

1. Lu Y. Industry 4.0: A Survey On Technologies, Applications And Open Research Issues. *J Ind Inf Integr* 2017; 6:1–10.
2. Reisgen U., Mann S., Middeldorf K., Sharma R. et al. Connected, Digitalized Welding Production – Industrie 4.0 In *Gas Metal Arc Welding. Weld World* 2019; 63:1121–31.
3. Fronius: официальный сайт // WELDCUBE [Электронный ресурс]. 2022. – URL: <https://www.fronius.com/en/welding-technology/innovative-solutions/weldcube> (дата обращения: 18.02.2022).
4. Posch G., Bruckner J., Ennsbrunner H. Industry 4.0 in welding. *Biuletyn Instytutu Spawalnictwa w Gliwicach* 2018;62.

5. КЕМППИ: официальный сайт. Программное обеспечение для управления сварочным производством [Электронный ресурс]. 2022. – URL: <https://www.kemppi.com/ru/offering/category/programmnoe-obespecenie-2/programmnoe-obespecenie-dla-upravlenia-svarocnym-proizvodstvom-2/> (дата обращения: 18.02.2022).
6. ESAB: официальный сайт компании // WELDQAS [Электронный ресурс]. 2022. – URL: <https://www.esab.ru/ru/products/esab-digital-solutions-eds/quality-assurance/weldqas.cfm> (дата обращения: 18.02.2022).
7. Wang B., Hu S., Sun L., Freiheit T. Intelligent Welding System Technologies: State-Of-The-Art Review And Perspectives // *Journal of Manufacturing Systems*. July 2020. Vol. 56, pp. 373-391.
8. Ji Zh., Peigen L., Zhou Y., Wang B. et al. Toward New-Generation Intelligent Manufacturing // *Engineering*. 2018. 4:11-20.
9. Ji Zh., Zhou Y., Wang B., Jiyuan Z. Human-Cyber-Physical Systems (HCPSs) In The Context Of New-Generation Intelligent Manufacturing // *Engineering*. 2019. 4:624-636.
10. Chen C., Lv N., Chen S. Data-Driven Welding Expert System Structure Based On Internet Of Things // *Transactions on intelligent welding manufacturing*. Springer. 2018. Pp. 45-60.
11. Sassi P., Tripicchio P., Avizzano C. A Smart Monitoring System For Automatic Welding Defect Detection // *IEEE Trans. Ind. Electron*. 2019. 66:9641-50.
12. Потаповский А.Г., Сараев Ю.Н., Чинахов Д.А. Сварка сталей в защитных газах плавящимся электродом. Техника и технология будущего: Монография. Юргинский технологический институт. – Томск: 2012. 208 с.
13. Милютин В.С. Испытания сварочных свойств оборудования для дуговой сварки. – Екатеринбург. 2019. 466 с.
14. Гладков Э.А., Юшин А.А., Перковский Р.А. и др. Оценка сварочных свойств оборудования с инверторными источниками питания по энергетическим характеристикам // *Сварка и диагностика*. 2011. №1. С.31-35.
15. Bolotov S.V. Investigation of the Criteria for Evaluating Electrode Metal Transfer in Short Circuit Gas-Shielded Arc Welding [Electronic resource] // *IOP Conference Series: Materials Science and Engineering: IOP Conf. Ser.: Mater. Sci. Eng.*, 2021. - №1118. - P.0120003. - URL: <https://iopscience.iop.org/article/10.1088/1757-899X/1118/1/0120003/pdf>.
16. Болотов С.В., Хомченко А.В., Шульга А.В., Болотова Е.Л. Информационно-измерительный комплекс для исследования процессов плавления и переноса электродного металла при дуговой сварке // *Вестник Брянского государственного технического университета*. 2020. №6(91). С.4-11.
17. Bolotov S., Zakharchenkov K., Makarov E., Furmanov V. Intelligent Welding Control System // *CEUR Workshop Proceedings*. 2021. 2965, pp. 260-267.
18. Болотов С.В. Регистратор параметров сварочных процессов РСП-БРУ-01 // Сб. тр. XVI междунар. науч.-техн. конф. «Современные проблемы машиностроения». – Томск: ТПУ, 2021. С.197-198.
19. Болотов С.В., Захарченков К.В., Макаров Е.В., Фурманов В.А. Технология повышения эффективности контроля сварочных процессов в интеллектуальной системе weldingcontrol //: Сб. тр. XI междунар. науч.-техн. конф. «Энергетика, информатика, инновации – 2021». В 2 т. Т 2. 2021. С. 227-230.

МОДЕЛИ ФОРМИРОВАНИЯ ОПЕРАТИВНЫХ КАЛЕНДАРНЫХ ПЛАНОВ НА ОСНОВЕ ПРИОРИТЕТНЫХ СХЕМ

Бурдо Г.Б., д.т.н., профессор

Тверской государственной технической университет

e-mail: gbtms@yandex.ru

Королева М.Н., к.т.н., доцент

МГТУ им. Н.Э. Баумана

e-mail: Maria.Svyatkina@gmail.com

1. ВВЕДЕНИЕ

Одной из наиболее трудоемких задач, решаемых в рамках управления работой технологических подразделений в условиях многономенклатурного машиностроительного производства, является определение последовательности прохождения деталей по рабочим местам (операциям) – построение календарных планов-графиков [1,2]. Задача является достаточно непростой и при использовании для этих целей автоматизированных систем управления технологическими процессами, так как строгое решение задачи методами полного перебора или «ветвей и границ» не обеспечивает нужной быстроты принятия решений в условиях динамичного многономенклатурного производства. Попытки использования различных приоритетных схем [3, 4, 5] не увенчались успехом, прежде всего в связи с тем, что они не учитывали назначение работ, сроки их выполнения и изменение приоритетов при переходе в следующие интервалы оперативного планирования. Поэтому постановка задачи формального синтеза календарных планов-графиков в условиях многономенклатурного машиностроительного производства с помощью приоритетных схем, отражающих иерархию планов организации, и учитывающих текущее состояние в технологических подразделениях, является востребованной.

2. ПРИОРИТЕТЫ ПО РАЗРЯДАМ И ГРУППАМ РАБОТ

Основой для построения календарных планов-графиков служат данные по номенклатуре изготавливаемых изделий (или объемах работ в

станкоёмкостях) с разбивкой по месячным периодам в пределах интервала выполнения текущего контракта, и сведения по маршрутам обработки деталей с нормами времени [5]. Суть приоритетных схем отвечает идеологии бережливого производства и управления качеством [6-10].

Календарные планы-графики разрабатываются для каждого технологического подразделения на интервалы оперативного планирования, длительность которых обычно принимается равной пяти дням.

На первом шаге исходя из пятидневного запаса времени по циклам определяется перечень плановых видов работ $\{B_{lmp}\} = P_n$, где l - соответствуют партии деталей (ее величина n_l); m - рабочему месту (группе оборудования в пределах типа), p - номеру операции в маршруте. Для деталей с выносными операциями (термообработка, сварка), осуществляемыми в рамках данного технологического подразделения, плановый объем работ по ним ограничивается, исходя из: а) соотношения длительности выносных операций и механообработки; б) числа деталей в партии; в) интервалом оперативного планирования. При принятом нами пятидневном периоде, для крупных и средних деталей обработка планируется до 1-2-ой операции после выносной, для мелких – до 2-3-ей операции после выносной. Для выносных операций, выполняемых на стороне, обработка планируется до них.

На втором шаге диспетчирования определяется множество невыполненных на предыдущем интервале оперативного планирования видов работ $\{B_{lmp}^a\} = P_a$, и формируется множество дополнительных работ. Учитывая последующее установление приоритетов, их разбивают на группы: исправление (замена) брака $\{B_{lmp}^b\} = P_b$; новые приоритетные заказы $\{B_{lmp}^c\} = P_c$; изготовления оснастки $\{B_{lmp}^d\} = P_d$; работы по ремонту изделий $\{B_{lmp}^p\} = P_p$. В некоторых случаях ряд работ может быть исключен $\{B_{lmp}^r\} = P_r$. Перечень некоторых других видов работ может быть специфичным для каждого типа предприятия и каждой производственной ситуации.

Множество работ на данном q -м интервале оперативного планирования P_q получается объединением (отсечением) всех видов работ:

$$P_q = (P_n \cup P_a \cup P_b \cup P_c \cup P_d \cup P_p) \setminus P_r.$$

Так как работы имеют различные сроки выполнения, их разбивают на **разряды**: работы P_1 , которые необходимо закончить в планируемом интервале оперативного планирования; работы P_2 , срок завершения которых находится за его пределами и жестко не определен: $P_q = P_1 \cup P_2$.

На третьем шаге устанавливаются приоритеты работ, используемые при дополнении и исключении работ. Работы P_1 представим группами:

$P_1 = P_1^1 \cup P_1^2 \cup P_1^3 \cup P_1^4 \cup P_1^5 \cup P_1^6 \cup P_1^7$, где P_1^1 - плановые работы, соответствующие полной обработке деталей; P_1^2 - плановые работы, соответствующие части маршрута обработки деталей с длинным циклом (большим числом операций); P_1^3 - работы по исправлению (переделке) брака по работам P_1^1 и P_1^2 , выполненных в $(q-1)$ интервале оперативного планирования; P_1^4 - новые срочные приоритетные заказы; P_1^5 - работы по изготовлению средств технологического оснащения, необходимых в данном, q -м интервале оперативного планирования; P_1^6 - другие работы со сроками исполнения в $q+(1\div 3)$ интервале оперативного планирования; P_1^7 - невыполненные работы в $(q-1)$ интервале оперативного планирования. Заметим, что приоритет работ P_1^4 устанавливается системой управления организацией (наивысший). Если ряд приоритетов большой, их можно сравнивать попарно, не создавая логического противоречия в пределах ряда. Скажем, работы P_1^3 должны иметь максимальный приоритет, т.к. должны быть закончены в предыдущем интервале оперативного планирования. Работы P_1^5 должны иметь более высокий приоритет, чем приоритет работ, для которых оснастка необходима. Можно продолжить ряд, однако в каждом конкретном случае приоритет во многом определяется логикой ситуации. Рекомендуется устанавливать шкалу приоритетов, «0» - максимальный. Если работы могут быть выполнены за счёт локальных резервов, то приоритет не присваивается. В нашем примере приоритеты могут быть такие: «0» приоритет имеет P_1^3 ; 1-й приоритет работы P_1^2 и P_1^5 (считаем, что оснащение необходимо для выполнения части работ P_1^1), 2-й - P_1^4 . Если работы P_1^6 не могут быть компенсированы локальными резервами, то их приоритет не может быть выше, чем плановых, и должен увеличиваться при переходе в следующий интервал оперативного планирования. Работы P_1^4 специально выделены в отдельную группу, им при необходимости присваивается приоритет «00» - безусловное выполнение (сверхсрочные работы). Работам P_1^7 , как и работам P_1^3 , должен быть

присвоен приоритет « \emptyset », так как они должны быть выполнены еще в $(q-1)$ -м интервале оперативного планирования.

Были сформулированы правила (продукции) назначения внутренних приоритетов работ, позволяющие формально управлять дополнением и исключением работ из разряда P_1 .

Работы, имеющие меньшее значение приоритета на q -м интервале оперативного планирования, имеют предпочтение для выполнения. Учитывая, что часть работ P_1 , может быть выполнена за счёт локальных резервов (P_1^l), окончательно получаем список работ разряда P_1 :

$$P_1 = (P_1^1 \cup P_1^2 \cup P_1^3 \cup P_1^4 \cup P_1^5 \cup P_1^6 \cup P_1^7) / P_1^l.$$

Проанализируем возможные множества (группы) работ разряда P_2 : $P_2 = P_2^1 \cup P_2^2 \cup P_2^3 \cup P_2^4$, где P_2^1 - плановые работы из следующего периода заключённых контрактов, начало которых, исходя из цикла их выполнения, отдалено более 4-5 интервалов оперативного планирования; P_2^2 - работы, окончание которых более, чем через 2-3 месяца, предназначенные для внутренних нужд - разовые заказы с длительным сроком; P_2^3 - работы по плановой замене изнашивающихся средств технологического оснащения; P_2^4 - другие работы, величина приоритета которых устанавливается системой управления организацией, вплоть до перевода их в множество P_1 с приоритетом « \emptyset ». Часть этих работ может быть выполнена за счёт локальных резервов (P_2^l). Тогда окончательное множество работ P_2 : $P_2 = (P_2^1 \cup P_2^2 \cup P_2^3 \cup P_2^4) / P_2^l$. Установлены правила назначения внутренних приоритетов работ разряда P_2 .

3. ИЕРАРХИЯ ПРИОРИТЕТОВ ПРИ ЗАПУСКЕ ИЗДЕЛИЙ

Составление календарных планов-графиков предполагает определение последовательности, в которой все множество деталей будет проходить по рабочим местам. Чтобы исключить полный перебор вариантов при закреплении детали-операций, необходимо установить систему приоритетов.

Приоритеты устанавливаются в два этапа. На первом этапе устанавливаются приоритеты по разрядам работ P_1 и P_2 , это приоритет глобальный (P_1 приоритетнее P_2). Внутри каждого разряда работ P_j имеются группы работ $\{P_j^i\}$, также имеющие приоритетные уровни. Назовём их внутренним приоритетом. Задача - установить приоритетность видов работ в пределах каждого уровня внутреннего приоритета, определяемого как частный приоритет. Таким образом, иерархия приоритетов выглядит как: глобальный приоритет (разряды

работ) - внутренний приоритет (группы работ) - частный приоритет (виды работ)

Установление приоритетности видов работ обязательно, т.к. внутри одной группы работ (одного внутреннего приоритета) может быть большое число ее видов, характеризующихся разной трудоёмкостью изготовления деталей, разным числом операций и длительностью цикла. Временной фактор окончания работ здесь не принимается во внимание, т.к. был учтён ранее при наборе работ на q -м интервале оперативного планирования. Рассмотрим возможные частные приоритеты применительно к позаказной системе планирования производства в условиях машиностроительного производства на основе анализа. Основные требования для системы частных приоритетов – их непротиворечивость и исключение неопределённых ситуации. Каждая система частных приоритетов должна иметь свою иерархию. Любая система управления должна иметь резервы - возможность учитывать сверхсрочные работы (заказы) [5,7,8]. С этой целью для этих вводятся высший приоритет (PR_1) заказов (вне очереди « $\emptyset\emptyset$ »).

