

## УДОСКОНАЛЕННЯ ПРОЦЕСІВ ЖИТТЄВОГО ЦИКЛУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

УДК 629.78.018

І.Б. Сироджа, І.Б. Туркин, Л.А. Волобуева,  
Е.В. Соколова, В.А. Постернакова

### ПРОБЛЕМА ИНТЕГРАЦИИ ПРОГРАММНОЙ ИНЖЕНЕРИИ С ИНЖЕНЕРИЕЙ ЗНАНИЙ

Национальный аэрокосмический университет им. Н.Е. Жуковского "ХАИ", Украина  
[po\\_as\\_k603@d6.khai.edu](mailto:po_as_k603@d6.khai.edu)

*Розглянуто актуальну і складну проблему інтеграції програмної інженерії з інженерією знань, що перебуває у стані свого початкового розвитку. Значення цієї проблеми для створення й удосконалювання методології, технології та інструментальних засобів програмування важко перебільшити. Проаналізовано сучасний стан проблеми та обґрунтовано необхідність інтеграції методів інженерії знань і програмної інженерії. Наведено прогресивні ідеї, принципи і парадигми інтеграції та приклади досліджень за даною темою, які виконуються у США, Росії і в Україні.*

*Рассмотрена актуальная и сложная проблема интеграции программной инженерии с инженерией знаний, пребывающая пока в начальной стадии своего развития. Значение этой проблемы для создания и совершенствования методологии, технологии и инструментальных средств программирования трудно переоценить. Выполнен анализ современного состояния проблемы и обоснована необходимость интеграции методов инженерии знаний и программной инженерии. Приведены прогрессивные идеи, принципы, парадигмы интеграции, а также примеры исследований по данной теме, которые проводятся в США, России и Украине.*

*And difficult the issue of the day of integration of the programmatic engineering with engineering of knowledge, being while in the initial stage of the development, is considered. Value of this problem it is difficult to over-estimate for creation and perfection of methodology, technology and tools of programming. The analysis of the modern state of problem is executed and the necessity of integration of methods of engineering of knowledge and programmatic engineering is grounded. Progressive ideas, principles, paradigms of integration, and also examples of researches, are resulted on this theme, which are conducted in the USA, Russia and Ukraine.*

**Ключові слова:** програмная інженерія, інженерія квантов знань, інтеграція, парадигма, концепт, транслятор, компілятор, фрейм, квант знань

#### Введение

Жизненная необходимость решения сложных научных и прикладных задач на этапах развития инженерии знаний (ИЗ) и программной инженерии (ПИ) привели их к одной **общей проблеме создания** методологии, технологии и инструментальных средств программирования. Но, в силу принципиального различия путей развития двух инженерий, возник глубокий смысловой и функциональный разрыв между методами программирования в ИЗ и ПИ. Это существенно препятствует разработке интеллектуальных информационных технологий для успешного развития многих реальных задач науки и практики. Более того, без интеграции средств и усилий специалистов в областях ИЗ и ПИ невозможно даже приступить к решению важнейшей задачи современности – автоматизации творческого труда, стоящей в очереди за успешной

компьютеризацией рутинной деятельности человека. К тому же, автоматизация (компьютеризация) творческих процессов непосредственно связана с вечной научной и производственной задачей знание-ориентированного принятия решений в условиях неопределенности. Большую потребность в интеллектуальных информационных технологиях (ИИТ) поддержки принимаемых решений испытывают современные конструкторы, технологи, диспетчеры, менеджеры бизнес-процессов и компаний, врачи, социологи, работники образования и др., деятельность которых базируется на интуиции, опыте и знаниях, систематизированных в компьютерных базах знаний. Особую актуальность получили методы представления и компьютерного манипулирования знаниями в связи с появлением экспертных систем (expert systems) в многочисленных предметных областях, а

также, с необходимостью повышения качества информационного поиска в Интернете. Именно стремление к разумному, индустриальному подходу к решению указанных задач и им подобных привело к общей проблеме интеграции ИЗ с ПИ.

Суть смыслового и функционального разрыва между обеими инженериями мы опишем кратко ниже в пункте анализа исследований и публикаций. Укажем на принципиальное различие путей развития инженерии знаний (knowledge engineering) и программной инженерии (software engineering), побудившее этот разрыв. Напомним, что ИЗ – одно из фундаментальных направлений искусственного интеллекта, возникшего в 1950-х годах, а также, современного программирования, занимающегося созданием и развитием языков, методов представления и компьютерной обработки знаний для решения задач творческого характера. Компьютерное решение таких задач требует использования знаний, которые имеют интегральный либо гибридный характер, то есть, применяются в комплексе: концептуальные, процедурные, фактуальные и эвристические знания. Концептуальные знания определяют концепции проблемной области и отношения между ними. Процедурные знания состоят из алгоритмов, типовых процедур решения задач, их программной реализации, которые всегда используются для компьютерного решения любой задачи и жестко встроены в исходный текст программы. Фактуальные знания – это конкретные факты из предметной области и их связи между собой. Эвристические знания представляют собой неформальные правила рассуждений, интуитивно отражающие практический опыт решения задач в проблемной области [1-4].

К основным моделям компьютерного представления знаний относятся: продукции или правила (rules), фреймы (frames), семантические сети (semantic nets) и кванты знаний (knowledge quantum) или k-знания. Продукции используются для представления эвристических и фактуальных знаний и структурируются в виде наборов правил (rule sets) для вывода некоторого целевого утверждения (goal). Фреймы относятся к иерархическим структурам, удобным для представления концептуальных знаний и их взаимосвязи, а также для сопоставления структур знаний с объектами предметной

области с целью их анализа и классификации. Семантические сети являются, по сути, нагруженными ориентированными мультиграфами, удобными, как и фреймы, для представления концептуальных знаний, а также других знаний [1,3,4].

Кванты знаний – принципиально новая универсальная модель представления произвольных знаний, которая наследует достоинства продукций, фреймов и семантических сетей, но имеет фундаментальное преимущество в том, что обладает возможностями математического представления в трех формах: множественной, векторно-матричной и предикатной. Тем самым, модели и методы инженерии квантов знаний (ИКЗ) имеют существенные преимущества перед существующими. Основные их достоинства состоят в непосредственной возможности представления информационных объектов (k-знаний) в фазовом пространстве состояний, а также в открытом доступе ко всему арсеналу средств математики для совершенствования инструментария и методологии ИКЗ [5,6]. В свою очередь, программная инженерия развивалась на основе совершенно иных вычислительных моделей и парадигм, нежели ИЗ. Комплексным предметом исследования ПИ являются модели и методы спецификации, проектирования, реализации, тестирования и сопровождения программ и программных систем, а также модульное, объектно-ориентированное программирование и т.п. И всё-таки, пути развития ИЗ и ПИ пересеклись под воздействием интенсивного практического востребования и применения систем искусственного интеллекта, а также ИИТ, что окончательно обусловило реальность рассматриваемой проблемы их интеграции [7-16].

### 1. Анализ исследований и публикаций

С целью конкретизации содержательной постановки рассматриваемой проблемы проведём краткий анализ исследований и публикаций по выбранной теме. При этом, внимание сосредоточим на основных очевидных факторах, вскрывающих не только содержательный и функциональный разрыв между методологиями ИЗ и ПИ, но одновременно указывающих на научно-технические подходы и принципы интеграции различных методов программирования.

### 1.1 Исторический фактор

Исторически сложилось так, что инженерия знаний начала в 1950-х годах интенсивно развиваться в рамках искусственного интеллекта (ИИ), (Artificial Intelligence (AI)) – науки о создании машин и программ, способных действовать таким образом, что человек будет воспринимать их как разумные. Это привело к разработке Ньюэллом, Симоном и Шоу программы Logic Theorist для обработки списков [1]. В конце 50-х в лабораториях Массачусетского технологического института (MIT) Джоном Мак Карти был разработан язык инженерии знаний LISP для символьной обработки, который заменил IPL и стал основным языком ИИ. В 1970г. Д. Ленет создал в Стэндфордском университете программу «Автоматический математик», а позднее – программу EURISKO, что позволило открыть новые теории в математике. В это же время в MIT был разработан язык ПРОЛОГ (Prolog – Программирование логики), основанный на идее использования метода резолюций для эффективного логического вывода в исчислении предикатов 1-го порядка. Далее последовали многочисленные примеры применения нечёткой логики Лотфи Заде для управления различными процессами [1,2].