Представляется целесообразным исключить и простой рабочих мест. Поэтому второй частный приоритет PR_2 : если не загружена в данный момент группа станков, на неё передаётся первая появившаяся партия деталей, требующая обработки на данной группе оборудования. Учитывая важность приоритета PR_2 для условий многономенклатурного машиностроения, его следует перенести в разряд высших приоритетов. Приоритеты PR_1 и PR_2 являются основополагающими, поэтому должны присутствовать в любой системе частных приоритетов. Третий приоритет исходит из посылки, что лучше быстрее разгрузить данную операцию (есть движение) – приоритет меньшего времени выполнения операции (приоритет «минимальной операции») - PR_3 . PR_3 : Первой из очереди выполняется операция над той партией деталей, для которой ее длительность меньше. Четвертый приоритет исходит из того, что предпочтение следует отдавать партиям деталей, имеющим максимальное остаточное время обработки [4,5]. PR_4 : Первой из очереди выполняется операция над той партией детали, для которой суммарная длительность обработки от данной операции до последней, запланированной в q -м интервале оперативного планирования, максимальна. Смысл приоритета в обеспечении загрузкой большего числа последующих рабочих мест, и создания резервов времени для более длительных маршрутов обработки, увязывая длину маршрута с остаточной трудоёмкостью. Приоритет абсолютный, предпочтительно

его сделать относительным. Обозначив коэффициент предпочтения K_{pr} , и определив его как приоритет «относительной остаточной длительности»:

$$K_{pr} = 1/(\sum_{K_p}^{K_q} t_o) \text{ или } K_{pr} = (1 + q_i - q_s)/(\sum_{K_p}^{K_o} t_o), \quad (1)$$

где K_q – номер последней операции для партии деталей, выполняемой в q -м интервале оперативного планирования; K_p – рассматриваемая операция, K_o – их общее число; t_o – длительность j -ой операции; q_i, q_s – рассматриваемый интервал оперативного планирования, в котором планировалось окончание обработки. Для деталей, полностью обрабатываемых в q -м интервале оперативного планирования:

$$K_{pr} = 1/(\sum_{K_p}^{K_o} t_o). \quad (2)$$

Чем меньше значение K_{pr} , тем выше частный приоритет партии деталей. Использование формы K_{pr} в соответствии с (1) удобно: первый вид записи пригоден для работ из разряда P_1 , второй для работ из разряда P_1 и P_2 , (для работ из разряда P_2 часть маршрута, выполняемая в q -м интервале оперативного планирования, строго не определена). Для работ разряда P_1 вместо первой формулы в (1), может применяться и (2). Исходя из посыла, что в связи с обновлением очередей, какая-то партия деталей может неопределенно долго ожидать обработки, следует ввести приоритет «максимального времени пребывания в очереди» - PR_5 : ЕСЛИ партия деталей пребывает в очереди больше величины t ($t = k \times T_i$, где $k < 1$, T_i – длительность интервала оперативного планирования), то она приобретает высший частый приоритет после PR_1 . Следующий приоритет PR_6 : - «минимальной длительности 1-й части маршрута с учетом длительности 2-й» исходит из следующих посылов: а) необходимо рассматривать маршрут обработки деталей состоящим из двух частей, что дает более взвешенную картину, чем анализ только данной операции или оставшегося маршрута; б) необходимо как можно быстрее пройти первую часть маршрута и загрузить станки на второй части; в) нельзя допускать перегрузку оборудования 2-ой части маршрута. Действия таковы. Определяются трудоёмкости обработки i -й партии деталей по первой T_{1i} и 2-ой T_{2i} частям маршрута, и рассчитывается $\Delta T_{12i} = T_{2i} - T_{1i}$. Для i -х партий деталей приоритет устанавливается по правилам:

а) партии деталей с $\Delta T_{12i} > 0$ имеют высший приоритет по отношению к деталям с $\Delta T_{12i} < 0$;

б) у партий деталей с $\Delta T_{12i} > 0$ приоритеты уменьшаются по мере возрастания величины T_{1i} (запуск в порядке увеличения T_{1i});

в) у партий деталей с $\Delta T_{12i} < 0$ приоритеты уменьшаются в порядке уменьшения T_{2i} . Учитывая, что на предприятиях многономенклатурного машиностроения оборудование расположено по типам, и нельзя гарантировать отсутствия обратных потоков деталей на пройденные ранее группы станков, следует рассматривать и синтез вариантов по способу, противоположному PR_6 (обозначим его как $-PR_6$ – приоритет «меньшей длительности 2-й части маршрута с учетом 1-й») (первыми запускаются детали с $\Delta T_{12i} < 0$ в порядке уменьшения T_{2i} , затем детали с $\Delta T_{12i} > 0$ в порядке уменьшения T_{1i} . Неопределенности, вызванные статистической длительностью операций и равенством приоритетов, исключаются добавлением приоритета PR_7 «первым пришёл - первым обслужен», т.е. порядком поступления в очередь.

Упрощённой разновидностью PR_6 может служить приоритет PR_8 «меньшей длительности первой части маршрута»: ЕСЛИ в очереди несколько партий деталей, ТО запуск осуществляется в порядке уменьшения величины ΔT_{12i} . Посыл – быстрее загрузить рабочие места 2-ой части маршрута. Следует рассмотреть и PR_8 , обратный PR_8 - «меньшая длительность 2-ой части маршрута».

Можно рассмотреть такой логический посыл: если за данной операцией имеются недогруженные группы станков (по данным подсистемы объемного планирования), то их следует быстро догрузить - приоритет PR_9 «загрузка последующих операций»: ЕСЛИ партия деталей после данной операции поступает на недогруженное оборудование, ТО она имеет на данной операции высший приоритет после PR_1 и PR_5 .

Так как таких партий может быть несколько, то для устранения неопределенности необходимо добавление приоритетов PR_3 и (или) PR_4 и (или) PR_7 . Добавление PR_3 будет означать быструю загрузку следующего рабочего места; PR_4 - загрузку большего числа последующих рабочих мест; PR_7 служит для разрешения статистической неопределенности.

Следующий посыл исходит из того, что могут быть группы станков, загруженные выше, нежели остальные, поэтому могут возникать «узкие места» в потоках деталей. Целесообразно обеспечить прохождение до «узкого места» партиями деталей большего числа операций. Поэтому приоритет PR_{10} может быть назван как «приоритет удалённости узкого места»: ЕСЛИ имеется несколько партий деталей, имеющих «узкие места», ТО первой запускается партия, имеющая более далекое расположение «узкого места». Для снятия неопределенности (одинаковое число операций до узкой) должны использоваться PR_3 и (или) PR_4 и (или) PR_5 и (или) PR_7 .

Рассмотрим такое предположение: если деталь пришла первой на какую-то операцию, значит, управляющие действия были направлены на это, и тенденцию надо продолжить. PR_{11} обозначим, как приоритет «первенства»: ЕСЛИ имеется несколько партий деталей, пришедших в очередь в интервал ΔT_i , то они имеют приоритет над деталями из ΔT_{i+1} интервала.

Исходя из конкретных ситуаций в производственной системе, возможно обоснование и других посылов, но исследованные нами ситуации охватывают основную область поиска решений.

4. ПРИНЦИПЫ ФОРМИРОВАНИЯ ПРИОРИТЕТОВ

Сформулированы основные принципы формирования приоритетов (иерархичность, целевая направленность, определенность выбора, динамичность параметров в системах приоритетов, постоянство важнейших приоритетов, вариабельность, ситуативность).

Иерархичность подразумевает определение уровней важности приоритетов и их ранжирование. Целевая направленность приоритетов должна отвечать целям функционирования организации, т.е. обеспечивать определенную направленность календарных планов-графиков. Определенность выбора - включение в систему приоритетов, позволяющих однозначно их трактовать. Динамичность и ситуативность предполагает при выборе схем анализировать фактические параметры состояния технологических подразделений (PR_2, PR_3, PR_4, PR_5 и т.д.), а также численное изменение (PR_4, PR_6) приоритетов. Приоритеты, отражающие смысл функционирования производственной системы (глобальный, внутренний), действуют постоянно. Это обеспечивает вариабельность календарных планов-графиков наложением частных (вариативных) приоритетов,

определяемых управляющей подсистемой, на инвариантную часть. Выбор приоритетов должен исходить из анализа ситуаций в технологических подразделениях, выраженных в посылах, что позволяет расширить пространство поиска решений.

Обосновано и сформировано 11 систем иерархических приоритетных схем, выбираемых продукционными моделями знаний на основе формального распознавания и анализа ситуации в производстве (загрузки оборудования по ходу основного детали- потока, длительности и числа операций в маршрутах запускаемых деталей, времени поступления деталей, наличия свободного оборудования, длительности первой следующей операции и т.д.). На основе правил разработаны формальные алгоритмы выбора схем для расчёта календарных планов-графиков. Расчёт только по одной схеме нецелесообразен, т.к. посылы определяются лишь основными факторами, действующими в производственных системах. Поэтому календарные планы-графики необходимо синтезировать по нескольким схемам, выбирая из них наилучшую. В некоторых случаях возможен расчёт и по 1-2-м упрощенным схемам, в случае удовлетворительного результата календарный план-график считается составленным.

5. ЗАКЛЮЧЕНИЕ

Предлагаемая методика может явиться эффективным инструментом для формирования календарных планов-графиков в автоматизированных системах управления технологическими процессами в условиях многономенклатурного машиностроения, так как учитывает большинство наиболее вероятных производственных ситуаций. Важным является и тот факт, что система приоритетов является достаточно гибкой, что обеспечивает ее малую инерционность по отношению к изменяющейся производственной ситуации в технологических подразделениях. Собственно календарный план-график на основе приоритетных схем может быть построен на основе имитационного моделирования [3]. Учитывая, что при разработке моделей системы могут применяться их различные классы (продукционные, аналитические и т.д., в том числе и человеко-машинные), их целесообразно реализовывать на основе методов нечетких множеств и гибридных систем [9, 11-13].

Литература

1. Бурдо Г.Б. Принципы построения автоматизированной системы управления технологическими процессами в многономенклатурных производствах //Вестник Саратовского государственного технического университета.- 2010.- №3(48).- С. 113–118.
2. Самойлович В.Г. Организация производства и менеджмент. - М.: Издательский центр «Академия», 2008.
3. Макаров И.М., Рахманов В.З., Назаретов В.М. и др. Робототехника и гибкие автоматизированные производства. Кн. 3. Управление робототехническими системами и гибкими автоматизированными производствами / Под ред. И.М. Макарова. - М.: Высшая школа, 1986.
4. Бурдо Г.Б., Стоянова О.В., Исаев А.А., Чаплыгин С.А. Гибридные интегрированные автоматизированные системы проектирования и управления технологическими процессами. //Труды конгресса по интеллектуальным системам и информационным технологиям "IS&IT'17". 2-9 сентября 2017, Дивноморское, Россия. – С. 94-101.
5. Суслов А. Г. Дальский А. М. Научные основы технологии машиностроения М.: Машиностроение, 2002.
6. Вумек Джеймс П., Джонс Даниел Т. Бережливое производство. Как избавиться от потерь и добиться процветания вашей компании. – М.: Альпина Бизнес Букс, 2008.
7. James P. Womack, Daniel T. Jones, Lean Thinking (Second Edition), New York: Free Press, 2003.
8. Shigeo Shingo. Kaizen and The Art of Creative Thinking, Enna Product Corporation and PCS Inc, 2007.
9. L Zadeh. Fuzzy Sets. Information and Control, 1965; 8, P. 338-353.
10. ГОСТ ISO 9001-2011. Системы менеджмента качества. Москва, Стандартинформ, 2012.
11. Mori H., Aoyamu H., Yamanaka T., Urano S. A Hybrid Intelligent System for Fault Detection in Power Systems. – IEEE, 2002. – P. 2138-2153.
12. Nikolopoulos C., Fellrath P. A Hybrid Expert System for Investment Advising. – IEEE, 1994. – P. 1818-1828.
13. Гаврилов А.В. Гибридные интеллектуальные системы: Монография – Новосибирск: Изд-во НГТУ, 2002.

КЛАССИФИКАЦИЯ МЕТОДОЛОГИЙ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Гурьянов Д.А., аспирант

e-mail: guryanovda@student.bmstu.ru

Мышенков К.С., д.т.н., проф.

e-mail: myshenkovks@bmstu.ru

МГТУ им. Н.Э. Баумана

1. ВВЕДЕНИЕ

Процесс управления моделью жизненного цикла (ЖЦ) программного обеспечения (ПО) способствует тому, чтобы гарантировать пригодность модели, процессов и практик ЖЦ для использования организацией при разработке программных проектов [1]. Согласно исследованиям Standish Group [2], по состоянию на 2020 г., только 31% проектов в области разработки ПО признаны успешными. Одним из ключевых факторов, влияющих на высокий процент провальных и завершенных с затруднениями проектов, является неправильный выбор методологии разработки при запуске проекта.

Проблемами при выборе методологии разработки ПО являются: большое количество вариантов; литературные источники обычно описывают только несколько известных методологий; низкий уровень формализации процесса выбора; выбор базируется только на предыдущем опыте менеджера проекта. Перечисленные проблемы приводят к увеличению риска неправильного выбора методологии и это уменьшает вероятность успешного завершения проекта [2, 3]. Сравнение и выбор наиболее подходящей методологии представляет собой сложную задачу, которая требует не только широких теоретических познаний, но и консультации экспертов-практиков [3].

2. МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНЫХ СИСТЕМ

Период существования программной системы, который начинается с момента замысла, продолжается в течение эволюции и заканчивается утилизацией программной системы называется жизненным циклом программной системы [4]. Рассмотрим наиболее

известные модели ЖЦ программных систем.

Взрывное проектирование является неструктурированной моделью ЖЦ и состоит только из одного этапа – разработки [5]. Эта модель ЖЦ применяется для коротких проектов, если клиент не может четко сформулировать свои требования. Область применения: академические проекты или адаптация сотрудников в рабочей группе.

Анализ-кодирование является двухэтапной моделью ЖЦ и состоит из двух этапов: анализа и кодирования [5]. Аналогично модели *Взрывного проектирования* эта модель предназначена для коротких проектов (академических), однако, отличается наличием формализованных требований, что позволяет уменьшить риски.

Кодирование-отладка. Данная модель ЖЦ была распространена в коммерческой разработке ПО до предложения доктора У.У. Ройса в 1970 г. распространить *Каскадную* модель ЖЦ на сферу разработки ПО [5]. Это двухэтапная модель ЖЦ, состоящая из этапа написания кода и этапа отладки, что повышает качество выходного продукта.

Каскадная (Waterfall, «Водопадная») модель ЖЦ описана У.У. Ройсом и представляет собой линейную последовательность выполнения этапов без перекрытия и без возможности возврата на предшествующие этапы. Требования к проекту разрабатываются до начала этапа проектирования и не подлежат изменению [5,6].

Модифицированная каскадная модель отличается от *Каскадной* добавлением обратных связей с предыдущими этапами, что позволяет организовать итерации, параллельное выполнение этапов и менеджмент изменений. Это делает модель менее жесткой [5,6].

Инкрементная (Increment, «Увеличение») модель ЖЦ предназначена для сокращения времени между этапами анализа и внедрения для сложных программных систем. Сначала проектируется общая архитектура системы, а затем проект делится на инкременты (подсистемы), которые интегрируются последовательно или параллельно в общую систему. Каждый инкремент проходит последовательно через основные фазы ЖЦ. Этот вид модели ЖЦ часто комбинируют с другими моделями: *Каскадной*, *Спиральной*, *V-образной*. Это позволяет снизить затраты и риски при разработке [7].

Итеративная (Iteration, «Повторение») модель ЖЦ расширяет возможности *Инкрементной* модели за счет того, что позволяет разрабатывать продукт в условиях неопределенности требований. Проект делится на небольшие части, разработчики показывают результаты текущей итерации, уточняют требования и получают отзывы от пользователей системы [7]. Каждая итерация представляет собой комбинацию упрощенной версии *Модифицированной каскадной*

и *Инкрементной* моделей ЖЦ. Модель *Эволюционного прототипирования* является вариантом *Итеративной* модели ЖЦ.

Спиральная (Spiral) модель ЖЦ представлена в 1986 г. Б. Бозмом [8]. Она основана на идеях *Каскадной* и *Итеративной* моделей ЖЦ с увеличенным числом промежуточных фаз выполнения, расположенных в виде спирали, разбитой на квадранты. Каждую итерацию система проходит через каждый из квадрантов, которые отвечают за следующие действия: определение целей, альтернатив и ограничений; выявление и устранение рисков, оценка альтернатив; разработка и тестирование прототипа системы; оценка результатов и планирование следующей итерации.

V-образная (Vee Model) модель ЖЦ была предложена в 1991 г. К. Форсбергом и Г. Музом [9]. Представляет собой проекцию правил *Каскадной* модели на новую модель в форме латинской буквы *V* с дополнительной взаимосвязью (ранним планированием тестирования) между этапами анализа и тестирования, что ведет к повышению качества и надежности разрабатываемого продукта, но при этом требует стабильных требований и предварительного анализа рисков.

W-образная (Dual Vee Model) модель ЖЦ была представлена в 2006 г. К. Форсбергом и Г. Музом [10] на основании анализа недостатков *Каскадной* и собственной *V-образной* модели. Модель предполагает наличие двух ЖЦ, выполняемых по *V-образной* модели – архитектурную *Vee* и сущностную *Vee*. Архитектурная предназначена для реализации взаимодействия архитектурных компонентов системы, сущностная – для создания сущностей и логики.

3. МЕТОДОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В процессе управления программными проектами модели ЖЦ ПО уточняются и реализуются конкретными методологиями разработки, объединяющими в себе корпоративные стандарты, правила, практики, методы и инструкции, способствующие успешной реализации проекта. Проанализируем основные популярные методологии разработки ПО.

Быстрая разработка приложений (RAD) основана на *Итеративной* модели ЖЦ с последовательным развитием продукта и регулярным анализом результатов проекта пользователями. Разработка версии ограничивается периодом 60-90 дней. Высокая скорость разработки достигается за счет CASE-средств, повторного использования кода и четкого планирования [6].

Унифицированный процесс Rational (RUP) – это итеративная

методология, в которой основное внимание уделяется классическим принципам разработки ПО. Это пошаговая методология продвижения качественных объектно-ориентированных проектов. *RUP* основан на детальном планировании, обширном системном анализе, принципах детального проектирования, стандартизированном процессе кодирования и расширенном количестве документации [11].

OpenUP – модификация методологии *RUP*, ориентированная на малые команды (3-6 человек) и небольшие проекты (до 6 месяцев).

Методология *Scrum* основана на принципах эмпирического управления процессами: прозрачность процесса, инспекция и адаптация [12]. Прозрачность означает, что артефакты процесса должны быть видны тем, кто управляет результатами. Инспекция означает, что процессы должны часто контролироваться, чтобы обнаружить отклонения. Адаптация означает необходимость проведения корректировок при отклонениях. Структура *Scrum* состоит из *Scrum*-команды и связанных с ней ролей, таймбоксов, артефактов и правил. Методология *Scrum* спроектирована для команд размером до 12 человек.

Large Scale Scrum (LeSS) – это адаптация методологии *Scrum* для больших команд (до 8 команд по 8 человек). Команды работают в одном спринте для создания общей версии продукта каждый спринт [13].

Dynamic System Development Method (DSDM) была опубликована в 1995 г. Методология позиционируется в качестве стандарта для быстрой разработки ПО и является эволюцией методологии *RAD* [14]. Основные идеи: «Ничто нельзя сделать идеально с первого раза» и принцип Парето – 80% работы можно сделать за 20% времени. Методология *DSDM* основана на *Итеративной* модели ЖЦ и состоит из 3 ключевых фаз: предпроектная, реализации проекта и постпроектная.

Agile Kanban Methodology в отличие от *Scrum* и *DSDM* основана на *Инкрементной* модели ЖЦ, где каждая пользовательская задача рассматривается как подпроект. Принцип, лежащий в основе *Kanban*, это ограниченная пропускная способность команды. Эти проекты имеют лимиты незавершенной работы, которые позволяют команде разработчиков сосредоточиться только на небольшом объеме работы за один этап. Контролируя количество активных задач, разработчики подходят к проекту в целом поэтапно, что дает возможность применять принципы гибкой разработки [6].

Экстремальное программирование (XP) было предложено К. Беком в 2000 году [15], основано на *Итеративной* модели ЖЦ и в

качестве основной идеи рассматривает разработку через тестирование. Экстремальное программирование состоит из известных «лучших практик» программирования, выявленных К. Бекем, но в *XP* они доведены до «экстремальных» уровней, что и отражается в названии.

eXRUP относится к гибридным методологиям, сочетающим в себе практики «парного программирования», являющегося отличительной чертой *XP*, с процессами и CASE-средствами из *RUP* [16].

Agile Unified Process (AUP) – это упрощенная версия *RUP*, ориентированная на совмещение структурного подхода унифицированного процесса с методами гибких методологий *Agile Alliance*. В отличие от *RUP*, в *AUP* процесс разработки сокращен до четырех стадий, используются гибкие практики экстремального проектирования, а также отсутствуют инструменты поддержки процесса. По мнению автора методологии С. Эмблера *AUP* является промежуточным вариантом между *RUP* и *XP* [16].

Методология DevOps основана на *Инкрементной* модели ЖЦ и ориентирована на интеграцию процессов разработки, тестирования и эксплуатации. Организуются кроссфункциональные команды из специально обученных специалистов, которые широко используют инструменты автоматизации разработки, тестирования и развертывания. Культура *DevOps* предполагает, что каждый член команды отвечает за конечный результат. В *DevOps* процессы тестирования и развертывания ПО происходят непрерывно, до нескольких раз в день [6].

Microsoft Solutions Framework (MSF) – методология, основанная на *Итеративной* модели ЖЦ, стимулирует более частую поставку в случае частичной готовности задачи и функционально неполного решения, одновременно повышая ценность каждой поставки. *MSF* рекомендует часто интегрировать и тестировать все компоненты решения, а не дожидаться окончания конструирования для проведения интеграционного тестирования [17].

4. КРИТЕРИИ СРАВНЕНИЯ МЕТОДОЛОГИЙ

Проведем анализ критериев, которые могут лечь в основу классификации методологий разработки ПО.

Способ декомпозиции процессов проекта. М. Фаулер выявил наличие двух концептуальных подходов к построению ЖЦ ПО – каскадного и итеративного [18]. В каскадном проходе применяется линейно-структурная декомпозиция согласно правилам РМВоК. При

итеративном подходе проект разбивается на части, каждая из которых представляет собой подпроект или версию продукта.

Модель жизненного цикла. Стандарт на процессы ЖЦ ISO 12207:2017 [1] явно не регулирует последовательность этапов разработки. В исследованиях У.У. Ройса [5], К. Форсберга и Г. Муза [9,10] и других, были описаны различные модели ЖЦ ПО, рассмотренные выше. Методологии разработки ПО реализуют одну из моделей ЖЦ и добавляют к ней свои особенности в виде практик, артефактов, специального ПО и инструментов, которые поддерживают процесс разработки.

Гибкость процесса. Данный показатель учитывает степень формализации процесса разработки, выраженную в виде возможности возврата на предыдущий этап ЖЦ, объеме контролируемых параметров, возможности корректировки плана и участия заказчика в процессе разработки. В работе А. Кокберна [19] данный показатель называется «методологический вес» и определяет, какие части процесса разработки регулируются методологией, а какие определяются участниками команды.

Метод декомпозиции архитектуры проекта. В работах [20, 21] в основу классификации методологий положены методы декомпозиции архитектуры проекта и методы моделирования аспектов предметной области (виды моделей, диаграмм, нотаций). Структурные методологии базируются на иерархической декомпозиции системы на автоматизируемые функции и раздельном проектировании функций и данных системы. Объектно-ориентированные методологии основаны на представлении системы в виде совокупности классов, которые используют иерархию наследования и взаимодействуют посредством механизмов пересылки сообщений.

Архитектурная сложность проекта. Данный критерий учитывает наличие этапа проектирования в процессе ЖЦ и количество уровней проектирования. Например, может использоваться принцип «простого дизайнера» [15], применяемый в *XP*, требующий дополнительного «непрерывного рефакторинга» или противоположное решение в виде двух этапов проектирования, применяемого *RUP* [16].

Размер команды. Данный критерий приведен в работе А. Кокберна [19] и описывает проблемы взаимодействия внутри команды проекта. Эти проблемы решаются путем введения ограничения на размер команды, как в *Scrum*, или разработкой специальных фреймворков для управления большими командами, например, в *LeSS*.

Приоритеты и принципы. Совпадение приоритетов проекта и приоритетов методологии является одним из важных факторов,

используемых при выборе методологии. Выбор методологии, ориентированной на средства автоматизации процессов, помогает сократить сроки выпуска продукта, а использование командно-ориентированных методологий ускоряет реакцию на изменение требований и повышает удовлетворенность заказчика [22].

5. СУЩЕСТВУЮЩИЕ КЛАССИФИКАЦИИ МЕТОДОЛОГИЙ

Рассмотрим существующие классификации методологий разработки ПО. Обратим особое внимание на то, что в приведенных ниже классификациях, как и в большинстве литературных источников, не разделяются понятия: модель ЖЦ и методология разработки ПО. Они объединены в этом разделе одним термином – методология.

Классификация по гибкости процессов рассматривает методологии с точки зрения выполнения этапов ЖЦ и приоритетов проекта. В случае приоритета процессов и документации – методология относится к классу традиционных, в случае приоритета команды и продукта – к классу гибких. Если методология учитывает оба подхода – она относится к классу гибридных [23]. Классификация рассмотренных выше методологий по гибкости процессов:

- традиционные – *Waterfall, Vee Model, Double Vee;*
- гибридные – *Increment, Iteration, Spiral, RUP, RAD, MSF;*
- гибкие – *Взрывное проектирование, Анализ-кодирование, Кодирование-отладка, Scrum, LeSS, Kanban, DSDM, XP, DevOps, AUP, OpenUP, eXRUP.*

Классификация по размеру команды распределяет методологии по возможности использования в больших и малых командах [13]:

- для малых команд – *Взрывное проектирование, Анализ-кодирование, Кодирование-отладка, RAD, Scrum, Kanban, DSDM, XP, DevOps, AUP, OpenUP;*

- для больших команд – *Waterfall, Increment, Iteration, Spiral, Vee Model, Double Vee, RUP, MSF, eXRUP, LeSS.*

Двухкритериальная классификация основана на наложении двух классификаций на двумерную систему координат. Такой способ позволяет оценить методологии по двум критериям [7]. На рис. 1 представлен пример такой классификации, основанной на критериях: способ декомпозиции процессов проекта и гибкость процесса.

Шкала Кокберна основана на матричном представлении классификации по двум критериям – размеру команды и шкале критичности, описывающей риски проекта: риск потери комфорта (С); риск потери дополнительного дохода (D); риск банкротства (E) и риск

потери жизни (L). Чем выше уровень критичности, тем более формализованная методология требуется для разработки [19]. На рис. 2 представлено распределение методологий по шкале Кокберна.

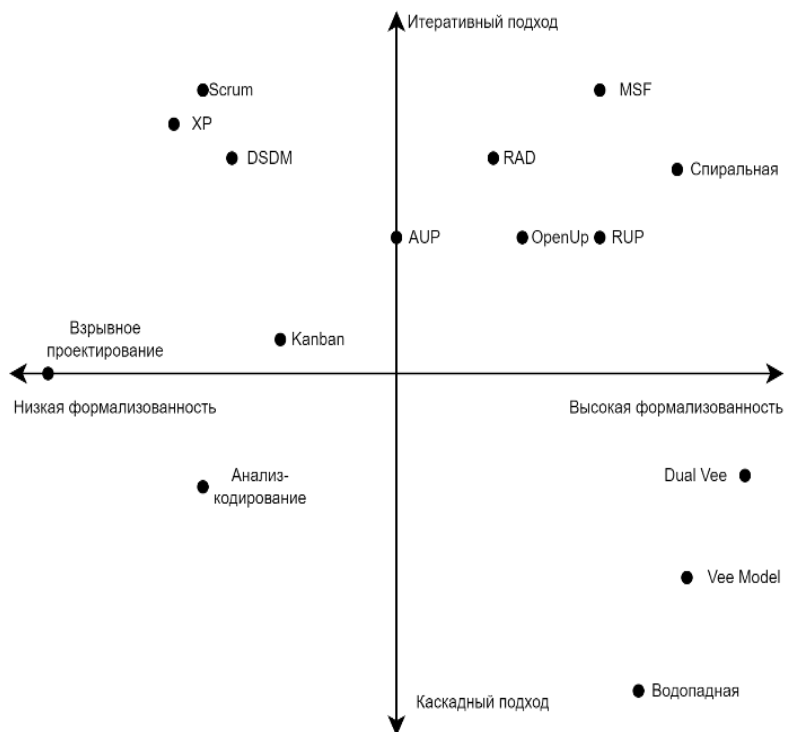


Рис. 1. Двухкритериальная классификация методологий разработки ПО

Для принятия решения о выборе конкретной методологии разработки ПО можно использовать многокритериальные экспертные методы, которые можно разделить на следующие группы с точки зрения измерений:

- основанные на количественных измерениях (многокритериальная теория полезности – MAUT [25]);
- основанные на качественных результатах переведённые в количественный вид (метод анализа иерархий – АНР [26], методы на основе нечеткой логики [3, 27-30]);

- основанные на количественных измерениях с использованием нескольких индикаторов при сравнении альтернатив (семейство методов Electre [31]);
- основанные на качественных характеристиках без перевода в количественный вид (вербальный анализ решений – ВАР [32-34]).

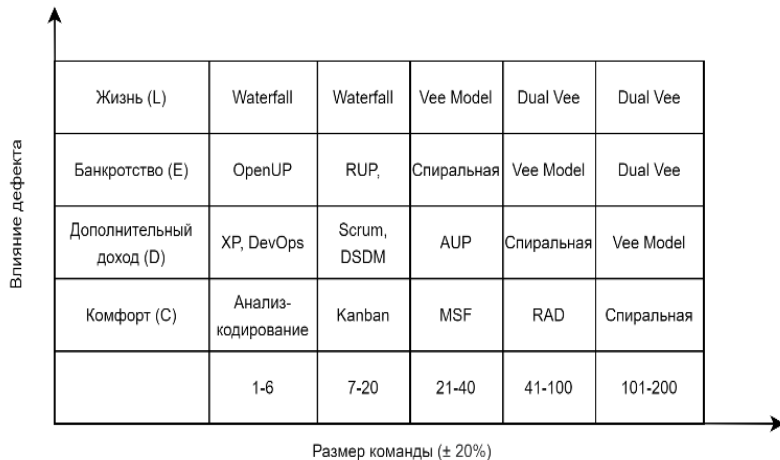


Рис. 2. Классификация методологий по шкале Кокберна

С точки зрения применения на практике особенно выделяют по популярности такие методы, как: экспертные системы на основе правил, куб принятия решений, АНР, методы с использованием нечеткой логики, привлечение стороннего экспертного мнения, директивные указания и метод неформализованного выбора [3]. После исключения из списка методов, основанных на интуиции, предыдущем опыте ЛПП и директивных указаниях, в списке остаются методы, к основным недостаткам которых следует отнести:

- недостаточную обоснованность при переводе качественных характеристик в количественные (АНР, куб принятия решений);
- ограничение применимости модели вынуждает объединять близкородственные методологии в группы альтернатив, что не дает ЛПП однозначного лучшего варианта выбора (куб принятия решений, методы нечеткой логики, экспертные системы на основе правил);
- отсутствие учёта индивидуальных особенностей команды, таких как её зрелость, опыт, навыки членов команды (куб принятия решений, экспертные системы на основе правил).

В качестве перспективного метода, лишенного вышеописанных недостатков, можно выделить метод ретроспективного анализа предыдущих реализаций проектов с применением методов машинного обучения для прогнозирования выходных параметров проекта и вычисления наиболее эффективной методологии. Однако, применение данного метода требует накопления достаточного объема исторических данных по проектам, что сужает границы его применимости только конкретной командой разработки. Таким образом, для недавно образованной команды рекомендуется выбрать один из ранее перечисленных методов, предназначенных для принятия решений в условиях неопределённости.

6. ИЕРАРХИЧЕСКАЯ КЛАССИФИКАЦИЯ МОДЕЛЕЙ ЖИЗНЕННОГО ЦИКЛА И МЕТОДОЛОГИЙ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Описанные способы классификации методологий обладают следующими недостатками: смешаны понятия модели ЖЦ и методологии разработки ПО; недостаточное количество анализируемых критериев и большие размеры кластеров. Предлагаемая иерархическая классификация моделей ЖЦ и методологий разработки ПО, свободная от перечисленных недостатков, приведена в таблице и представляет собой иерархию, каждый уровень которой определяет одну из важных характеристик качества проекта и ориентацию методологии [24].