1980 годы оказались плодотворными для ИЗ, после того, как продажи аппаратных средств (LISP-машин, PROLOG-машин) и программного обеспечения с использованием искусственного интеллекта (ИИ) превысили в 1986 году 400 млн. долларов. Большую часть дохода принесли продажи экспертных систем (ЭС) на LISP и PROLOG, которые использовались многими компаниями для разработки ископаемых, прогнозирования инвестиций, медицинской и технической диагностики, а также для поддержки принятия решений в управлении бизнес-процессами и в технологической подготовке производства [1-4]. Нейронные сети в 1980-е годы также переживали возрождение, и нашли применение в решении задач распознавания речи, графических образов, а также продемонстрировали возможность обучения машин [1,2].

1990-е годы стали новой эпохой в развитии приложений инженерии знаний, когда было установлено, что встраивание элементов ИИ в программные системы позволяет добиться решения многих проблем быстрее и эффективнее, чем при использовании

традиционных средств программной инженерии [2-6]. Знаменательным событием в развитии компьютерных игр с использованием ИИ стало создание в 1997 году в университете Карнеги Меллон суперкомпьютера для игры в шахматы Deep Blue, который смог победить чемпиона мира по шахматам Гарри Каспарова. Еще одно важное событие в развитии ИИ в эти годы произошло на расстоянии 60 млн. миль от Земли. На борту космического аппарата США была установлена программная система Deep Space 1 (DS1), которая могла тестировать технологии 12-ой степени риска, включая полет кометы и тестирование КА для будущих космических полетов. В составе DS1 функционировала система ИИ под названием Remote Agent, которая продемонстрировала способность управлять некоторое время космическим кораблем, позволяя экипажу сконцентрироваться на выполнении других задач.

Несомненно, исторический фактор оказывал и продолжает оказывать существенное влияние на формирование разнохарактерных теоретических основ развития инженерии знаний и программной инженерии, различий в мышлении, терминологии, а также в наметившихся подходах и принципах интеграции их методов программирования. Это влияние осуществлялось в силу объективных исторических закономерностей становления и развития научно-технического прогресса во времени.

### 1.2. Фактор разнохарактерного научного базиса

Анализ публикаций показывает, что научный базис ИЗ формировался на основе использования моделей и методов системного анализа, математической логики, теории алгоритмов и продукций, кибернетики и системы преобразования термов (term-rewriting systems). Методологические результаты и опыт программирования накапливались, в основном, усилиями специалистов в указанных областях, программирующих на языках ЛИСП, ПРОЛОГ и их диалектах. Применение таких языков требует принципиально отличных способов мышления от используемых в программной инженерии, которая базируется на применении процедурных языков ФОРТРАН, АЛГОЛ, ПАСКАЛЬ, структурного, модульного и

объектно-ориентированного программирования.

Теоретический базис ПИ был сформирован на основе использования теории формальных грамматик и языков, теории графов, теории компиляторов и трансляторов, а также численных методов и дискретных структур математики, что не очень способствовало взаимопониманию и взаимодействию ИЗ и ПИ.

### 1.3. Фактор различия терминов

Наличие разнохарактерного теоретического базиса обеих дисциплин обусловило различия и в терминологии. Специалисты по ИЗ мыслят в терминах, связанных с понятиями: «гипотеза», «концепт», «продукционные и фреймовые базы знаний», «квант знаний», «атом», «список», «демон», «индуктивный и дедуктивный выводы на знаниях», «фазификация и дефазификация», «нечёткость», «степень истинности», «импликативные и функциональные закономерности» и др.

Характерные термины, используемые в ПИ, связаны с понятиями: «ячейка памяти», «значение», «массив», «присваивание», «стек», «объект», «класс», «инкапсуляция», «программный модуль», и др., что затрудняет при конструировании программ использование знание-ориентированного инструментария ИЗ. Более того, отсутствуют пока программисты, использующие, например, современные платформы Java, .Net и владеющие логическим программированием в системах Turbo Prolog либо Visual Prolog. А решение многих практических задач требует использования методов программирования ПИ именно в сочетании с методами ИЗ [2].

### 1.4. Фактор несогласованности платформ программирования

Все существующие платформы программирования в ИЗ и в ПИ, как свидетельствует анализ публикаций, создавались разобщено и несогласованно, то есть, без предусмотренных адекватных и удобно реализованных программных средств для совместного использования при обработке гибридных знаний, абстрактных типов данных и объектно-ориентированных структур в решениях практических задач. К сожалению, ни одна из широко используемых платформ программной инженерии: Java, .Net, Active Oberon, Eiffel и др. не содержит адекватных средств представления знаний, используемых

как ключевые понятия и конструкции базовых языков. И с другой стороны, ни одна из известных систем инженерии знаний: KEE, Common LISP, Turbo Prolog, Visual Prolog, Ontolingua, Protege и др. не содержат адекватных, эффективно реализованных, современных средств программной инженерии. Возможности использования программных модулей на традиционных языках ПИ в системах ИЗ либо отсутствуют, либо ограничены, например, возможностью запустить из набора продукций целое приложение для Windows (.exe) и использовать его результат в виде записанного файла [8-12].

Приведенные факторы не только свидетельствуют о существовании семантического разрыва между методами ПИ и ИЗ, обусловившего проблему интеграции, но и вполне отчетливо указывают на возможные принципы и подходы к решению проблемы путем научно-инженерного преодоления описанного разрыва. В настоящее время, на наш взгляд, целесообразно реализовывать параллельно два подхода к решению проблемы интеграции программной инженерии с инженерией знаний: «интеграция в большом» и «интеграция в малом». На сей счет имеются реальные основания.

Начиная с 1980 г. В Санкт-Петербургском университете под руководством проф. В. О. Сафронова (до 2001 г. в лаборатории технологии программирования и экспертных систем, а с 2001 г. в лаборатории Java-технологии НИИ математики и механики) разрабатываются методы интеграции ИЗ и инженерии программ. Сафронов В. О. и его ученики развивают подход «интеграции в большом», в основе которого лежит принцип расширения современных языков и систем программирования средствами инженерии знаний: концепциями, библиотеками (базами) знаний, языковыми конструкциями и т.п. Существуют уже реализации интегрированных систем программирования в языке представления знаний Турбо-Эксперт [7] и в экспериментальных версиях языков и систем Java Expert [8], где расширения соответственно Турбо Паскаля и языка Java выполнены средствами представления гибридных знаний. В работе [12] содержится информация о том, что в системе C# Expert расширение языка C# реализовано средствами представления фреймовых и продукционных знаний.

В последние годы идеи и принципы интеграции «в большом» методов ИЗ и ПИ получили распространение в Европе и США. К примеру, в работе [11] и докладах на крупных международных конференциях подход к «интеграции в большом» развивает профессор Jeff Zhuk из University of Phoenix, Arizona, USA, с которым сотрудничают некоторые исследователи из России.

На Украине и практически во всех развитых странах мира уже продолжительное время развивается подход к интеграции «в малом», который базируется на принципе встраивания технологий и алгоритмов искусственного интеллекта (ИИ) в программное обеспечение (ПО) автоматизированных систем с целью создания более «умного», адаптирующегося к среде и практически полезного ПО. Об этом свидетельствуют многочисленные публикации отечественных и зарубежных авторов, начиная с 1960-х годов, а также работы приведенные ниже [1-6, 17-25].

Достаточно оригинальные теоретические и практические результаты в направлении интеграции «в малом» методов инженерии знаний (ИКЗ) и программной инженерии получены исследователями Харьковской школы в Национальном аэрокосмическом университете (ХАИ) имени Н. Е. Жуковского. Благодаря отмеченным во введении достоинствам, методология ИКЗ обеспечила создание научно-обоснованного инженерного базиса для разработки не дорогостоящих, но эффективных интеллектуальных информационных технологий (ИИТ) для автоматизации творческого труда. Квантовые ИИТ легко встраиваются в существующие ПО АСУ производством и технологическими процессами [5,6] без никакого дополнительного расширения современных языков программирования и гибко подстраиваются под требования пользователя. Такой эффект встраивания ИИТ, на наш взгляд, существенно способствует общему развитию ИИ по пути его интенсивного и массового практического востребования. На основании изложенного сформулируем содержательную постановку проблемы интеграции и методов инженерии знаний и программной инженерии на современном этапе развития информатизации общества.

## **2. Содержательная постановка проблемы**

На содержательном уровне постановка проблемы интеграции методов ИЗ и ПИ «в большом» заключается в следующем.

Учитывая современный прогресс в информатизации мирового сообщества, изложенные в п. 1 факторы, вскрывающие семантический разрыв между методами программирования в ИЗ и в ПИ, а также всё нарастающую потребность в ИИТ для решения сложных научных и прикладных задач, необходимо создать научно обоснованную интегрированную систему (платформу) программирования на основе разработки новых концепций, парадигм и принципов расширения современных языков программирования средствами инженерии знаний с целью реализации эффективных интеллектуальных решений (intelligent solutions) реальных задач.

Наряду с этим общим подходом, постановка проблемы интеграции «в малом» состоит в создании парадигм, методов и программного инструментария инженерии знаний, интегрированных с современными средствами программной инженерии, на принципах встраивания алгоритмов и технологий искусственного интеллекта в ПО автоматизированных систем для обеспечения их адаптируемости к среде и практической полезности.

Далее рассмотрим на конкретных примерах подходы к интеграции ИЗ и ПИ «в большом» и «в малом».

## **3. Пример интеграции ИЗ и ПИ «в большом»**

Интеграцию ИЗ и ПИ «в большом» проиллюстрируем на примере системы KNOWLEDGE.NET, разработанной под руководством проф. В. О. Сафонова в Санкт-Петербургском университете [12]. Идея интеграции базируется на расширении (add-in) одной из мощных интегрированных сред современного программирования – Microsoft Visual Studio .Net 2005 (Whidbey) с целью обеспечения возможности применения всего спектра средств проектирования, разработки и отладки программ, предоставляемых этой средой, при разработке баз знаний в системе KNOWLEDGE.NET. Суть идеи заключается в способах реализации расширений, предназначенных для представления знаний, а

также их компиляции в исполняемый объектный код на базовом языке. Для реализации расширений используется конвертирование, т.е. перевод расширений в программу на базовом языке (например, Java, C# и др.), содержащую вызовы методов специализированной библиотеки (API) для представления и обработки знаний. Для компиляции результатов конвертирования в исполняемый объектный код используется «штатный» компилятор базового языка. Например, для Java используют компилятор Javac, входящий в состав Java Developer's Kit (JDK) фирмы Sun Microsystems, а также для C# -- компилятор фирмы Microsoft, входящий в состав Microsoft .Net Framework.

В связи с этим, система KNOWLEDGE.Net состоит из следующих 4-х основных программных компонент.

1. Конвертор с языка Knowledge .Net в базовый язык C# платформы Microsoft .Net. Последующая компиляция программ на C#, как результатов конвертирования, обеспечивается штатными средствами интегрированной среды Visual Studio .Net.

2. Редактор и визуализатор знаний Knowledge Editor, обеспечивающий визуализацию, проектирование, реализацию и модификацию в интерактивном режиме исходных текстов баз знаний на языке Knowledge .Net.

3. Система Knowledge Prospector для извлечения знаний в формате Knowledge .Net из текстов на естественном языке, которая ориентирована также на извлечение знаний из Internet.

4. Конвертор KIF (Knowledge Interchange Format) [15]. В целом система Knowledge .Net [14] предназначена для разработки и использования библиотек (баз) знаний из разнообразных предметных областей при совместном функционировании с другой системой аспектно-ориентированного программирования ASPECT .NET. [10, 13]. Семантика онтологии Knowledge .NET схожа с концепцией языка представления знаний OWL, разработанного W3C консорциумом [16]. В KNOWLEDGE.NET используется терминология, принятая в онтологических системах (Concept-Property-Individual), вместо терминологии фреймовых систем (Frame-Slot-Instance).

#### 4. Пример интеграции ИЗ с ПИ «в малом»

##### 4.1. Парадигма инженерии квантов знаний

В отличие от интеграции «в большом», интеграция инженерии знаний с программной инженерией «в малом» имеет множество реализаций встраивания алгоритмов ИИ в ПО автоматизированных систем различного назначения в соответствии с постановкой проблемы в п. 2. Существует достаточно много разнообразных идей, тенденций, парадигм и способов реализации этого подхода к интеграции ИЗ с ПИ [1-6].

Остановимся кратко на оригинальном квантовом подходе к интеграции в «малом», который основан на парадигме инженерии квантов знаний (ИКЗ) [5] для принятия идентификационных и прогнозных решений в условиях неопределенности. Оригинальность парадигмы ИКЗ, прежде всего, состоит в представлении знаний порциями (квантами) различного уровня сложности (0-й уровень – представление числом, 1-й уровень – числовым кортежем, 2-й уровень – числовой матрицей). Кванты знаний имеют строгое формальное определение в терминах теории алгоритмов как разноуровневые алгоритмические структуры, образуемые из нескольких терминальных элементов путем конечного числа применений операторов суперпозиции и конкатенации. Кванты знаний (к-знания) образуют класс алгоритмических структур, каждая из которых имеет три составляющие: смысловую, информационную и операторную. Смысловая составляющая определяет семантику и логику квантового события относительно объекта принятия решений (ОПР) с его именем и шкалами измерения разнотипных признаков. Информационная составляющая представляет квантовое событие логической зависимостью целевого признака ОПР с его внешними признаками посредством доменов в векторно-матричной числовой и символьной форме. Операторная составляющая обеспечивает алгоритмическую обработку информации об ОПР, предусмотренную информационной и смысловой составляющей кванта знаний. В монографиях [5] [6] детально изложена методология ИКЗ на основе разработанных системных моделей разноуровневых алгоритмических квантов знаний (δ PAKЗ-

моделей) и  $\delta$  РАКЗ-методов. Они допускают модификацию по параметру  $\delta \in \{t, \pi, \nu, \varphi\}$  в зависимости от условий  $\delta$ -неопределенности и видов  $\delta k$ -знаний: точных (при  $\delta = t$ ), приближенных ( $\delta = \pi$ ), вероятностных ( $\delta = \nu$ ) и нечетких ( $\delta = \varphi$ ). В отличие от существующих общих  $\delta$  РАКЗ-метод принятия решений в условиях  $\delta$ -неопределенности обеспечивает построение обучаемых баз  $\delta k$ -знаний (БдкЗ) в виде квантовых матриц либо квантовых сетей вывода решений ( $\delta$ -КСВР) как моделей логических рассуждений человека от посылок к целевым следствиям, т.е. принимаемым решениям.