По характеристике проекта СОПРОВОЖДАЕМОСТЬ (уровень стабильности требований) методологии разделены на три класса: ЖЕСТКИЕ – требования к проекту полностью определены и в процессе реализации могут лишь уточняться; ГИБРИДНЫЕ – требования к проекту частично определены, а в процессе реализации изменяются; ГИБКИЕ - требования к проекту уточняются в процессе его реализации.

По характеристике проекта НАДЕЖНОСТЬ методологии разделены на следующие классы: ВЫСОКОНАДЕЖНЫЕ; НАДЕЖНЫЕ; СРЕДНЕЙ НАДЕЖНОСТИ; НИЗКОЙ НАДЕЖНОСТИ.

По характеристике проекта ФУНКЦИОНАЛЬНОСТЬ (уровень сложности ПО, на этом же уровне определены модели ЖЦ, на которых базируются методологии) методологии разделены на три класса: ВЫСОКОЙ СЛОЖНОСТИ; СЛОЖНЫЕ; ПРОСТЫЕ.

Таблица. Иерархическая классификация моделей ЖЦ и методологий разработки ПО

| СОПРОВОЖДАЕМОСТЬ (стабильность требований) | НАДЕЖНОСТЬ | ФУНКЦИОНАЛЬНОСТЬ (сложность) <i>Модель ЖЦ</i> | ОРИЕНТАЦИЯ (приоритет методологии) <i>Методология</i> |
|--|-------------------------------------|--|--|
| ЖЕСТКИЕ (строгие требования) | ВЫСОКОЙ НАДЕЖНОСТИ | ВЫСОКОЙ СЛОЖНОСТИ <i>W-образная</i> | |
| | | СЛОЖНЫЕ <i>V-образная</i> | |
| | НАДЕЖНЫЕ | ВЫСОКОЙ СЛОЖНОСТИ <i>Инкрементная</i> | |
| | | СЛОЖНЫЕ <i>Каскадная</i> | |
| ГИБРИДНЫЕ (требования изменяются) | НАДЕЖНЫЕ | СЛОЖНЫЕ <i>Итеративная, Спиральная</i> | Процессно-ориентированные <i>MSF</i> |
| | | | Объектно-ориентированные <i>RUP</i> |
| ГИБКИЕ (требования постоянно уточняются) | СРЕДНЕЙ НАДЕЖНОСТИ | СЛОЖНЫЕ <i>Итеративная</i> | Процессно-ориентированные <i>AUP, eXRUP</i> |
| | | ПРОСТЫЕ <i>Итеративная</i> | Инструментальные <i>RAD, Kanban, DevOps, LeSS, OpenUP</i> |
| | Командные <i>XP, Scrum, DSDM</i> | | |
| | НИЗКОЙ НАДЕЖНОСТИ | ПРОСТЫЕ <i>Взрывное проектирование, Анализирование, Кодирование-отладка</i> | |

По **ОРИЕНТАЦИИ** проекта (приоритетам и принципам) методологии разделены на следующие группы: **ПРОЦЕССНО-ОРИЕНТИРОВАННЫЕ** – ориентированные на структуру процессов (фазы) ЖЦ ПО; **ОБЪЕКТНО-ОРИЕНТИРОВАННЫЕ** – ориентированные на сценарии процессов ЖЦ ПО и объектно-ориентированные методы разработки; **ИНСТРУМЕНТАЛЬНЫЕ** – ориентированные на команды разработчиков, активно использующие инструменты автоматизации разработки, тестирования и развертывания ПО; **КОМАНДНЫЕ** – ориентированные на небольшие команды разработчиков, для которых программный продукт важнее процессов и документации проекта.

7. ЗАКЛЮЧЕНИЕ

Согласно исследованиям Standish, правильный выбор методологии разработки программного обеспечения повышает вероятность успешного завершения проекта на 16% [2]. В данной статье рассмотрены основные модели ЖЦ и методологии разработки ПО, проанализированы существующие классификации методологий и выявлены их недостатки. Предложен новый вариант классификации, предлагающий более полное и широкое иерархическое распределение моделей ЖЦ и методологий разработки ПО. Рассмотрены существующие многокритериальные методы, с помощью которых можно принять решение о выборе методологии при запуске новых проектов в области разработки ПО.

Полученные результаты помогут более обоснованно осуществлять выбор между существующими методами проектирования, улучшить качество планирования проектной деятельности, повысить эффективность управления проектами, а также могут быть использованы для разработки новых подходов по оптимизации процесса разработки ПО, таких как метод синтеза методологии разработки из отдельных независимых элементов: модели ЖЦ, методов, средств и принципов, применяемых в процессе разработки ПО.

Литература

1. ISO/IEC/IEEE 12207:2017 Systems and software engineering – Software life cycle processes [Электронный ресурс]. – URL: <https://www.iso.org/ru/standard/63712.html> (дата обращения: 10.02.2022).
2. Review Standish Group – CHAOS 2020: Beyond Infinity [Электронный ресурс]. – URL: <https://hennyportman.wordpress.com/2021/01/06/review-standish-group-chaos-2020-beyond-infinity/> (дата обращения: 10.02.2022).

3. Ozturk V. et al. Selection of Appropriate Software Development Life Cycle Using Fuzzy Logic // *Journal of Intelligent and Fuzzy Systems*. – 2013. – Vol. 25. – P. 797-810.
4. ISO/IEC 15704:2019 Enterprise modelling and architecture – Requirements for enterprise-referencing architectures and methodologies [Электронный ресурс]. – URL: <https://www.iso.org/obp/ui/#iso:std:iso:15704:ed-2:v1:en> (дата обращения: 10.02.2022).
5. Royce W.W. *Managing the Development of Large Software Systems: Concepts and Techniques* // *Proc. IEEE WESCOM*. IEEE Computer Society Press. – Los Alamitos, 1970. – P. 1-9.
6. Font Jr. V.M. *The Ultimate Guide to SDLC*. – FontLife, 2012.
7. Фатрелл Р.Т., Шафер Д.Ф., Шафер Л.И. *Управление программными проектами: достижение оптимального качества при минимуме затрат*. – М.: Изд. дом «Вильямс», 2004. – 1136 с.
8. Boehm B.W. *A Spiral Model of Software Development and Enhancement* // *Proceedings of an International Workshop on Software Process and Software Environments*. – Coto de Caza, Trabuco Canyon, California, 27-29 March 1985.
9. Mooz H., Forsberg K. *The Relationship of System Engineering to the Project Cycle* // *Proc. International Symposium of the International Council on Systems Engineering (INCOSE)*. – Oct. 1991. – P. 57-65.
10. Mooz H., Forsberg K. *The Dual Vee – Illuminating the Management of Complexity* // *Proc. Sixteenth Annual International Symposium of the International Council on Systems Engineering (INCOSE)*. – Jul. 2006. – P. 1368-1381.
11. Sharma N., Wadhwa M. *eXRUP: Hybrid Software Development Model Integrating Extreme Programming, Scrum & Rational Unified Process* // *TELKOMNIKA Indonesian Journal of Electrical Engineering*. – 2015. – Vol. 16. – № 2.
12. Schwaber K. *The Scrum Guide*. – Scrum Alliance, 2011.
13. Ahmed Z., Waleed A., Waqas M. *Preference in using Agile Development with Larger Team Size* // *International Journal of Advanced Computer Science and Applications*. – 2018. – Vol. 9. – P. 116-123.
14. Stapleton J. *DSDM: Dynamic Systems Development Method* // *Proceedings Technology of Object-Oriented Languages and Systems*. – 1999. – TOOLS 29. – Cat. № PR00275.
15. Beck K. *Extreme Programming Explained: Embrace Change*. – Addison-Wesley, 1999.
16. Amber S.W. *The Agile Unified Process (AUP)*. – Ambyssoft, 2005.
17. Essama Ze S. *Microsoft Solutions Framework (MSF) for Agile Software Development*. – 2014.
18. Fowler M. *The New Methodology* // *Journal of Natural Science*, Wuhan Univ. – 2001. – Vol. 6. – P. 12-24.
19. Cockburn A. *Selecting a Project's Methodology* // *IEEE Software*. – July-Aug. 2000. – Vol. 17. – № 4. – P. 64-71.

20. Мышенков К.С., Беляшов А.Н. Методологии, методы и модели для анализа и проектирования систем управления // Вестник МГТУ «Станкин». – 2013. – № 3 (26). – С. 76-82.
21. Мышенков К.С. Методика обоснования выбора CASE-средств для анализа и проектирования систем управления предприятиями // Инновации. – 2013. – № 10. – С. 33-43.
22. Dingsoy T. et al. A Decade of Agile Methodologies: Towards Explaining Agile Software Development // The Journal of Systems and Software. – 2012. – Vol. 85. – P. 1213-1221.
23. Ciric D. et al. Agile vs. Traditional Approach in Project Management: Strategies, Challenges and Reasons to Introduce Agile // Proc. 25th International Conference on Production Research Manufacturing Innovation: Cyber Physical Manufacturing. – Aug. 2019. – P. 1408-1414.
24. Гурьянов Д.А., Мышенков К.С., Строганов В.Ю. Анализ методологий разработки программного обеспечения // Нейрокомпьютеры и их применение: Тезисы докладов XX Всеросс. научн. конф. – М.: МГППУ, 2022. – С. 52-53.
25. Кини Р.Л., Райфа Х. Принятие решений при многих критериях: предпочтения и замещения. – М.: Радио и связь, 1981. – 560 с.
26. Саати Т.Л. Принятие решений при зависимостях и обратных связях: Аналитические сети / Под ред. А.В. Андрейчикова, О.Н. Андрейчиковой. – М.: URSS, 2019. – 360 с.
27. Аверкин А.Н. и др. Нечеткие множества в моделях управления и искусственного интеллекта / Под ред. Д.А. Поспелова. – М.: Наука, 1986. – 312 с.
28. Zadeh L.A. Fuzzy sets // Information and Control. – 1965. – Vol. 8. – № 3. – P. 338-353.
29. Aliev R.A. Fundamentals of the Fuzzy Logic-Based Generalized Theory of Decisions. – Berlin, Heidelberg: Springer-Verlag, 2013. – 340 p.
30. Zadeh L.A., Kacprzyk J. Fuzzy Logic for the Management of Uncertainty. – New York: John Wiley & Sons, 1992. – 676 p.
31. Figueira J., Mousseau V., Roy B. ELECTRE Methods // Multiple Criteria Decision Analysis: State of the Art Surveys / Ed. by J. Figueira, S. Greco, M. Ehrgott. – International Series in Operations Research & Management Science. – Boston: Springer, 2005. – P. 609-637.
32. Ройзензон Г.В. Синергетический эффект в принятии решений // Системные исследования. Методологические проблемы. Ежегодник / Под ред. Ю.С. Попкова, В.Н. Садовского, В.И. Тищенко. 2011-2012. – М.: УРСС, 2012. – № 36. – С. 248-272.
33. Ларичев О.И. Вербальный анализ решений. – М.: Наука, 2006. – 181 с.
34. Асанов А.А. и др. Метод многокритериальной классификации ЦИКЛ и его применение для анализа кредитного риска // Экономика и математические методы. – 2001. – Т. 37. – № 2. – С. 14-21.

ОНТОЛОГИЯ ПРОЕКТИРОВАНИЯ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ

*Евгений Г.Б., д.т.н., профессор
МГТУ им. Н. Э. Баумана
e-mail: g.evgenov@mail.ru*

1. МОДЕЛИ ТЕХНОЛОГИЧЕСКИХ ОБЪЕКТОВ

Проектирование технологических процессов осуществляется методами групповой технологии [1, 2, 3, 4, 5], а также проектированием групповых и единичных техпроцессов [6, 7, 8].

Применительно к построению технологической диаграммы классов центральной является группа описания технологических объектов. Остальные группы понятий необходимы для задания свойств этих объектов.

Диаграмма классов технологических объектов представлена на рис. 1. Корневым объектом в этой диаграмме является «Технологический процесс», представляющий собой часть производственного процесса, содержащую целенаправленные действия по изменению и (или) определению состояния предмета труда. Технологический процесс – это родовое понятие, которое имеет разновидности, соответствующие различным технологическим методам. На диаграмме отображены основные методы: формообразование, обработка давлением, обработка, сборка и получение покрытия. В число прочих методов входят: технический контроль, испытание, перемещение, консервация, расконсервация и упаковывание.

Технический процесс по своей структуре состоит из операций, которые принадлежат к различным методам. Например, операция литье относится к методу формообразования, штамповка – к обработке давлением, токарная – к обработке, слесарно-сборочная – к сборке, а хромирование – к получению покрытия.

Операция состоит из установов, каждый из которых выполняется при неизменном закреплении обрабатываемых заготовок или собираемой сборочной единицы. На каждом установе может выполняться один или несколько основных и вспомогательных

переходов. Наконец, переход может состоять из одного или нескольких рабочих и вспомогательных ходов.

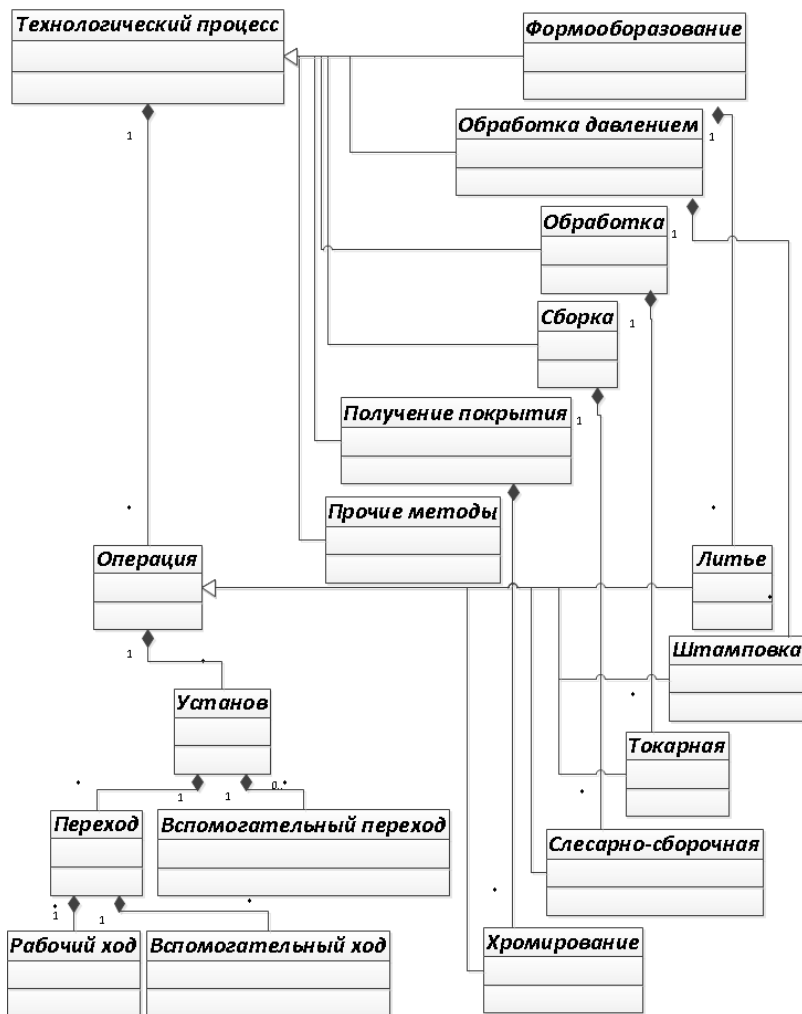


Рис. 1. Диаграмма классов технологических объектов

Классы основных технологических методов, представленные на рис. 1, делятся на подклассы. Формообразование включает подклассы: литье, формование и спекание. Обработка давлением состоит из подклассовковки, штамповки и поверхностного пластического

деформирования. В состав класса обработки входят следующие виды обработки: резанием, термической, электрофизической, электрохимической и слесарной обработок, а также раскроя материала. Сборка включает сборку разъемных соединений, сварку, пайку, клепку, склеивание и монтаж. Получение покрытий делится на получение покрытий неорганических и органических.