По существу, БдкЗ впервые представляет собой систему причинно-следственных импликативных и/или функциональных закономерностей предметной области. Именно поэтому стало возможным одновременное использование  $\delta$ -КСВР в роли БдкЗ и механизма вывода решений, а также непосредственное эффективное манипулирование  $\delta k$ -знаниями посредством машинных алгебр, не прибегая к специальным языковым расширениям. Тем самым, средства ИКЗ в настоящее время наиболее адекватны и эффективны для решения проблемы интеграции «в малом», сформулированной в п. 2. Однако, главным результатом в разработке парадигмы ИКЗ являются приведенные ниже теоремы о соотношениях для оценивания достоверности гипотез о существовании импликативных и функциональных закономерностей требуемого ранга  $r$  в пространстве признаков ОПР, исходя из объема  $m \times n$  обучающей таблицы эмпирических данных (ТЭД)  $T_0(m, n)$  и/или сценарных примеров обучающих знаний (СПОЗ). Ранг закономерности определяется количеством  $r$  признаков ОПР из общего числа  $n$  в комбинациях  $C_n^r (2 \leq r \leq n)$  находящихся в устойчивой импликативной или функциональной связи при количестве  $t$  наблюдений. Оценка достоверности существования гипотезы представлена функцией от аргументов  $m, n, r$ . Это позволяет разработчику сформировать обучающие ТЭД и СПОЗ такого объема ( $m \times n$ ), при котором обеспечивается с заданной надежностью  $\eta$  индуктивное построение БдкЗ или  $\delta$ -КСВР как

системы закономерностей адекватной сложности по рангу  $r$ .

#### 4.2. Оценивание сложности импликативной БдкЗ по рангу запретов и их индуктивный поиск по обучающим $\delta k$ -знаниям

Построим расширенное булево  $N$ -множество  $B$ :

$$B = \text{CON}_{j=1}^n \langle B^j \rangle = \langle B^1, B^2, \dots, B^n \rangle = \langle \beta_1^1, \beta_1^2, \dots, \beta_1^n, \beta_{\eta+1}^1, \beta_{\eta+1}^2, \dots, \beta_{\eta+1}^n, \dots, \beta_{\eta+r_2+\dots+r_j}^j, \dots, \beta_N^n \rangle, \quad (1)$$

элементы которого отвечают бинарным значениям  $\beta_i^j$ , ( $i=1, 2, \dots, r_j$ ;  $j=1, 2, \dots, n$ ) всех признаков  $x_1, x_2, \dots, x_n$  ОПР. Очевидно,  $N$ -я декартова степень множества  $B$  (1) и образует расширенное булево пространство  $B^N$ :

$$B^N = \underbrace{B \times B \times \dots \times B}_{N \text{ раз}} = \{\bar{\beta}\}, \quad (2)$$

которое будем называть *логическим пространством реализаций  $\delta$ РАКЗ-моделей*, а его векторные элементы (точки, реализации)  $\beta$ :

$$\bar{\beta} = (\beta_1, \beta_2, \dots, \beta_N) \in B^N \quad (3)$$

назовем *булевскими моделями ОПР* для описания результатов наблюдений за объектами.

Предположим, что точки пространства  $B^N$  (2) отвечают множеству  $T$  мыслимых объектов, часть которых  $T_r \subseteq T$  относится к реальным ОПР исследуемых классов. Поскольку, на практике  $T$  и  $T_k$  недоступны непосредственному перечислению (их мощности очень велики и соотносятся как  $|T_r| \ll |T|$ ), то о закономерностях в  $T_r$  приходится судить по экспериментальному (обучающему) множеству  $T_0 = T_0(m, N) \subseteq T_r$ , которое также намного меньше  $T_r$ . Следовательно, в силу реальных практических условий:

$$T_0(m, N) \ll T_r \ll T \quad (4)$$

возникает необходимость построения *индуктивной модели мира* (т.е. БдкЗ) с целью принятия *идентификационных* и *прогнозных* решений на основе  $\delta k$ -знаний, извлекаемых из любых источников и любыми способами.

**Определение 1.** Устойчивая связь между  $r$  характеристиками ОПР из общего числа  $n$ , ( $r \leq n$ ), выражающая *недопустимость (запрет)* хотя бы одной комбинации их значений на множестве  $\delta k$ -знаний, ( $\delta \in \{t, \pi, \nu, \varphi\}$ ), называется *импликативной закономерностью* или *запретом  $r$ -го ранга*.

Кроме *импликативной* связи между признаками может существовать и *функциональная* связь как частный случай *импликативной*. Отличие их состоит в том, что при *функциональной* связи значения некоторых признаков, называемых *аргументами*, всегда определяют значение другого признака-*функции*, в то время, как при *импликативной* связи – не всегда, а лишь при некоторых комбинациях значений исходных признаков.

**Определение 2.** *Функциональной закономерностью*  $r$ -го ранга на множестве  $\delta k$ -знаний, ( $\delta \in \{t, \pi, v, \varphi\}$ ), называется *устойчивая* связь между  $r$ , ( $r < n$ ) признаками ОПР и некоторым  $(r+1)$ -м признаком, позволяющая по значениям признаков-*аргументов*, однозначно определить значение признака-*функции*. Понятие устойчивости связей базируется на статистических представлениях. [5]

В нашем случае, согласно **определению 1** требуется синтезировать индуктивную модель в форме *импликативной* Б $\delta k$ З как систему запретов, определяемых путем анализа ТЭД или СПОЗ. При условиях (4) об *импликативных закономерностях* исследуемых ОПР в множестве  $T_r$  будем судить по обнаруженным в ТЭД  $T_0(m, N) \subseteq T_r$  запретным связям между признаками ОПР. Сначала, с помощью доказанных ниже теорем, будем оценивать априори сложность *импликативных закономерностей* по рангу  $r$ , а затем их искать по ТЭД согласно следующему **принципу внешнего дополнения**, который в теории инженерии квантов знаний состоит в следующем.

По **определению 1**, существование *запретной* комбинации значений признаков означает, что в соответствующий *запретный интервал*  $\bar{J}$  пространства  $B^N \sim T_r$  не попадает ни один элемент из множества  $T_r$ , и, следовательно, из множества  $T_0$ , т.к.  $T_0 \subseteq T_r$ . Назовем интервал  $\bar{J}$  *пустым* относительно  $T_r$  и  $T_0$ . Чем сильнее *импликативная* связь, тем шире соответствующий  $\bar{J} \in T_r$ , а следовательно, больше вероятность проявления этой закономерности, что обнадеживает её обнаружение именно по обучающей ТЭД  $T_0(m, N)$ . Действительно, если в  $T_0(m, N)$  обнаружена *импликативная* связь между  $r$ , ( $2 \leq r \leq N$ ) признаками, которой отвечает пустой интервал  $\bar{J}$   $r$ -го ранга в пространстве  $T_r$ , то должно выполняться отношение  $\bar{J} \cap T_r = \emptyset$ , а значит, и отношение  $\bar{J} \cap T_0 = \emptyset$ . Отсюда следует,

что если  $\bar{J} \cap T_0 = \emptyset$ , то можно выдвинуть *гипотезу* о существовании *импликативной связи*  $r$ -го ранга в  $T_r$ , судя по ТЭД  $T_0(m, N)$ . Чтобы принять или отвергнуть гипотезу, нужно оценить её *достоверность*, т.е. вероятность того, что гипотеза *ошибочна*. *Достоверными* принято считать *гипотезы*, вероятность *ошибочности* которых настолько мала, что ею можно пренебречь. Поэтому, при выявлении *импликативных закономерностей* ограничимся поиском достаточно *крупных запретных интервалов*, начиная со 2-го ранга ( $r=2$ ) и пренебрегая *мелкими*, что существенно сокращает объем перебора интервалов. Очевидно, **достоверность** выдвигаемой гипотезы должна зависеть от общего числа  $N$  признаков-*столбцов* и числа  $m$  наблюдений-*строк* ТЭД  $T_0(m, N)$ , а также от ранга  $r$  искомым связей между признаками ОПР. Тогда **достоверность** гипотезы о существовании *импликативной* закономерности  $r$ -го ранга можно оценить вероятностью *отсутствия* соответствующего *пустого* интервала  $\bar{J}$ . В вычислительном плане это сложная задача, поэтому проще оценить эту вероятность математическим ожиданием  $M\{m, N, r\}$ , т.е. средним числом *запретных (пустых)* интервалов  $r$ -го ранга, возникающих в пространстве  $B^N$  (2) при **чисто случайном** формировании ТЭД  $T_0(m, N)$  из элементов множества  $T_r \subseteq B^N$ . При этом,  $m$  *строк*  $T_0(m, N)$  выбираются равновероятно из множества  $T_r$  в предположении, что отсутствуют какие-либо закономерности. Очевидно, такую гипотезу можно выдвигать лишь при *малых* значениях величины  $M\{m, N, r\}$  в интервале  $[0, 1]$ . Действительно, чем *меньше* вероятность *отсутствия* *пустого* интервала, тем *больше* **достоверность** существования соответствующей *импликативной закономерности*. На основании изложенного **принципа внешнего дополнения** для индуктивного поиска *импликативных* знаний приходим к **теореме** об оценке их сложности по рангу, исходя из объема обучающих  $\delta k$ -знаний.

**Теорема 1.** Пусть задана обучающая ТЭД  $T_0(m, N)$ , которая выборочно представляет пространство признаков  $B^N$  и состоит из  $m$  *строк-наблюдений* и  $N$  *столбцов-признаков* исследуемых классов **реальных** ОПР из множества  $T_r \subseteq B^N$ . Полагаем, что  $T_0(m, N)$  есть *случайная равновероятная*  $(m, N)$ -**выборка** элементов из  $T_r$ , представленная матричными

**δk-знаниями** 2-го уровня  $\delta k_2 T_0$ . Пусть существованию имплицативной (запретной) связи  $r$ -го ранга между некоторыми признаками ОНР отвечает событие  $S(m, N, r)$ : «некоторые интервалы  $r$ -го ранга в  $T_r \subseteq B_N$  не пересекаются с  $T_0(m, N)$  (т.е. запретные), но в реальности запретные связи между признаками отсутствуют». Тогда, оценка  $D_z$  достоверности гипотезы о существовании имплицативных закономерностей  $r$ -го ранга в  $T_r$  определяется по обучающему  $\delta k_2 T_0$  оценочной величиной  $M_S\{m, N, r\}$  для вероятности  $p_S(m, N, r)$  события  $S(m, N, r)$  в области малых её значений из интервала  $[0, 1]$  по формуле:

$$D_z = p_S(m, N, r) \leq M_S\{m, N, r\} = \frac{N! 2^{r(1-m)} (2^r - 1)^m}{r!(N-r)!}, \quad (5)$$

где  $M_S\{m, N, r\}$  – величина математического ожидания числа запретных интервалов  $r$ -го ранга, обнаруженных в ТЭД  $T_0(m, N)$  случайно;  $r_{\min} \leq r \leq r_{\max}$ ;  $2 \leq r_{\min} < r_{\max}$ ;  $r_{\max} \leq N$ .

#### Доказательство теоремы 1

Из определения 1 и содержания события  $S(m, N, r)$  очевидно вытекает суждение: «уменьшение вероятности проявления запретного интервала согласно событию  $S$  ведёт к увеличению достоверности существования соответствующей имплицативной связи». Отсюда следует правомерность использования вероятности  $p_S(m, N, r)$  события  $S$  для оценки достоверности  $D_z$ . Прежде чем определить  $p_S(m, N, r)$  заметим, что событие  $S$  сложное и представляет собой сумму простых событий  $c(m, N, r)$  типа «конкретный интервал  $r$ -го ранга не пересекается с  $T_0(m, N)$ ». Событие  $S(m, N, r)$  наступает при любом событии  $c(m, N, r)$ , соответствующем выбору любого интервала  $r$ -го ранга. Поскольку события  $c(m, N, r)$  в составном событии  $S(m, N, r)$  связаны не просто, образуя сложные взаимные пересечения интервалов, вычисление вероятности  $p_S(m, N, r)$  события  $S$  сопряжено с большими вычислительными трудностями. Поэтому, величину  $p_S(m, N, r)$  при малых её значениях оценим величиной математического ожидания  $M_S\{m, N, r\}$  числа интервалов  $r$ -го ранга, не пересекающихся со случайной ТЭД  $T_0(m, N)$ . Как известно, математическое ожидание случайной величины определяется усреднённым её значением по всем возможным реализациям с учётом их вероятностей. В нашем случае, реализациями являются

равновероятные различные булевы матрицы размером  $(m \times N)$ , отвечающие ТЭД, т.е. обучающим  $\delta k$ -знаниям  $\delta k_1 T_0$ , что упрощает вычисление  $M_S(m, N, r)$  следующим образом. Напомним, что в каждой реализации элемент “1” означает наличие соответствующего признака в данном ОНР, а “0” – его отсутствие. Обозначим множество различных булевых матриц размером  $(m \times N)$  через  $A = \{a_i\}$ , а множество всех интервалов  $r$ -го ранга в пространстве  $T_r$  – через  $B = \{b_j\}$ . В силу конечности множеств  $A$  и  $B$  их мощности определяют соответственно число различных булевых матриц данного размера (реализаций)  $|A| = 2^{mN}$  и количество всех интервалов  $r$ -го ранга  $|B| = C_N^r \cdot 2^r$ , где  $C_N^r$  – число сочетаний из общего числа  $N$  признаков по  $r$ , а  $2^r$  – количество фиксированных комбинаций значений  $r$  признаков. Очевидно, что мощность декартова произведения  $|A \times B|$  указанных множеств определяет количество таких пар  $(a_i, b_j) = (\text{матрица}, \text{интервал})$ , которые находятся в отношении непересечения, т.е. запретной связи между  $r$  признаками. Усредняя величину  $|A \times B|$  по всевозможным реализациям  $|A|$ , получим величину искомого математического ожидания  $M_S\{m, N, r\} = |A \times B| / |A|$ . Для нахождения величины  $|A \times B|$  сначала вычислим количество  $Q$  различных матриц, непересекающихся с некоторым конкретным, но произвольно выбранным интервалом  $r$ -го ранга. Предположим, что интервал образован значениями “0” первых  $r$  признаков  $x_1, x_2, \dots, x_r$ . Тогда интервал окажется пустым (запретным) в  $T_0(m, N)$ , если ни одна из  $m$  её строк не будет содержать в первых  $r$  столбцах только “0”. Очевидно, доля матриц, у которых первые  $r$  столбцов в  $m$  строках заполнены “0”, составляет  $\frac{1}{2^r}$  от общего числа матриц  $2^{mN}$ , а

$$\text{доля остатка} \quad C = \left(1 - \frac{1}{2^r}\right)^m = \frac{(2^r - 1)^m}{2^{r \cdot m}}.$$

Величину  $Q$  определим по формуле:  $Q = |A| \cdot C = 2^{mN} \cdot \frac{(2^r - 1)^m}{2^{r \cdot m}}$ . Следовательно,

$$|A \times B| = Q \cdot |B| = 2^{mN} \cdot \frac{(2^r - 1)^m}{2^{r \cdot m}} \cdot C_N^r \cdot 2^r. \quad \text{В}$$

результате деления этой величины на  $|A| = 2^{mN}$  получим

$$M_S\{m, N, r\} = \frac{(2^r - 1)^m \cdot C_N^r \cdot 2^r}{2^{r \cdot m}} = \frac{N! \cdot 2^{r(1-m)} \cdot (2^r - 1)^m}{r!(N-r)!}$$

, что совпадает с правой частью выражения

$D_z$  (5). Из описанного выше способа вычисления величины  $M_S\{m, N, r\}$  как суммы вероятностей  $p_S(m, N, r)$  событий  $c(m, N, r)$ , взятой по всевозможным интервалам  $r$ -го ранга, следует, что сумма вероятностей не может быть меньше вероятности суммы этих же событий. Формально это означает, что  $\delta_S(m, N, r) \leq M_S\{m, N, r\}$ . Результат полностью совпадает с выражением (5) для  $D_z$ .