В состав прочих технологических методов входят технический контроль, испытание, перемещение, консервация, расконсервация и упаковывание.

Родовидовая декомпозиция технологических операций задана стандартом ЕСТД, который определяют коды и наименования операций, подлежащие обязательному использованию в технологической документации. В табл. 1 приведены группы операций.

Таблица 1. Группы операций

| Код группы | Наименование группы операций |
|------------|--|
| 4150 | Зубообрабатывающие операции |
| 0600 | Испытания |
| 1000 | Операции литья металлов и сплавов |
| 7100 | Операции нанесения покрытий |
| 2100 | Операции обработки давлением |
| 0100 | Операции общего назначения |
| 9000 | Операции пайки и сварки |
| 0400 | Операции перемещения и консервации |
| 4230 | Операции с ЧПУ |
| 5000 | Операции термообработки |
| 4190 | Отделочные операции |
| 4280 | Отрезные операции |
| 6500 | Порошковая металлургия |
| 4100 | Различные операции резанием |
| 8800 | Сборочные операции |
| 4210 | Сверлильно-расточные операции |
| 4170 | Строгальные протяжные и долбежные опе |
| 0200 | Технический контроль |
| 4110 | Токарные операции |
| 6000 | Формообразование из полимерных матери |
| 5500 | Фотохимико-физическая обработка |
| 4260 | Фрезерные операции |
| 4130 | Шлифовально-заточные операции |
| 8500 | Электромонтажные операции |
| 7500 | Электрохимические и электророзфизическ |

Группы включают родственные технологические операции. Например, перечень операций с ЧПУ приведен в табл.2.

Таблица 2. Операции с ЧПУ

| Код операции | Наименование операции |
|--------------|-----------------------|
| 4231 | Расточная с ЧПУ |
| 4232 | Сверлильная с ЧПУ |
| 4233 | Токарная с ЧПУ |
| 4234 | Фрезерная с ЧПУ |
| 4236 | Шлифовальная с ЧПУ |
| 4237 | Комплексная с ЧПУ |

Объектная модель технологических элементов зависит от используемого варианта интеграции конструкторского и технологического проектирования.

При использовании систем интегрированных на уровне знаний проектирование технологии осуществляется методами соответствующих конструкторских объектов с занесением результатов в базу данных.

В системах, интегрированных на уровне данных и автономных системах проектирования ТП, применяются активные технологические объекты, диаграмма классов которых со свойствами и методами представлена на рис. 2. Для ввода данных изделия эти объекты используют соответствующие интерфейсы. При интеграции с конструкторской системой на уровне данных интерфейс обеспечивает считывание объектных моделей, сформированных этой системой. В случае автономной системы проектирования ТП интерфейс организует диалог с пользователем, с помощью которого технолог описывает конструктивные данные изделия.

Вывод данных спроектированных технологических процессов в фактографической форме (в форме базы данных) и в документографической форме (в форме технологических документов) производится интерфейсным модулем (рис. 2).

Набор свойств базовых технологических объектов (рис. 2) определяется набором параметров, необходимых для заполнения соответствующей технологической документации. Для объекта «Технологический процесс обработки» эти свойства включают обозначение ТП, также данные материала детали и данные заготовки. Метод этого объекта обеспечивает проектирование и нормирование ТП.

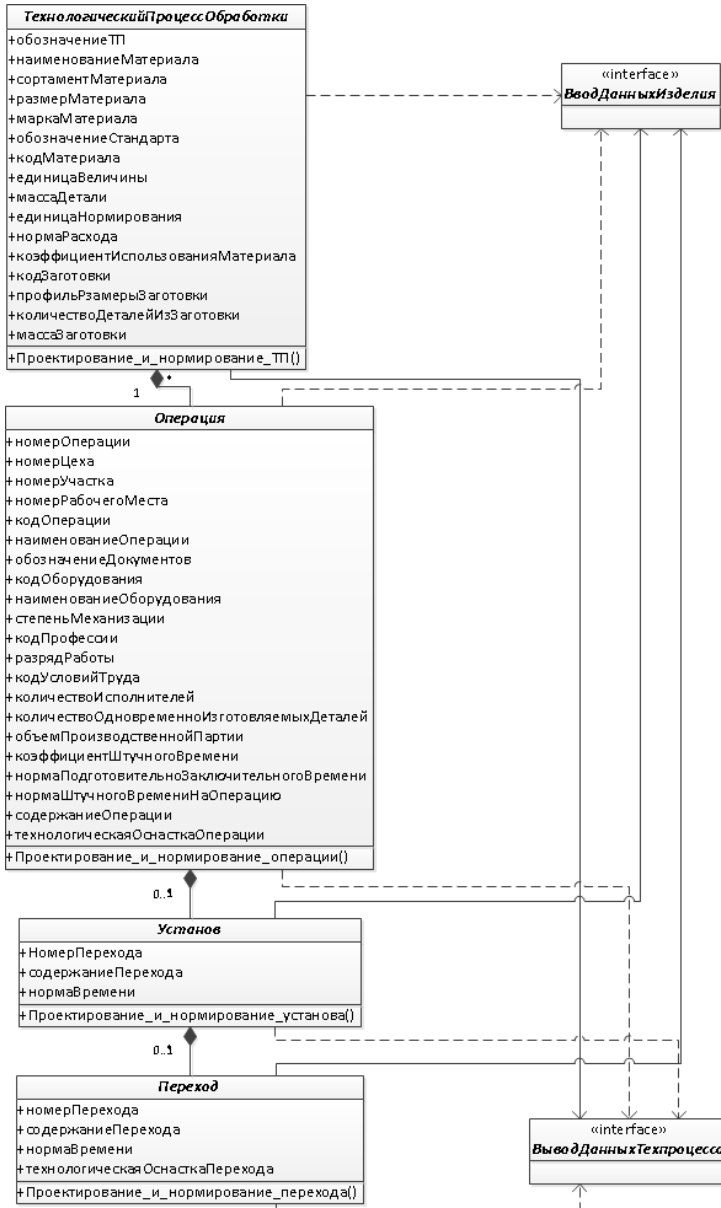


Рис. 2. Диаграмма классов базовых технологических объектов

При формировании технологической документации принято заносить данные установка как одного из переходов операции. Поэтому он нумеруется в сквозном порядке с остальными переходами. Однако его содержание и нормирование отличается от остальных переходов, что учитывается в его методе.

Свойства перехода аналогичны свойствам установка, но дополнительно включают данные используемой на этом переходе оснастки. Метод этого объекта зависит от специфики выполняемой обработки или сборки. Однако его содержание и нормирование отличается от остальных переходов, что учитывается в его методе.

Свойства перехода аналогичны свойствам установка, но дополнительно включают данные используемой на этом переходе оснастки. Метод этого объекта зависит от специфики выполняемой обработки или сборки.

В некоторых случаях (рис. 3) операция может не иметь деления на установки и переходы.

| <i>Переход Механической Обработки</i> | |
|---------------------------------------|--|
| + | номер Перехода |
| + | наименование Перехода |
| + | наименование Обрабатываемого Элемента |
| + | расчетный Размер Диаметра-Ширины_мм |
| + | отклонение Размера Верхнее_мм |
| + | отклонение Размера Нижнее_мм |
| + | расчетный Размер Длины_мм |
| + | глубина Резания_мм |
| + | число Проходов |
| + | подача_мм/об |
| + | число Оборотов Шпинделя_в_мин |
| + | скорость Резания_м/мин |
| + | основное Время На Переход_мин |
| + | вспомогательное Время На Переход_мин |
| + | технологическая Оснастка Перехода |
| + | Проектирование_и_нормирование_п_рехода() |

Рис. 3. Класс объектов «Переход механической обработки»

На рис. 4 представлена структура класса переходов без привязки к методам обработки. Для некоторых методов используются специфические классы операций и переходов. На рис. 4 приведен пример класса переходов механической обработки. Здесь содержание перехода разложено на две компоненты: наименование перехода и наименование обрабатываемого элемента. Из этих двух компонент формируется текст, заносимый в документацию. Приведены

размерные характеристики этого элемента, включая допустимые отклонения для исполняемого размера. Заданы параметры режимов обработки, включая глубину резания, число проходов, подачу и число оборотов шпинделя, скорость резания, а также нормы основного и вспомогательного времени.

Диаграмма классов технологических процессов инвариантна и не зависит от изготавливаемых изделий (рис. 4). Но в результате проектирования получается для каждого изделия множество взаимосвязанных по времени экземпляров классов технологических объектов (например, операций и переходов), формирующих в своей совокупности технологический процесс.

Методы, осуществляющие проектирование и нормирование технологических процессов, операций и переходов, свои результаты записывают в базу данных. Пример концептуальной модели такой реляционной базы данных описывается ниже. Структура этой модели подобна диаграмме классов технологических процессов (рис. 4). Классам объектов соответствуют сущности (реляционные отношения), которые могут содержать множество записей, описывающих экземпляры класса. Каждый экземпляр имеет уникальное имя, составляющее ключ сущности. Имя технологического процесса определяют обозначения изделия и ТП (рис. 4). Это связано с тем, что каждое изделие может иметь несколько вариантов технологических процессов. Поскольку сущности «Технологический процесс», «Операция», «Установ» и «Переход» взаимосвязаны как целое-часть, то происходит миграция ключей от целого к части. Составной ключ операции включает обозначения изделия и ТП, мигрировавшие из технологического процесса (FK) и идентифицирующие принадлежность операции к технологическому процессу. Помимо этого операция имеет собственный ключ – номер операции, позволяющий сформировать уникальное имя операции.

Аналогичным образом формируются ключи установов и переходов. При одинаковой структуре ключей уникальность имен установов и переходов определяется их сквозной нумерацией в пределах операции, исключающей совпадение номеров переходов и установов.

В концептуальной модели на рис.4 имеется родовидовое отношение, связывающее общую сущность «Переход» с сущностями-категориями «Переход общий» и «Переход механообработки». При родовидовых отношениях ключ общей сущности мигрирует полностью в сущности-категории. Состав свойств сущностей-категорий различен.



Рис 4 . Концептуальная модель данных ТП механической обработки

2. МОДЕЛИ ПРОЕКТИРОВАНИЯ ПРОЦЕССОВ СБОРКИ

Системологические модели обработки строятся на основе графов. Обобщением понятия графа является понятие *гиперграфа*, который удобен для построения моделей сборки. В гиперграфе имеется множество вершин и множество ребер, называемых иногда множеством букв и слов соответственно. Основное отличие гиперграфа от графа состоит в том, что число вершин, инцидентных ребру, не ограничено двумя. *Гиперграф* определен, если указан его носитель и заданы все кортежи его сигнатуры. *Кортеж* полностью определяется множеством элементов носителя, входящих в него. Поэтому гиперграф можно рассматривать как совокупность множества с заданным семейством подмножеств M этого множества $B = \langle E, M \rangle$.

Каждый гиперграф может быть однозначно задан с помощью матрицы инцидентности, которая представляет собой матрицу $C = \| c_{ij} \|$ размерности $m \times n$, где $n = |E|$, $m = |M|$ и M множество слов, причем

$$c_{ij} = \begin{cases} 1, & \text{если } x_j \in M_i \\ 0, & \text{если } x_j \notin M_i \end{cases}$$

Последовательность сборки в основном определяется конструкцией изделия и может быть представлена в виде технологической схемы сборки, являющейся условным изображением порядка комплектования изделия и узлов при сборке. Схемы сборки позволяют наглядно представить весь технологический процесс. На этих схемах каждый элемент изделия представляется прямоугольником, в котором указываются наименование составной части, ее индекс и количество (рис. 5).

Деталь или ранее собранная сборочная единица, с которой начинают сборку, присоединяя к ней другие детали и сборочные единицы, называют *базовой деталью* или *базовой сборочной единицей*.

На рис. 6 базовыми деталями являются: для сборочной единицы СЕ 3 – деталь Д15, для СЕ 2 – Д10, для СЕ 1 – Д6, а для изделия – Д1. Процесс сборки на схеме изображается горизонтальной линией в направлении от прямоугольника с изображением базовой составной части до прямоугольника, изображающего готовое изделие или сборочную единицу. Выше горизонтальной линии показываются в порядке последовательности сборки прямоугольники, условно обозначающие детали, а ниже – прямоугольники, условно изображающие сборочные единицы. Например, для изделия на рис.6. к базовой детали Д1 сначала присоединяют деталь Д2, затем сборочную

единицу СЕ 1, потом детали Д3 и Д4, далее сборочную единицу СЕ 2 и наконец деталь Д5.

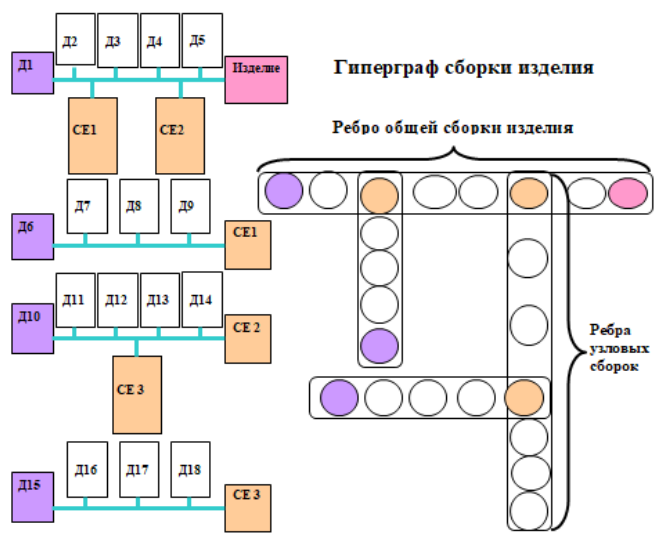


Рис. 5. Гиперграфовые модели технологических схем сборки

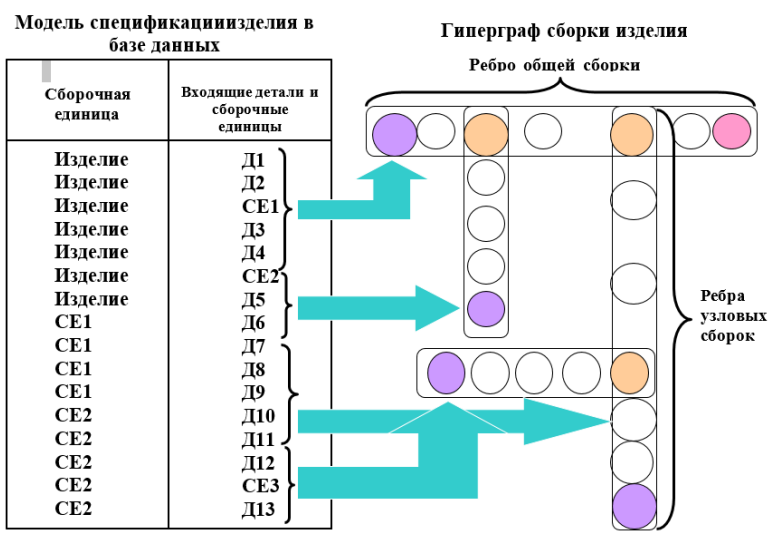


Рис. 6. Гиперграфовые модели технологических схем сборки

Деталь или ранее собранная сборочная единица, с которой начинают сборку, присоединяя к ней другие детали и сборочные единицы, называют *базовой деталью* или *базовой сборочной единицей*.

3. ЗАКЛЮЧЕНИЕ

Онтология технологического проектирования представляет собой фундамент для цифровизации производства. На ее основе строятся компьютерные системы автоматизации проектирования и программирования технологических процессов, а также оперативного управления производством.