**Теорема 1** доказана.

Величина  $M_S\{m, N, r\}$  в области малых её значений из интервала  $[0, 1]$  ограничивает сверху величину вероятности  $p_S(m, N, r)$  события  $S(m, N, r)$ . Выбор допустимого значения  $M_S^*$  для  $M_S\{m, N, r\}$  практически не сложен, т.к. величина  $M_S$  сильно зависит от ранга  $r$ . Например, при  $m=200$  и  $N=100$  значениям  $r=2, 3, 4, 5$  соответствуют с точностью до порядка значения  $M_S\{200, 100, r\}=10^{-21}, 10^{-6}, 10^2, 10^6$ . Очевидно, в данном случае при  $r_{max}=3$  и  $M_S^*=10^{-3}$  следует учитывать импликативные закономерности ранга  $r \leq 3$ . Следовательно, судя по ТЭД  $T_0(m, N)$ , гипотеза о существовании импликативной закономерности  $r$ -го ранга ( $2 \leq r \leq r_{max}$ ) как некоторого запретного  $\delta k$ -знания  $\delta k_{\nu} \bar{Y}$   $\nu$ -го уровня ( $\nu=1$ ) в  $T_r$ , принимается, если оценка её достоверности по теореме 1 при заданном значении  $M_S^*$  удовлетворяет неравенству:

$$D_z = M_S\{m, N, r\} = \frac{N! 2^{r(1-m)} (2^r - 1)^m}{r!(N-r)!} \leq M_S^* \quad (6)$$

Это означает, что при выполнении условия (6) гарантированно обеспечивается адекватность импликативной БдкЗ по рангу  $r$  ( $2 \leq r \leq r_{max}$ ) её запретов объёму  $(m \times N)$  обучающей ТЭД  $T_0(m, N)$ , используемой для синтеза данной базы знаний.

Опираясь на теорему 1, обоснуем алгоритм AZ индуктивного поиска импликативных закономерностей на основании анализа обучающей ТЭД. В практическом диапазоне значений  $m$  и  $N$  ранг  $r_{max}$  оказывается небольшим, что позволяет обнаружить все импликативные закономерности путём проверки на “запретность” интервалов ранга не выше  $r_{max}$ . Например, в случае двоичных признаков  $N$  число таких интервалов  $q_z$  определяется по формуле:

$$q_z = \sum_{r=2}^{r_{max}} = \frac{2^r \cdot N!}{r!(N-r)!}, \quad (7)$$

что допускает реализацию их перебора на ЭВМ.

Очевидно, дизъюнктивное объединение всех найденных запретных интервалов как конъюнкций комбинаций информативных признаков ОПР образует аналитическое (предикатное) описание запретной области в пространстве  $T_r$ , отвечающей импликативной БдкЗ. Отсюда вытекает следующий алгоритм AZ для индуктивного поиска импликативных закономерностей по ТЕД.

#### 4.3. Алгоритм AZ для индуктивного поиска импликативных закономерностей БдкЗ

**Вход:** обучающая ТЭД  $T_0(m, N)$  в форме матричного кванта обучающих знаний  $\delta k_2 T_0$ ;  $M_S^*$  – граничное допустимое значение оценки  $M_S$  достоверности гипотезы о существовании импликативных закономерностей в  $T_r$ ; контрольная ТЭД  $\delta k_2 T_k$  и величина допустимого риска  $R_z^*$  (при оценке качества БдкЗ величиной эмпирического риска  $R_z$ ).

**Выход:** безизбыточная БдкЗ (из простых запретов) в форме файла запретного матричного кванта знаний  $\delta k_2 \bar{\Sigma}_{BZ}$  с именем SAZ.

#### Действия

**Д1.** Определить по заданной ТЭД максимально допустимое значение  $r_{max}$  ранга импликативных закономерностей согласно заданному  $M_S^*$  по формуле (6). В случае невыполнения неравенства (6) дополнить либо заменить исходную ТЭД и повторять Д1, пока условие (6) не выполнится, после чего перейти к Д2.

**Д2.** Найти по формуле (7) все  $q_z$  импликативных связей ранга  $r$ , ( $2 \leq r \leq r_{max}$ ) путем перебора строк матрицы  $T_0(m, N)$  в обучающем  $tk_2 T_0$  и выделения отсутствующих в нем (запретных) интервалов указанных рангов; сформировать избыточную БдкЗ  $\delta k_2 \bar{\Sigma}_Z$ .

**Д3.** Преобразовать БдкЗ  $\delta k_2 \bar{\Sigma}_Z$  в безизбыточную БдкЗ  $\delta k_2 \bar{\Sigma}_{BZ}$ , состоящую из простых запретов, путем применения к  $\delta k_2 \bar{\Sigma}_Z$   $T$ -,  $T_{i-}$ , RED- и POZ-операторов манипулирования tk-знаниями, изложенных в [17]; сформировать безизбыточную БдкЗ  $\delta k_2 \bar{\Sigma}_{BZ}$ .

**Д4.** Оценить качество **БδкЗ**  $\delta k_2 \bar{\Sigma}_{BZ}$  величиною эмпирического риска  $R_Z \leq R_Z^*$  (% *ошибочных распознающих* или *прогнозных решений*) на контрольных входных ситуациях  $\delta k_2 T_k$  посредством применения процедуры «скользящего экзамена». В маловероятном случае, когда  $R_Z \leq R_Z^*$ , дополнить **БδкЗ**  $\delta k_2 \bar{\Sigma}_Z$  с помощью экспертов и перейти к **Д3**, иначе, выполнить **Д5**.

**Д5.** Сформировать рабочий файл **SAZ** для **БδкЗ**  $\delta k_2 \bar{\Sigma}_{BZ}$  допустимого качества по риску  $R_Z$ .

**Д6.** Конец.

На основе использования алгоритма **AZ** в **δРАКЗ**-методе построен оператор индуктивного вывода *импликативных*  $\delta k$ -знаний, обозначаемый **INDS** (**ТЭД**; **AZ**; **БδкЗ**) и применяемый для синтеза *импликативной* **БδкЗ**  $\delta k_2 \bar{\Sigma}_{BZ}$  по обучающим знаниям  $\delta k_2 T_0 = \text{ТЭД}$ .

#### 4.4. Оценивание сложности функциональной **БδкЗ** по рангу связей и индуктивный поиск функциональных закономерностей по обучающим $\delta k$ -знаниям

Задача состоит в нахождении оценки достоверности гипотезы о существовании *функциональных закономерностей*  $r$ -го ранга между признаками ОПР в пространстве  $T_r$ , опираясь на обнаруженные функциональные связи между  $r$  признаками в обучающей ТЭД  $T_0(m, N)$  при условиях (4). Из *определения 2* следует, что *функциональная связь* между  $r$  признаками и фиксированным  $(r+1)$ -м признаком *существует*, когда происходит событие  $f(m, N, r)$ : «некоторые строки матрицы  $T_0(m, N)$  имеют одинаковые комбинации значений в столбцах-аргументах, и они же имеют одинаковые значения в столбце-функции». Установим и для данного случая принцип внешнего дополнения.

Как и прежде, полагаем, что матрица  $T_0(m, N)$ , представляет обучающие  $\delta k$ -знания  $\delta k_2 T_0$ , заполняется нулями и единицами случайно, равновероятно и независимо друг от друга. Тогда, *достоверность* гипотезы о существовании *функциональной закономерности* в  $T_r$  по обнаруженным функциональным связям в  $T_0(m, N)$  можно оценить вероятностью  $p_F(m, N, r)$  логической суммы событий типа  $f(m, N, r)$  как *сложного*

*события*  $F$ : «в обучающем кванте знаний  $\delta k_2 T_0$  объемом  $(m \times N)$  обнаружена хотя бы одна произвольная функциональная связь  $r$ -го ранга». Событие  $F$  наступает тогда, когда в ТЭД  $T_0(m, N)$  обнаруживается произвольное событие типа  $f(m, N, r)$ . Очевидно, искомая вероятность  $p_F(m, N, r)$  суммы указанных событий  $f$  ограничена сверху суммой вероятностей отдельных событий  $f$ , которую обозначим через  $M_F(m, N, r)$ , т.е.  $p_F(m, N, r) \leq M_F(m, N, r)$ . Именно это суждение определяет суть *принципа внешнего дополнения* для *индуктивного поиска функциональных знаний* (закономерностей) по обучающим  $\delta k$ -знаниям и приводит к *теореме 2*.