Литература

1. Групповая технология изготовления заготовок серийного производства. – Л., 1985. – 240 с.
2. Научные основы организации группового производства. – М. –Л., 1963. – 308 с
3. Митрофанов С. П., Гульнов Ю. А., Куликов Д. Д. Автоматизация технологической подготовки серийного производства. – М.: «Машиностроение», 1974. – 360 с.
4. Митрофанов С. П., Гульнов Ю. А., Куликов Д. Д., Падун Б. С. Применение ЭВМ в технологической подготовке производства. – М.: «Машиностроение», 1981. – 286 с.
5. Митрофанов С. П., Куликов Д. Д., Миляев О. Н., Падун Б. С. Технологическая подготовка гибких производственных систем. – Л.: «Машиностроение», 1987. – 352 с.
6. Евгеньев Г.Б. Системология инженерных знаний. Москва, Изд-во МГТУ им. Н.Э. Баумана, 2001. – 376 с.
7. Основы автоматизации технологических процессов и производств. В 2-х. Т.1/ Под ред. Проф. Г.Б. Евгенева – Информационные модели: Учебное пособие для вузов. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2012. – 441 с.
8. Основы автоматизации технологических процессов и производств. В 2-х. Т.2/ Под ред. Проф. Г.Б. Евгенева – Методы проектирования и управления: Учебное пособие для вузов. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2012. – 479 с.

ИССЛЕДОВАНИЕ ПОДХОДОВ СЖАТИЯ ДАННЫХ ПРИ ПОСТРОЕНИИ ЦИФРОВЫХ ДВОЙНИКОВ

Скалкин А.М., аспирант

e-mail: anton.skalkin73@gmail.com

Романов А.А., к.т.н., доцент

e-mail: romanov73@gmail.com

Ульяновский государственный технический университет

1. ВВЕДЕНИЕ

В настоящее время происходит активное накопление данных в различных информационных системах, увеличивается количество устройств, которые генерируют новые данные (IoT, носимые устройства, автоматизированные системы и так далее). Для того чтобы решать задачи управления эффективно требуется использовать гетерогенные массивы данных. Следствием этого является рост временных и вычислительных затрат. Вертикальное и горизонтальное масштабирование вычислительных систем может дать положительный эффект в сокращении таких затрат, однако приведет увеличению стоимости вычислительных мощностей.

В таком случае необходимо решить задачу сокращения объёма анализируемых данных при сохранении их основных свойств важных для задач управления.

В качестве моделей для хранения временных рядов возможно использование цифровых двойников, как инструмента моделирования и манипуляций над реальными объектами для проведения анализа и прогнозирования. Цифровые двойники могут включать данные в виде временных рядов, которые необходимо обрабатывать, чтобы избежать излишнего разрастания данных.

Для этого могут использоваться различные методы сжатия, сглаживания, а также нечеткие методы – F-преобразование. Большое количество методов существует, которые можно использовать для решения поставленной задачи, но для выбора метода и параметров его работы необходимо отталкиваясь от целей анализа сжимаемых данных произвести оценку качества производимого сжатия.

2. ЦИФРОВЫЕ ДВОЙНИКИ

Под цифровым двойником подразумевается синхронизированная виртуальная модель любых объектов, систем, людей, процессов и сред [1]. Также данная модель может представлять абстрактные объекты, такие как процессы, и отражать в себе текущее состояние моделируемого объекта. Также особенной чертой цифровых двойников является предсказательная способность будущих состояний сущности на основе текущих данных.

Существует большое количество методов создания цифровых двойников [2], которые могут уточняться и перестраиваться в соответствии с данными наблюдений над объектом моделирования.

В текущем случае концепция цифровых двойников рассматривается для представления процессов, изменяющиеся данные которых представляют собой временной ряд. Способность синхронизации отличает цифровых двойников от статических моделей и позволяет поддерживать актуальным временной ряд, содержащийся в модели. Хранение данных в данном виде подразумевает анализ для достижения целей создания цифрового двойника. Кроме того, стоит обратить внимание на особенности временных рядов, создаваемых датчиками и метриками: избыточность, аппроксимируемость, предсказуемость [3].

В зависимости от различных факторов, включающих поставленную задачу анализа, предметную область, частоту получения новых данных и т.д., временной ряд может содержать огромное количество элементов. Это будет требовать больших ресурсозатрат при обработке и хранении этих данных.

3. МЕТОДЫ СЖАТИЯ ВРЕМЕННЫХ РЯДОВ

Для решения поставленной проблемы существуют алгоритмы сжатия и сглаживания временных рядов. Алгоритмы сжатия могут быть описаны с помощью противопоставления различных классов [3].

Неадаптивный или адаптивный: неадаптивный алгоритм ориентирован на сжатие определенного типа или типов данных, в то время как адаптивный алгоритм заранее не настраивается на определенный вид данных;

С потерями или без потерь: алгоритмы могут допускать потери, если результат декодирования не идентичен исходным данным, или работать без потерь, если результат декодирования идентичен исходным данным.

Симметричный – несимметричный: симметричный алгоритм использует один алгоритм при кодировании и декодировании,

отличаясь работой в разных направлениях, тогда как несимметричный использует два разных алгоритма.

Также алгоритмы сжатия можно разделить на блочные и поточные.

Большое число типов сжатия порождает еще большее количество алгоритмов сжатия. Например, в сжатии с потерями хорошо себя зарекомендовали метод выделения структур, фрактальные методы, дискретное преобразование Фурье, дискретное косинусное преобразование. Методы сжатия без потерь включают метод Хаффмана, арифметический метод, преобразующие оптимизации.

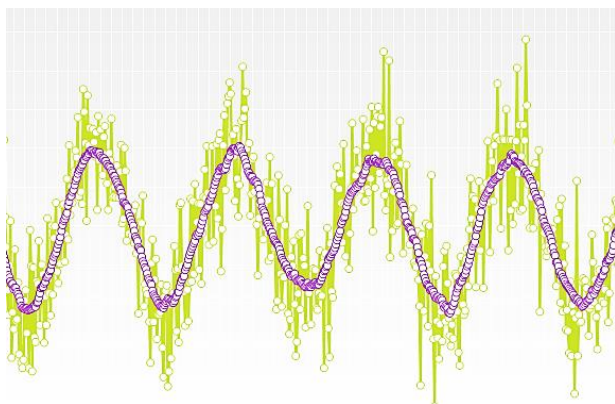


Рис.1. Пример временного ряда и его сглаженное представление

Основными методами сглаживания временных рядов являются скользящее среднее и экспоненциальное сглаживание. В свою очередь экспоненциальное сглаживание делится на 3 варианта:

1. одинарное сглаживание для рядов, у которых нет тренда и сезонности;
2. двойное сглаживание для рядов, у которых есть тренд, но нет сезонности;
3. тройное сглаживание для рядов, у которых есть и тренд, и сезонность.

Также для сжатия можно использовать методы, основанные на нечеткой логике, одним из которых является F-преобразование [4]. Использование F-преобразования для сглаживания временных рядов имеет преимущества перед другими методами сглаживания, такими как экспоненциальное сглаживание, из-за возможности включения информации о знаниях о временных рядах [5].

F-преобразование устанавливает соответствие между множеством непрерывных функций на интервале вещественных чисел и множеством n -мерных (вещественных) векторов. Каждый компонент результирующего вектора является взвешенным локальным средним соответствующей функции над областью, покрытой соответствующей базовой функцией.

Высокая эффективность сжатия данным методом подтверждается в большом количестве статей. Особенности данного метода также является возможность производить сжатие различных представлений данных [6]. Большая часть исследований описывает использование данного подхода для сжатия изображений, позволяющим добиться большего качества для восстановленного изображения [7].

4. ОЦЕНКА КАЧЕСТВА СЖАТИЯ

При анализе временных рядов нас интересует выявление их структуры и оценка основных компонентов – тренда, сезонности, шума и других особенностей, а также возможность строить прогнозы изменения величины в будущих периодах.

В большинстве случаев гораздо важнее получить более точные показатели, чем экономить ресурсы, но окончательный выбор следует принять с учетом предметной области и поставленной задачи анализа сжимаемых данных. Используя различные методы с различными коэффициентами работы можно получить разные результаты и трудно выделить один наилучший результат. Для дальнейшего использования конкретизируем некоторые цели анализа временных рядов: прогнозирование, поиск аномалий, восстановление пропущенных значений.

Каждая из поставленных целей анализа по-разному соотносится со сглаживанием. Так при прогнозировании и восстановлении пропущенных значений излишнее сглаживание может скрыть сезонность, что отрицательно скажется на качестве прогноза, а при поиске аномалий – сгладить их или критично уменьшить их значение.

Для оценки эффективности производимых изменений можно использовать метрики MAPE, MPE, MSE, а также коэффициент несовпадения Тейлора. Основной целью использования данных метрик является оценка качества работы моделей прогнозирования временных рядов. Хотя их использование с целью проверки качества сжатия будет неправильным, так как наилучшие результаты будут при минимальном сглаживании.

Для оценки качества сжатия с учетом различных целей использования временных рядов необходимо использовать новый метод оценки, представленный в формуле:

$$M_C(F_A(TS), F_A(TS')) \in [0,1],$$

где TS – исходный временной ряд, TS' – сжатый временной ряд, $F_A()$ – функция целевого алгоритма, а M_C (compression metric) – метрика качества сжатия.

Сжатый временной ряд представляет собой ряд полученный из исходного после обработки методом сжатия/сглаживания для уменьшения количества данных, содержащихся в ряду:

$$TS' = F_C(TS), |TS'| < |TS|,$$

где TS – исходный временной ряд, TS' – сжатый временной ряд, а F_C – функция сжатия.

Для использования данной метрики планируется её выполнение не зависимо от используемого метода сжатия или сглаживания. Входными данными являются изначальный временной ряд, тип анализа временного ряда и метод сжатия, планируемый для использования. В общем виде задача подразумевает использование множества комбинаций параметров функции сжатия F_C для выявления наилучших параметров, подходящих к выбранной задаче F_A .

При первом итерации разработки подхода оценки хотелось бы обобщить процесс оценки для различных функций анализа F_A . Подразумевается, что разрабатываемая метрика M_C должна позволить подбирать показатели для работы алгоритмов сжатия при отсутствии снижения качества анализа. Так, при решении задачи прогнозирования временного ряда сжатого/сглаженного выбранным способом F_A необходимо выполнять сравнение тестового фрагмента исходного временного ряда TS и прогноз, построенный на основе измененного временного ряда TS' . Саму схожесть предлагается определять с использованием выше представленных метрик, например MAPE. В результате всех комбинаций необходимо выбрать наиболее соответствующую следующим условиям:

$$M_C \rightarrow \max, |TS'| \rightarrow \min,$$

где $M_C \rightarrow \max$ указывает на стремлении качества к собственному максимуму, то есть к 1, при уменьшении мощности временного ряда $|TS'| \rightarrow \min$.

Также при построении данной модели M_C можно использовать кросс-валидацию на скользящем окне для повышения качества создаваемой модели.

5. ЗАКЛЮЧЕНИЕ

В результате проделанной работы был произведен анализ подходов сжатия временных рядов на примере моделей цифровых двойников. Для оценки многообразия методов сжатия был предложен новый подход по оценке качества производимого сжатия.

Использование данной методики позволит: абстрагироваться от типа проводимой аналитики и тем более методов, которые будут для них использоваться; абстрагироваться от метода сжатия/сглаживания; автоматизировано получить наилучшие параметры модели сжатия для будущего использования; получить результаты анализа на том же уровне при наибольшем сжатии данных.

Литература

1. Елин В. М. Технология цифрового двойника. Понятие и особенности подхода к организационно-правовому обеспечению комплексной безопасности // Вестник ЮУрГУ. Серия: Право. 2020. №3. С. 68-75
2. Васильев А. Н., Тархов Д. А., Малыхина Г. Ф. Методы создания цифровых двойников на основе нейросетевого моделирования // Современные информационные технологии и ИТ-образование. 2018. №3. С. 521-532.
3. Chiarot G., Silvestri C. Time Series Compression: A Survey //arXiv preprint arXiv:2101.08784. 2021.
4. Novák V. et al. Filtering Out High Frequencies In Time Series Using F-transform //Information Sciences. – 2014. – Т. 274. – С. 192-209.
5. Romanov A., Filippov A., Yarushkina N. Extraction And Forecasting Time Series Of Production Processes //International Conference on Information Technologies. – Springer, Cham, 2019. – С. 173-184.
6. Perfilieva I., Novák V., Dvořák A. Fuzzy Transform In The Analysis Of Data //International Journal of Approximate Reasoning. – 2008. – Т. 48. – №. 1. – С. 36-46.
7. Hurtik P., Perfilieva I. A Hybrid Image Compression Algorithm Based On Jpeg And Fuzzy Transform //2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). – IEEE, 2017. – С. 1-6.

ABSTRACTS

Section 4

DEVELOPMENT OF A METHOD FOR CREATING PATIENT-ORIENTED RECOMMENDATIONS FOR THE MANAGEMENT OF CARDIOVASCULAR HEALTH IN MEN

*Tatiana V. Afanasieva, PhD
Plekhanov Russian University of Economics
e-mail: afanaseva.tv@rea.ru*

Currently, the development of methods and tools for automating the management of the risk of cardiovascular (CV) events in men is a strategic task for Russian healthcare and a key task for preventive medicine. For the prevention, early detection and management of CV health, a method is proposed that allows automation. Oriented recommendations included descriptive, predictive and prescriptive components. The process of creating patient-oriented recommendations is described. The application of the proposed method will provide an opportunity to optimize the costs of healthcare and improve its quality.

Keywords: management, cardiovascular health, patient-oriented recommendations, model, method.

CONSTRUCTION OF GRANULAR ONTOLOGIES FOR COGNITIVE AGENTS

*Vadim V. Borisov, Dr. Sci. (Tech.), Professor
Branch of the National Research University "MPEI" in Smolensk
e-mail: vbor67@mail.ru*

Smolensk, Russia

A.E. Misnik, PhD

*Inter-state educational institution of higher education "Belarusian-Russian
university"*

Mogilev, Republic of Belarus

anton@misnik.by

Modern educational establishments can be treated as cyber-physical systems with the following features:

- the complexity of the structure, multicomponent, the presence of functional subsystems that solve various target tasks, a large number of parameters that characterize the processes of the system;
- dynamic change in the structure and parameters of the system;
- incomplete information about the functioning and state of the system;
- variety of external and anthropogenic impacts on the system;
- the presence of complex nonlinear relationships between parameters;
- the complexity of making managerial decisions to improve the efficiency of the system;
- limited opportunities for experimental studies of the system and ongoing processes;
- impossibility of using a unified approach to the creation of models of system and information-analytical processes occurring in such systems.

Since 2006, at the Belarusian-Russian University, we have been developing process automation systems within the university.

This year we are launching the Electronic University system, built on the principles of ontological engineering, which, will allow us to integrate the educational process and all university services into a single system.

The conducted studies made it possible to substantiate the use of metagraphs for the corresponding class of systems, taking into account the selected tools that combine the ontological and metagraph representations within a single tool, which is not achieved by the traditional integration of ontologies.

Keywords: education, cyber-physical systems, metagraphs, ontology, temporal nero-fuzzy Petri nets.

THE MAIN PROVISIONS OF THE MULTIDIMENSIONAL METAGRAPH MODEL OF DATA AND KNOWLEDGE

Yuriy E. Gapanyuk

Bauman Moscow State Technical University

e-mail: gapyu@bmstu.ru

The article discusses the basic concepts of complex graphs, hypergraph and hypernetwork models, the main provisions of the annotated metagraph model, the basic principles of the multidimensional metagraph model and its connection with metagraph processes. A multidimensional metagraph model can process arbitrary data, knowledge and processes represented in the form of a metagraph using aggregation between model levels or using metagraph processes.

Keywords: complex network, complex graph, emergence, metagraph, metavertex, metaedge, metagraph process.