**Теорема 2.** Пусть обучающая ТЭД  $T_0(m, N)$  выборочно представляет пространство признаков  $B^N$  и состоит из  $t$  строк-наблюдений и  $N$  столбцов-признаков исследуемых классов реальных ОПР из множества  $T_r \subseteq B^N$ . Пусть  $T_0(m, N)$  есть случайная равновероятная  $(m, N)$ -выборка элементов из  $T_r$ , представленная матричным  $\square$ -квантом знаний 2-го уровня  $\square k_2 T_0$ . Полагая, что проявлению функциональной связи  $r$ -го ранга между некоторыми признаками ОПР в пространстве  $B^N$  отвечает событие  $f(m, N, r)$ : «некоторые строки ТЭД  $T_0(m, N)$  имеют одинаковые комбинации значений в  $r$  столбцах-аргументах и одинаковые значения в  $(r+1)$ -м столбце-функции», рассмотрим событие  $F(m, N, r)$ : «в обучающем кванте  $\square k_2 T_0$  обнаружена хотя бы одна функциональная связь  $r$ -го ранга, которой отвечает событие  $f(m, N, r)$ ». Тогда оценка  $D_F$  достоверности гипотезы о существовании функциональных закономерностей  $r$ -го ранга в  $T_r$  определяется по обучающему  $\square k_2 T_0$  оценочной величиной  $M_F(m, N, r)$  для вероятности  $p_F(m, N, r)$  события  $F(m, N, r)$ , по формуле:

$$D_F = p_F(m, N, r) \leq M_F(m, N, r) = \frac{N! 2^{(2^r - m)}}{r!(N - r - 1)!}, \quad (8)$$

где  $M_F(m, N, r)$  – сумма вероятностей отдельных событий типа  $f(m, N, r)$ , ограничивающая сверху величину  $p_F(m, N, r)$  в области *малых* её значений из интервала  $[0, 1]$ ;  $r_{\min} \leq r \leq r_{\max}$ ;  $2 \leq r_{\min} < r_{\max}$ ;  $r_{\max} < N$ ;

#### Доказательство теоремы 2

Очевидно, доказательство теоремы сводится к определению величины  $M_F(m, N, r)$  по схеме правила произведения двух сомножителей  $K$  и  $P$ . Сомножитель  $K$

определяет количество отдельных событий  $f_i(m, N, r)$  в логической сумме событий  $F(m, N, r)$ , а сомножитель  $P$  – величину вероятности  $p_F(m, N, r)$  события  $f(m, N, r)$ , т.е. вероятности того, что выбранные заранее в ТЭД  $T_0(m, N)$   $r$  столбцов окажутся в функциональной связи  $r$ -го ранга. Определим величину  $K$ . Очевидно, что количество различных выборов на роль  $r$  аргументов – столбцов из матричного  $\delta k_2 T_0$  равно числу сочетаний  $C_N^r$ . Число выборов одного столбца-функции из оставшихся столбцов определяется разностью  $(N - r)$ . Следовательно, величина  $K$  равна:

$$K = C_N^r \cdot (N - r). \quad (9)$$

Для определения сомножителя  $P$  выберем в матричном кванте  $\delta k_2 T_0$  некоторый минор, образуемый  $r$  столбцами, претендующими на роль аргументов. Выделим в миноре строки, не совпадающие по значениям ни с одной из расположенных выше. Выделенные строки, очевидно, будут обладать различными значениями, а все остальные – совпадать по значению с какой-либо из строк. Количество выделенных строк будет не больше чем  $2^r$ , а оставшихся строк – не меньше чем  $(m - 2^r)$ . Таким образом, строки  $\delta k_2 T_0$  окажутся разбитыми на классы, содержащие строки только с одинаковыми комбинациями значений в столбцах-аргументах. Общее число таких классов не превысит  $2^r$ . Наличие функциональной связи означает, что столбец-функция принимает произвольные значения в выделенных строках, если в любой другой строке его значения будут те же, что и в выделенной из того же класса. Следовательно, при интересующем нас событии  $f(m, N, r)$  о функциональных знаниях количество строк  $\delta k_2 T_0$ , имеющих одинаковые комбинации значений в столбцах-аргументах и обладающих одинаковыми значениями в столбце-функции, не превысит величины  $(m - 2^r)$ . Отсюда вытекает соотношение, определяющее сомножитель  $P$ :

$$P = p_f(m, N, r) \leq \frac{1}{2^{(m-2^r)}} = 2^{(2^r-m)}. \quad (10)$$

Перемножив величины  $K$  (9) и  $P$  (10), получим искомое выражение для  $M_F(m, N, r)$ :

$$\begin{aligned} M_F(m, N, r) &= C_N^r \cdot (N - r) \cdot 2^{(2^r-m)} = \\ &= \frac{N!(N - r) 2^{(2^r-m)}}{r!(N - r)!} = \frac{N! 2^{(2^r-m)}}{r!(N - r - 1)!}, \end{aligned}$$

что совпадает с правой частью выражения (8). Поскольку  $M_F(m, N, r)$  оценивает величину

вероятности  $p_F(m, N, r)$  события  $F(m, N, r)$  сверху, то

$$p_F(m, N, r) \leq M_F(m, N, r).$$

Следовательно, теорема 2 доказана.

Из теоремы 2 вытекает следующее обоснование алгоритма АФ для индуктивного поиска функциональных закономерностей по обучающей ТЭД. Гипотеза о существовании функциональных закономерностей  $r$ -го ранга ( $2 \leq r \leq r_{max}$ ) в множестве  $T_r$ , судя по выборке  $T_0(m, N)$ , принимается, если оценка её достоверности по теореме 2 при допустимом значении  $M_F^*$  удовлетворяет неравенству:

$$D_F = M_F(m, N, r) = \frac{N! 2^{(2^r-m)}}{r!(N - r - 1)!} \leq M_F^*. \quad (11)$$

Это означает, что при выполнении условия (11) гарантированно обеспечивается адекватность функциональной ББКЗ по рангу  $r$  ( $2 \leq r \leq r_{max}$ ) объёму  $(m \times N)$  обучающей ТЭД  $T_0(m, N)$ , используемой при синтезе данной базы знаний. Характерно, что область значений  $r \leq 7$  удовлетворяет неравенству  $2^r < m$ , при котором оценка (11) достаточно точна, но при  $m < 2^r$  оценка оказывается слишком грубой. Однако, нас интересует область малых значений вероятности  $p_F(m, N, r)$ , т.к. именно в этом случае можно принимать гипотезы о функциональных закономерностях на основании проведенных наблюдений. Невыполнение неравенства  $2^r < m$  влечет за собой резкое увеличение значения  $M_F$ . Например, при  $m = 400$ ,  $N = 100$  и  $M_F^* = 10^{-3}$  допустимо вести поиск функциональных закономерностей, ранг которых не превышает  $r_{max} = 8$ . В практическом диапазоне значений  $m$  и  $N$  ранг  $r_{max}$  оказывается небольшим, что позволяет обнаруживать все функциональные закономерности путем выявления в  $T_0(m, N)$  события  $f$  и перебора комбинаций признаков ОПР рангом не выше  $r_{max}$ .

Эксперименты показывают, что при одном и том же объеме  $m$  обучающих знаний в  $\delta k_2 T_0$  ранги функциональных закономерностей, которые имеет смысл искать, заметно выше по сравнению с имплицативными закономерностями, т.к. функциональные связи сильнее имплицативных. Действительно, функциональная связь сводится к серии имплицативных связей, интервалы которых упорядоченно покрывают половину булева пространства  $B^N$  (20), превращая её в область запрета. Известно [5, 6], что любая функциональная связь между признаком  $x_i$  и  $t$

признаками  $x_{j_1}, x_{k_2}, \dots, x_{h_i}$  раскладывается на  $2^i$  имплицитивные связи, задаваемые трюичными  $\{0, 1, -\}$  векторами-запретами. В них комбинации значений компонент с номерами  $j_1, k_2, \dots, h_i$  попарно различны, а компонент с номером  $i$  принимает значение, обратное значению функции.