ONTOLOGY MODEL FOR DECISION SUPPORT BASED ON FUZZY LOGICAL INFERENCE

Gleb Y. Guskov, PhD

e-mail: guskovgleb@gmail.com

Anton A. Romanov, PhD

e-mail: romanov73@gmail.com

Anton M. Skalkin, P.G.

e-mail: anton.skalkin73@gmail.com

Julia V. Stroeva, P.G.

e-mail: stroeva95@mail.ru

Aleksey A. Filippov, PhD

e-mail: al.filippov@ulstu.ru

Ulyanovsk State Technical University

The central part of DSS is an inference system that allows inferring results based on information about an analyzed object within a context.

This paper presents: novel ontology models for decision support and fuzzy inference, the fuzzy inference algorithm, and the example of the basic DSS for balancing production capacities.

Keywords: fuzzy inference, ontology, decision support system.

HERITABILITY OF EMPIRICAL DEPENDENCIES AND EXPLAINABILITY OF INTELLIGENT DATA ANALYSIS RESULTS

*Michael I. Zabezhailo, D.Sc.
FRC "Computer Science and Control", Russian Academy of Sciences
e-mail: m.zabezhailo@yandex.ru*

The problem of trust in results of the Intelligent Data Analysis (IDA) is considered with respect to sequence of the analyzed data extensions and varying the languages of data\ knowledge representation. To form explanations of the calculated IDA results, it is proposed to use empirical dependencies of a cause-effect type inferred from the analyzed data. There is discussed the problem of identifying such causal empirical dependencies that would be inherited (remain stable) when expanding the initial empirical data with descriptions of new precedents to be analyzed and varying the Knowledge Representation Language used for their description

Keywords: AI, intelligent data analysis (IDA), empirical dependencies (EDs) in data, heritability of EDs with respect to data extensions, cause-effect EDs in data, explanation of IDA results.

ON THE ARGUMENTATIVE APPROACH TO OPINIONS ANALYSIS

*Victor K. Finn, D.Sc., Professor
e-mail: ira.finn@gmail.com
Maria A. Mikheyenkova, D.Sc.
e-mail: m.mikheyenkova@yandex.ru
FRC "Computer Science and Control", RAS; RSUH*

The paper considers argumentation as a means of rational acceptance of statements. Variants of argumentative semantics for four-valued logics are presented, rationality criteria are proposed taking into account objective knowledge, their dependence on the chosen logical semantics is shown.

Keywords: cognitive sociology, argumentation, logic of argumentation, four-valued logics, opinions rationality.

Section 5

INTEGRATION OF COGNITIVE GRAPHICS AND FUZZY LOGIC TOOLS FOR THE PRESENTATION AND DIAGNOSIS OF COMPLEX VISUAL PATHOLOGY

*Aleksandr P. Ereemeev, Professor
National Research University "Moscow Power Engineering Institute"
e-mail: eremeev@appmat.ru*

*Irina V. Tsapenko, Chief specialist
Helmholtz National Medical Research Center of Eye Diseases
e-mail: sunvision@mail.ru*

This paper discusses the possibilities of integrating methods and tools of cognitive graphics (cognitive images) and fuzzy logic for imaginative representation of complex visual pathologies (retinal pathologies) to help ophthalmologists in the analysis and diagnosis of eye diseases using an approach based on electroretinography (electroretinograms, ERG), which consists in the study of biopotentials arising on the retina of the eye when a certain light stimulation.

Keywords: intelligent system, fuzzy logic, cognitive graphics, visual pathology.

SEMIOTIC COGNITIVE MAPS

*Kulinich A.A., Ph.D., senior researcher
V.A. Trapeznikov ICS RAS
e-mail: alexkul@rambler.ru*

The paper proposes semiotic cognitive maps aimed at supporting the interpretation of the cognitive modeling results. A mathematical model of a semiotic cognitive map is proposed, which consists of three models connected by parameters: syntactic, semantic and pragmatic. A method for interpreting solutions of the inverse problem, which is based on searching for the solution classes' names on the Internet, is proposed. For this, methods for extracting and processing unstructured Internet information are proposed. These are methods of lexico-syntactic templates for highlighting generic relations in the text corpus and distributive analysis for determining frequently used names of decision classes. Experiments have shown the efficiency of the proposed methods and approaches. The application of the proposed semiotic cognitive map model will expand the capabilities of decision support systems in dynamic ill-defined situations.

Keywords: cognitive maps, applied semiotics, natural language processing, interpretation, decision support.

COGNITIVE BASIS OF ELLIPSIS CLASSIFICATION

Naidenova X.A. PhD, s.s.r.
e-mail: ksennaidd@gmail.com,
Martirova T.A., s. r.
e-mail: martta462@yandex.ru
Military medical academy

In this paper, a classification of ellipses is given based on the cognitive approach. Some rules of cognitive transformations of complete sentences into their elliptical variants are formulated for Verb and Noun Phrase ellipses and illustrated by many examples from planimetry and fiction. An approach to resolving ellipses is given. Three types of equivalent relations between constituents of sentences are introduced: weak syntactic equivalence, strong syntactic equivalence and semantic equivalence. These equivalences are necessary used in the process of resolving ellipses. The importance of the cognitive direction to restoring ellipses in the light of the study of the interaction of cognition and syntax in human thinking is shown.

Keywords: cognitive approach, ellipsis, natural language processing.

HOW AN AUTONOMOUS COMPUTER AGENT CAN DISCOVER THE LAWS OF NATURE

Vladimir G. Red'ko, PhD
Scientific Research Institute for System Analysis,
Russian Academy of Sciences (SRISA RAS)
e-mail: vgreddko@gmail.com

The possibilities of constructing of models of autonomous computer cognitive agents, which are able to discover the laws of nature, have been analyzed. The prerequisites and approaches to building of models of such autonomous cognitive agents are analyzed. The initial biologically inspired models of cognitive agents cognizing the external world are characterized. The generalized characteristics of cognitive agents in biologically inspired models are given.

Keywords: cognitive agents, cognition of the laws of nature.

THE MODEL OF COGNIZING REGULARITIES BY AN AUTONOMOUS AGENT

*Vladimir G. Red'ko, PhD,
e-mail: vcredko@gmail.com*

*Zarema B. Sokhova, PhD
e-mail: zarema.sokhova@gmail.com*

*Scientific Research Institute for System Analysis,
Russian Academy of Sciences (SRISA RAS)*

The article considers a population of cognitive autonomous agents that learn and use the regularities of the external world. The regularities are determined by the dependence of the illumination of the environment on time. The ability to recognize such regularities is optimized by agents both through learning and in the course of the evolution of a population of learning agents. A computer model of cognition of regularities by an autonomous agent has been constructed and investigated. The interaction between learning and evolution in the developed model has been analyzed. The model can be viewed as a stage in the study of more general properties of autonomous cognitive agents.

Keywords: autonomous agents, learning, evolution, interaction between learning and evolution.

Section 6

RESEARCH IN THE DEVELOPMENT OF EXPLANATORY ARTIFICIAL INTELLIGENCE SYSTEMS

Alexey N. Averkin, Ph.D.

Dorodnicyn Computing Centre, FRC CSC RAS

e-mail: averkin2003@inbox.ru

Sergey A. Yarushev, Ph.D.

Department of Informatics

Plekhanov Russian University of Economic

e-mail: Sergey.Yarushev@icloud.com

This paper explores the taxonomy and several directions in the study of explainable neural networks related to the extraction of rules from neural networks, which allow the user to get an idea of how the neural network uses the input data, as well as using the rules to reveal hidden relationships between the input data and the results found. Thus, this review focuses on the connection between the most common rule-based explanation systems in artificial intelligence and the most powerful machine learning algorithms using neural networks.

Keywords: XAI, artificial intelligence, neural networks, rules extraction, fuzzy logic.

THE CONCEPT OF A HYBRID INTELLIGENT DECISION SUPPORT SYSTEM FOR LASER-BASED ADDITIVE MANUFACTURING

Valeria V. Gribova, Dr.Sc. (Tech.)

e-mail: gribova@iacp.dvo.ru

Yuriy N. Kulchin, Academician

e-mail: kulchin@iacp.dvo.ru

Alexander I. Nikitin

e-mail: anikitin@iacp.dvo.ru

Vadim A. Timchenko, PhD (Tech.)

e-mail: vadim@dvo.ru

Institute of Automation and Control Processes Far Eastern Branch of the Russian Academy of Sciences (IACP FEB RAS)

The conceptual architecture of the cloud knowledge portal for laser-based additive manufacturing is presented. Its information and software components are described. A hybrid approach to intelligent support of technological equipment operators is proposed. It is based on the methods of knowledge engineering, case-based reasoning, as well as methods of mathematical modeling of physico-chemical processes that accompany laser powder cladding.

Keywords: decision support system, laser-based additive manufacturing, laser powder cladding, knowledge portal, ontologies for additive manufacturing.

CLLOUD IACPAAS HYBRID SERVICES FOR ASSESSING RISK OF CRITICAL CONDITIONS

Valeriya V. Gribova, Dr.Sc. (Tech.)

e-mail: gribova@iacp.dvo.ru

Elena A. Shalfeeva, PhD

e-mail: shalf@dvo.ru

Institute of Automation and Control Processes Far Eastern Branch of the Russian Academy of Sciences (IACP FEB RAS)

Various methods are used to support specialists solving of intellectual tasks and conducting observational or retrospective studies, and various approaches and tools are used for their implementation. This makes it difficult to use them comprehensively and cover complex tasks of professional activity. Ontologies are proposed as a basis for combining of intellectual tasks and for synthesis of intellectual spaces. However, they are practically not used as an integrating basis when creating hybrid intelligent systems that solve the problem by various methods, and with an explanation of the generated results. This paper proposes a unified semantic model for describing knowledge that accumulates various methods for solving problems of predicting states, assessing the risks of various events.

The model is implemented and tested on the IACPaaS platform. A version of the risk knowledge base has been created. A developed dictionary of all terms allowed in the description of knowledge and in documents from real medical practice has been implemented. Combining many models of forecasting and risk assessment on a single semantic structured basis makes it possible to comprehensively solve important problems of important problems of practical activity.

Keywords: critical conditions, risk assessment, semantic structured information model, hybrid services.

DEVELOPMENT OF AN APPROACH TO EXTRACTION AND CLUSTERIZATION OF BIBLIOGRAPHICAL DATA OF THE ELIBRARY.RU SYSTEM

*Alexander Dyrnochkin, Assistant of department
e-mail: dymo4kin@gmail.com*

*Vadim Moshkin, PhD
e-mail: postforvadim@ya.ru
Ulyanovsk State Technical University*

This article presents a system for extracting bibliographic information for subsequent scientometric analysis of publications. An algorithm for loading and preprocessing articles is described. An approach is proposed for the formation of scientific groups on a given topic by clustering texts of abstracts of articles. The paper also presents the results of experiments with data on articles from the elibrary scientific library.

Keywords: text analysis, web page parsing, text vectorization, clustering, k-means clustering.

SYMBOLIC DYNAMICS AND FORMATION OF COGNITIVE PROCESSES OF INTELLIGENT SYSTEMS

*Yu. T. Kaganov, PhD
Bauman Moscow State Technical University
e-mail: yurijkaganov@gmail.com*

The paper considers an approach based on nonlinear and symbolic dynamics for the analysis of intelligent systems. The process of formation of semantic structures as a result of self-organization processes in complex nonlinear systems is investigated. To implement the proposed approach, the theory of metagraphs and granular computations is used. The possibility of using the proposed approach for the study of cognitive processes of intelligent systems and the further development of artificial intelligence systems is shown.

Keywords: nonlinear dynamical systems, symbolic dynamics, metagraphic approach, granular computing, semiotic systems, artificial intelligence.

DEVELOPMENT OF A SYSTEM TO SUPPORT THE FORMATION OF RECOMMENDATIONS BASED ON CARDIOVASCULAR HEALTH ASSESSMENTS

*Pavel V. Platov, Aspirant
Ulyanovsk State Technical University
e-mail: chipa.platov@gmail.com*

Various areas of human activity, including health care, are currently being digitized. At the same time, the patient is becoming increasingly involved in this process. This article discusses a patient-centered system that analyzes the risk of cardiovascular events and generates recommendations. In particular, the functional requirements, data requirements, architecture and implementation details of the «CardioAnalytic» system are described.

Keywords: data mining techniques, cardiovascular risk management, system architecture, descriptive analysis, predictive analysis, prescriptive analysis, data analysis, functional requirements, architectural requirements.

FEATURES OF CONSTRUCTION OF APPLIED ONTOLOGY OF TYPICAL ARCHITECTURES OF INTEGRATED EXPERT SYSTEMS USING THE INTELLIGENT SOFTWARE ENVIRONMENT OF THE AT- TECHNOLOGY WORKBENCH

*Galina V. Rybina, D. of Techn. Sc., professor
NRNU MEPhI
e-mail: GVRybina@mephi.ru
Alexander A. Slinkov, graduate student
NRNU MEPhI
e-mail: sanch971@mail.ru
Andrey A. Grigoryev, undergraduate
NRNU MEPhI
e-mail: grigandal625@gmail.com*

The paper is devoted to the analysis of the results of construction and use of applied ontology of typical integrated expert systems (IES) architectures in the context of ontological approach to the construction of IES basis on the problem-oriented methodology. Special attention is paid to the ontology model in the form of a semantic network. A fragment of applied ontology for a standard design procedure (SDP) "Construction of tutoring IES", describing the interrelationships of individual processes associated with the construction of student models and model of tutoring, is considered as an example.

Keywords: integrated expert systems, problem-oriented methodology, ATTECHNOLOGY workbench, intelligent software environment, ontology model.

THREE-MODAL SYSTEM FOR TEXTUAL DESCRIPTION OF THE VISIBLE WORLD

Alexander A. Kharlamov, DSc

Institute of Higher Nervous Activity and Neurophysiology, RAS

e-mail: kharlamov@analyst.ru

Roman M. Zharkoy., President

ISS, Corp.

e-mail: roman.jarkoi@iss.ru

The strong variability of the extralinguistic (usually visual) representation of information in the mind has led to the need for a language that can describe the world less variably. Therefore, the transition to a textual description of the extralinguistic model of the world is the main trend in the modern analysis of world representations. Currently, the world is extralinguistically described most often through visual sensory capabilities. However, when it is represented in terms of visual modality, gaps arise that can be deaccentuated using representations of the somatosensory part of the model, which can be implemented in terms of the so-called 3D model. The paper presents a solution to the problem of converting a video sequence as a representation of the agent's behavior into a text script by representing the video sequence on a 2.5D model as recognition of a set of interconnected scene elements with a subsequent description of the scenes as the results of the production rules applied. The text as a sequence of scene descriptions is subjected to analysis procedures, including classification and comparison with instruction texts. Such an analysis, in particular, makes it possible to reveal the consistency of the description of a new type of scene with the existing scenario.

Keywords: Linguistic model of the world, Extralinguistic model of the world, Somatosensory representation as a part of the extralinguistic model of the world, Description of a scene by text, Sequence of scenes as a script, Script text analysis/

Section 7

LOCALIZATION TASK OF A MINI ROBOT USING SOSL METHOD

*Dmitriy A. Dobrynin, PhD Sc. Tech.
Federal Research Center for Computer Science and Control RAS
e-mail: rabota51@mail.ru*

The paper describes the task of localization of a mobile robot using distance sensors and an intelligent SOSL system. A method of teaching an intelligent SOSL system, map construction and localization process is proposed. It is shown that this localization method does not require large computing resources and is applicable on small mobile robots.

Keywords: robots, machine learning, navigation.

ANIMAT SPATIAL MEMORY MODEL

*Karpov V.E., PhD
NRC Kurchatov Institute
e-mail: karpov_je@mail.ru*

The paper considers the model of the animat's spatial memory. The model is based on a set of elements with dynamically formed associative links. The problem of remembering and subsequent reproduction of the route by an artificial agent is considered. Associative connections are determined by combinations of sensory signals, active motor elements, and the activity of internal aggregating elements. An important element of such an associative network is the counter elements responsible for the implementation of the concept of time. These are analogues of special time cells responsible for organizing the internal clock. The paper shows that the influence of such unfavorable factors as inaccuracy of sensory measurements, interference and control errors can be partly compensated by using the mechanism of residual activity of excited memory elements..

Keywords: associative memory, animat, route memorization, models of social behavior.

USING OF SYMPATHY IN EMOTIONAL-NEED AGENT WITH SEMIOTIC ARCHITECTURE

Sorokoumov P.S.

e-mail: petr.sorokoumov@gmail.com

Karpov V.E., PhD

e-mail: karpov_ve@mail.ru

NRC Kurchatov Institute

The tasks of the joint existence of a person and a robot in an environment are now becoming relevant. This may be necessary both for working in stable groups, when the robot is under more or less direct human control, and for episodic, spontaneous interaction. At the same time, it is necessary to reduce the risk of misunderstanding problems that are dangerous for both humans and robots. It is desirable to use for high-level control of intelligent robots that communicate with people models that are fairly simple, predictable, easily perceived by humans. Their obligatory element should be the ability of the robot to distinguish people from other objects of the environment, to treat them especially carefully. In this paper, we propose a model of an intelligent agent that changes behavior by inference in a dynamic environment, while taking into account both the agent's own needs and the behavior of the people around him. This is done by combining a semiotic model of the world and an emotional-need decision-making model, supplemented by a mechanism of empathy for other agents similar to the current one. This system allows users to naturally program such phenomena of group behavior as rapid social learning or contagiousness.

Keywords: emotional-need architecture, semiotics, intellectual robot.

INVESTIGATION OF PATH PLANNING ALGORITHMS OF A MOBILE ROBOT WITH ORIENTATION RESTRICTIONS

Kushnarev V.N. student

Bauman Moscow State Technical University

e-mail: grand_yarl@mail.ru

This article describes global and local planning algorithms for mobile robots that take into account orientation constraints. The global planning method is based on the A* algorithm for a regular square grid, which contains a new estimate for the transition from one cell to another. Estimation was implemented for 8 and 16 neighbour cells. Simulation results show advantages of using 16 neighbour cells instead of 8. The local planning method involves a new dynamic artificial potential field, which leads the robot in a safe direction. This new artificial field is described by a cone equation that can be modified for user's purposes. Advantages of dynamic artificial potential field algorithm are high computational speed and low requirement of memory.

Keywords: motion planning, path finding, orientation constraints, A* algorithm, artificial potential field.

INTEGRATED MODEL OF PLANNING AND OBJECT-CENTRIC REINFORCEMENT LEARNING

*Aleksandr I. Panov, PhD
FRC CSC RAS, MIPT, AIRI
e-mail: panov.ai@mipt.ru*

Modern model-based reinforcement learning methods are effective integrated models of simultaneous learning and planning. However, reducing the number of steps of interaction of a cognitive agent with the environment through the use of a dynamics model is limited by the fact that model learning can take quite a long time, especially in complex environments. In this paper, a new approach is proposed to structure the representations of states and dynamics models used by the agent by dynamically extracting objects and defining a specific set of actions that will bring these objects into the configurations necessary for the agent. The paper presents the reduction of the problem of object-centric reinforcement learning to a hierarchical case that uses time abstraction in the form of options. The Crafter environment is used as a demonstration example.

Keywords: reinforcement learning, simultaneous learning and planning, cognitive agent, object-centric reinforcement learning.

Section 8

CONCEPT OF CREATING OF INFORMATION SYSTEMS FOR AUTOMATIC CONTROL OF WELDERS' WORK

*Sergey V. Bolotov, Ph.D. (Engineering), Associate Professor
Belarusian-Russian University
e-mail: s.v.bolotov@mail.ru*

*Konstantin V. Zakharchenkov, Ph.D. (Engineering)
Belarusian-Russian University
e-mail: zaharchenkovky@mail.ru*

Vladislav A. Furmanov
*FLLC «GoDel Technologies»,
e-mail: fupkaa@gmail.com*

Eugene V. Makarov
*FLLC «iTechArt»,
e-mail: jastapanda@gmail.com*

Nikita K. Bobkov
Belarusian-Russian University

The concept of creating of information systems for automatic control of welders' work, based on the integrated use of hardware and software tools to register the parameters of welding processes and the subsequent automatic processing of information is proposed. The described approach provides a comprehensive objective assessment of welders' work due to the automatic registration of welding processes parameters using a recorder and a sensor unit and subsequent information processing without human involvement.

An integrated approach to assessing the welders' work involves the development of new hardware (recorders) for automatic control of welding process parameters and new algorithms for assessing the work of welders in the information systems of automatic control of welders' work. The advantage of the proposed approach is the implementation of automatic control of welding process parameters in real time, the ability to instantly inform the welder about the output of the controlled process parameters beyond the permissible values, an objective assessment of the welder's work, integrated processing of information about the progress of welding processes for subsequent analysis of the qualifications of each welder.

The proposed concept has been tested when creating the information system of welding control at the Certification and Testing Center of the Belarusian-Russian University and at State Production Association Belorusneft when welding main pipelines.

Keywords: information system, welding, recorder, welding process, complex data analysis.

MODELS FOR FORMING OPERATIONAL CALENDAR PLANS ON THE BASIS OF PRIORITY SCHEMES

G.B. Burdo, DSc

Tver State Technical University

e-mail: gbtms@yandex.ru

M.N.Koroleva, PhD

Bauman Moscow State Technical University

e-mail: Maria.Svyatkina@gmail.com

An intelligent approach to the development of algorithms for the formation of operational schedules for the passage of parts through operations based on priority schemes for automated process control systems in machine-building production systems is shown. Priority schemes are formed on the basis of a systematic approach, taking into account the hierarchy of organization plans and taking into account the state of production. The hierarchy of plans looks like global priority, internal priority, private priority. The rules for setting priorities themselves are based on the methodologies of lean manufacturing, quality management and artificial intelligence.

Keywords: mechanical engineering, production system, automated process control, system analysis, artificial intelligence, priority schemes, lean manufacturing.

CLASSIFICATION OF SOFTWARE DEVELOPMENT METHODOLOGIES

Dmitrii A. Gurianov, Post-graduate Student

e-mail: guryanovda@student.bmstu.ru

Konstantin S. Myshenkov, Dr. Sc. (Eng.), Professor

e-mail: myshenkovks@bmstu.ru

Bauman Moscow State Technical University

This paper considers the main life cycle models and software development methodologies, analyzes the existing classifications of methodologies and identifies their shortcomings. A new version of the classification has been developed, offering a more complete and broader hierarchical distribution of life cycle models and software development methodologies.

Keywords: project management, software, development, life cycle model, methodology, classification.

ONTOLOGY OF PROCESS DESIGN

G.B. Evgenev, DSc

Bauman Moscow State Technical University

e-mail: g.evgenev@mail.ru

An ontology of technological processes in mechanical engineering is proposed. A formal model of the ontology is constructed, which includes a finite set of concepts and relations between them, as well as set functions defined on them. Methods and means of developing technological processes are excluded. Examples of automatic design of individual processes, typical and group technological processes are given. Models and class diagrams of technological objects, as well as a conceptual data model of technological processing processes, have been developed. Models for designing assembly processes have been developed.

Keywords: artificial intelligence, ontology, intelligent design systems.

RESEARCH OF DATA COMPRESSION APPROACHES IN THE DEVELOPMENT OF DIGITAL TWINS

Anton M. Skalkin, master's degree

e-mail: anton.skalkin73@gmail.com

Anton A. Romanov, PhD

e-mail: romanov73@gmail.com

Ulyanovsk State Technical University

Digital doubles can include data as time series that needs to be processed to avoid excessive data overgrowth. To do this, various compression, smoothing, and fuzzy methods — F-transformation can be used. Using these methods is a necessity in our world, where there are increased devices with high data refresh rates, such as sensors, metrics, IoT.

To evaluate the variety of compression methods, an original approach was proposed to assess the quality of the compression produced.

The use of this technique will allow: to abstract from the type of analytics conducted and, moreover, the methods that will be used for them; to abstract from the compression/smoothing method; to obtain the best compression model parameters for future use automatically; to obtain analysis results at the same level with the greatest data compression.

Keywords: digital twins, time series, compression, smoothing, F-transform, compression metric.

АВТОРСКИЙ УКАЗАТЕЛЬ

| | | | |
|------------------|----------|-----------------|----------|
| Аверкин А.Н. | 127 | Мартирова Т.А. | 97 |
| Афанасьева Т.В. | 6 | Мисник А.Е. | 18 |
| Бобков Н.К. | 257 | Михеенкова М.А. | 60 |
| Болотов С.В. | 257 | Мошкин В.С. | 156 |
| Борисов В.В. | 18 | Мышенков К.С. | 279 |
| Бурдо Г.Б. | 269 | Найденова К.А., | 97 |
| Гапанюк Ю.Е. | 28 | Никитин А.И. | 135 |
| Грибова В.В. | 135, 146 | Панов А.И. | 248 |
| Григорьев А.А. | 188 | Платов П.В. | 175 |
| Гурьянов Д.А. | 279 | Редько В.Г. | 108, 119 |
| Гуськов Г.Ю. | 39 | Романов А.А. | 39, 304 |
| Добрынин Д.А. | 206 | Рыбина Г.В. | 188 |
| Дырночкин А.А. | 156 | Скалкин А.М. | 39, 304 |
| Евгеньев Г.Б. | 293 | Слиньков А.А. | 188 |
| Еремеев А.П. | 74 | Сорокоумов П.С. | 228 |
| Жаркой Р.М. | 197 | Сохова З.Б. | 119 |
| Забежайло М.И. | 50 | Строева Ю.В. | 39 |
| Захарченков К.В. | 257 | Тимченко В.А. | 135 |
| Каганов Ю.Т. | 163 | Филиппов А.А. | 39 |
| Карпов В.Э. | 218, 228 | Финн В.К. | 60 |
| Королева М.Н. | 269 | Фурманов В.А., | 257 |
| Кулинич А.А. | 85 | Харламов А.А. | 197 |
| Кульчин Ю.Н. | 135 | Цапенко И.В. | 74 |
| Кушнарев В.Н. | 237 | Шалфеева Е.А. | 146 |
| Макаров Е.В. | 257 | Ярушев С.А. | 127 |

СОДЕРЖАНИЕ

СЕКЦИЯ 4. ИНЖЕНЕРИЯ ЗНАНИЙ И ОНТОЛОГИИ

| | |
|--|----|
| <i>Афанасьева Т.В.</i> Разработка метода создания пациент-ориентированных рекомендаций по управлению сердечно-сосудистым здоровьем у мужчин..... | 6 |
| <i>Борисов В.В., Мисник А.Е.</i> Применение онтологического инжиниринга в образовательных системах..... | 18 |
| <i>Гапанюк Ю.Е.</i> Основные положения многомерно-метаграфовой модели данных и знаний..... | 28 |
| <i>Гуськов Г.Ю., Романов А.А., Скалкин А.М., Строева Ю.В., Филиппов А.А.</i> Модель онтологии для поддержки принятия решений на основе нечеткого логического вывода..... | 39 |
| <i>Забезжайло М.И.</i> Устойчивость эмпирических зависимостей и проблема объяснения результатов интеллектуального анализа данных..... | 50 |
| <i>Михеенкова М.А., Финн В.К.</i> Об аргументационном подходе к анализу мнений..... | 60 |

СЕКЦИЯ 5. КОГНИТИВНЫЕ МОДЕЛИ В ИСКУССТВЕННОМ ИНТЕЛЛЕКТЕ

| | |
|---|-----|
| <i>Еремеев А.П., Цапенко И.В.</i> Интеграция средств когнитивной графики и нечеткой логики для представления и диагностирования сложных патологий зрения..... | 74 |
| <i>Кулинич А.А.</i> Семиотические когнитивные карты..... | 85 |
| <i>Найденова К.А., Мартирова Т.А.</i> Когнитивные основы классификации эллипсисов..... | 97 |
| <i>Редько В.Г.</i> Как автономный компьютерный агент может самостоятельно открывать законы природы..... | 108 |
| <i>Редько В.Г., Сохова З.Б.</i> Модель познания закономерностей автономным агентом..... | 119 |

СЕКЦИЯ 6. ГИБРИДНЫЕ ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

| | |
|--|-----|
| <i>Аверкин А.Н., Ярушев С.А.</i> Исследование развития систем объяснительного искусственного интеллекта..... | 127 |
|--|-----|

| | |
|--|-----|
| <i>Грибова В.В., Кульчин Ю.Н., Никитин А.И., Тимченко В.А.</i> Концепция гибридной интеллектуальной системы поддержки принятия решений в процессах лазерного аддитивного производства..... | 135 |
| <i>Грибова В.В., Шалфеева Е.А.</i> Гибридные облачные IASaaS -сервисы оценки рисков критических состояний..... | 146 |
| <i>Дырочкин А.А., Мошкин В.С.</i> Разработка подхода к извлечению и кластеризации библиографических данных системы elibrary.ru..... | 156 |
| <i>Каганов Ю.Т.</i> Символическая динамика и формирование когнитивных процессов интеллектуальных систем..... | 163 |
| <i>Платов П.В.</i> Разработка системы поддержки формирования рекомендаций на основе оценки сердечно-сосудистого здоровья..... | 175 |
| <i>Рыбина Г.В., Слинков А.А., Григорьев А.А.</i> Особенности построения прикладной онтологии типовых архитектур интегрированных экспертных систем с использованием средств интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ..... | 188 |
| <i>Харламов А.А., Жаркой Р.М.</i> Трехмодальная система для текстового описания видимого мира..... | 197 |

СЕКЦИЯ 7. ИНТЕЛЛЕКТУАЛЬНЫЕ АГЕНТЫ, РОБОТЫ И КОЛЛЕКТИВЫ РОБОТОВ

| | |
|---|-----|
| <i>Добрынин Д.А.</i> Задача локализации миниробота с использованием SOSL метода..... | 206 |
| <i>Карпов В.Э.</i> Модель пространственной памяти анимата..... | 218 |
| <i>Карпов В.Э., Сорокоумов П.С.</i> Реализация симпатии в эмоционально-потребностном агенте с семиотической архитектурой..... | 228 |
| <i>Кушнарев В.Н.</i> Исследование алгоритмов планирования маршрута мобильного робота с учетом ограничений на ориентацию..... | 237 |
| <i>Панов А.И.</i> Иерархическая постановка задачи объектно-центричного обучения с подкреплением..... | 248 |

**СЕКЦИЯ 8. ИНТЕЛЛЕКТУАЛЬНОЕ ПРОИЗВОДСТВО,
ИНТЕЛЛЕКТУАЛЬНОЕ УПРАВЛЕНИЕ ЖИЗНЕННЫМ ЦИКЛОМ**

| | |
|---|-----|
| <i>Болотов С.В., Захарченков К.В., Фурманов В.А., Макаров Е.В., Бобков Н.К.</i> Концепция создания информационных систем автоматического контроля работы сварщиков..... | 257 |
| <i>Бурдо Г.Б., Королева М.Н.</i> Модели формирования оперативных календарных планов на основе приоритетных схем..... | 269 |
| <i>Гурьянов Д.А., Мышенков К.С.</i> Классификация методологий разработки программного обеспечения..... | 279 |
| <i>Евгеньев Г.Б.</i> Онтология проектирования технологических процессов..... | 293 |
| <i>Скалкин А.М., Романов А.А.</i> Исследование подходов сжатия данных при построении цифровых двойников..... | 304 |
| Abstracts | 310 |
| Авторский указатель..... | 328 |

Научное издание

Сборник трудов
XI Международной научно-практической конференции
«Интегрированные модели и мягкие вычисления
в искусственном интеллекте» ИММВ-2022.
Труды конференции в 2-х томах
Том 2

Российская ассоциация искусственного интеллекта

Подписано в печать 11.05.2022 г.
Формат 60x84¹/₁₆. Тираж 200 экз. Усл. печ. л. 19,3.

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