На основе использования **теоремы 2** синтезирован в [5] алгоритм **АФ** индуктивного поиска функциональных закономерностей, который применяется в **БРАКЗ-методе** при синтезе оператора индуктивного вывода функциональных **Бк**-знаний с именем **INDF(ТЭД; АФ; БкЗ)** для построения функциональной (идентификационной либо прогнозной) **БкЗ** =  $\delta k_2 \Sigma_{DF}$  по обучающим знаниям  $\delta k_2 T_0$ .

#### 4.5. Интеллектуальные информационные технологии на базе ИКЗ

На основе разработанной методологии и технологии инженерии квантов знаний уже выполнено и внедрено большое число программных систем прикладного назначения в различных производственных областях начиная с 1992 г. [5-6, 17-25]. По проблеме интеграции ИКЗ и программной инженерии «в малом» защищено 10 кандидатских диссертаций и 3 докторских за период с 1998 г. по настоящее время. Наиболее значимые интеллектуальные информационные технологии, разработанные учениками Харьковской школы исследователей по проблеме интеграции «в малом» приведены в публикациях [18-25]. Исследования и прикладные разработки по данному направлению продолжают и развиваются в настоящее время на кафедре инженерии программного обеспечения Национального аэрокосмического университета «ХАИ» имени Н.Е. Жуковского под руководством профессоров И.Б. Сироджа, И.Б. Туркина и И.В. Шостака.

#### Заключение

Из всего изложенного в работе следует вывод об актуальности в настоящее время проблемы интеграции научных и программных средств инженерии знаний и программной инженерии, а также о необходимости их совершенствования и развития параллельными курсами как «в большом» так и «в малом».

#### Литература

1. Люгер Дж. Ф. Искусственный интеллект: стратегии и методы решения сложных проблем. 4-е изд. – Пер. с англ.:– М.: Издательский дом «Вильямс», 2003.– 863 с.

2. Джонс М. Т. Программирование искусственного интеллекта в приложениях. – Пер. с англ. Осипов А. И.:– М.: ДМК. Пресс, 2006.– 312 с.

3. Джексон Питер. Введение в экспертные системы. – Пер. с англ.: Уч. пос.– М.: Издательский дом «Вильямс», 2001.– 624 с.

4. Сафронов В. О. Экспертные системы – интеллектуальные помощники специалистов. – СПб.: «Знание», 1992. – 208 с.

5. Сироджа И. Б. Квантовые модели и методы искусственного интеллекта для принятия решений и управления. – К.: Наукова думка, 2002. – 490 с.

6. Сироджа И. Б., Петренко Т. Ю. Метод разноуровневых алгоритмических квантов знаний для принятия производственных решений при недостатке и нечеткости данных. – К.: Наукова думка, 2000. – 247 с.

7. Сафонов В. О. и др. Язык представления знаний Турбо-Эксперт// Кибернетика, 1991, №5. – с. 26-35.

8. Safonov V. O., Chherepanov D.V. Java extension by production knowledge representation constructs and its implementation// Proceedings of International Conference «110<sup>th</sup> Anniversary of Radio Invention» and Regional IEEE Conference, St Petersburg, 2005. – P. 55-65.

9. Сафонов В. О. Платформа Microsoft .NET: принципы, возможности, перспективы// Компьютерные инструменты в образовании. 2004, №5. – С. 38-51.

10. Safonov V. O., Grigoryev D. A. Aspect. NET: aspect-oriented programming for Microsoft. NET in practice.// .NET Developer's Journal, 2005, № 7. – P. 16-28.

11. Zhuk J. Integration-ready architecture and design. Cambridge University Press, 2004. – 180 p.

12. Сафонов В. О., Новиков А. В., Сигалин М. В., Смоляков А. Л., Черепанов Д. Г. Интеграция методов инженерии знаний и инженерии программ: система управления знаниями Knowledge .NET.// Компьютерные инструменты в образовании. 2005, №5. – С. 52-68.

13. Web-страницы проекта Aspect. NET: <http://www.msdn.net/curriculum/?id=6219>

14. Web-страницы проекта Knowledge. NET: <http://www.knowledge-net.ru>

15. Спецификация языка KIF: <http://logic.stanford.edu/kif/dpns.html>

16. OWL. Web Ontology Overview. <http://www.w3.org/TR/owl-features/>

17. Сироджа И.Б. Математическое и программное обеспечение интеллектуальных компьютерных систем: Уч. Пособие – Харьков: ХАИ. 1992 – 100 с.

18. Петренко Т.Ю. Знаниеориентированное принятие прогнозных решений для оперативного календарного планирования и управления в

умовлях неопределенности // Авиационно-космическая техника и технология. – Харьков: ХАИ. 1998. – С. 421-429.

19. Голобродский О.Ю. Практическое применение ИВК-модели и разработка программных средств поддержки принятия решений в криминалистике // Харьков: АСУ и приборы автоматики, 1998. №107, С. 421-429.

20. Зорик В.Я., Филипковская Л.А., Третьяк В.В. Информационная технология классификационной обработки данных в проектировании техпроцессов листовой штамповки взрывом // Удосконалення процесів та обладнання обробки тиском у металургії і машинобудуванні. Зб. Наук. Праць. – Краматорськ. 2001. – С. 286-389.

21. Гордиенко Л.А., Киричук Е.П. Проблема интеллектуальной поддержки принятия технологических решений в листовой штамповке // Авиационно-космическая техника и технология. – 2004. - №3 (11). – С. 83-89.

22. Гордиенко Л.А. Интеллектуальная подсистема поддержки принятия технологических решений при проектировании оснастки в заготовительно-штамповочном производстве // Проблемы бионики: Всеукраинский научн.-техн.

сб. – Харьков ХНУРЭ. - № 57. – С. 55-60.

23. Сироджа И.Б., Россоха И.Е., Россоха С.В. Интеллектуальная информационная технология поддержки принятия диспетчерских решений// Открытые информационные и компьютерные интегрированные технологии. – Харьков: «ХАИ», 2006. – Вып. 32. – С. 114-118.

24. Сироджа И.Б., Фролова Г.А. Многокритериальное принятие решений при разработке межцеховых технологических маршрутов средствами инженерии квантов знаний. Часть I. Постановка и метод решения многокритериальной задачи о назначениях «расцеховка»// Авиационно-космическая техника и технология. – Харьков: «ХАИ», 2009. №3 (60). С. 83-95

25. Сироджа И.Б., Фролова Г.А. Многокритериальное принятие решений при разработке межцеховых технологических маршрутов средствами инженерии квантов знаний. Часть II. Производственная реализация многокритериальной задачи о назначениях «расцеховка» // Авиационно-космическая техника и технология. – Харьков: «ХАИ», 2009. №4 (61). С. 71-78

### Відомості про авторів:



**Сироджа Игорь Борисович** – д.т.н., профессор кафедры ИПО Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ»; научное направление – программная инженерия, инженерия знаний, прикладная математика  
e-mail: ro\_as\_k603@d6.khai.edu.



**Туркин Игорь Борисович** – д.т.н., профессор, зав каф. ИПО Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ»; научное направление – проектирование ПО систем реального времени.  
e-mail: energy@d4.khai.edu.



**Волобуева Лина Алексеевна** - к.т.н., доцент кафедры ИПО Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ»; научное направление – программная инженерия, знание-ориентированная поддержка принятия управляющих решений инженерия знаний.  
e-mail: linavolobueva@gmail.com.



**Соколова Евгения Витальевна** – старший преподаватель кафедры ИПО Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ»; научное направление – программная инженерия, модели вычислений, управляемые данными.  
e-mail: sev\_ai@mail.ru.



**Постернакова Вероника Альбертовна** – старший преподаватель кафедры ИПО Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ»; научное направление – программная инженерия, знаниеориентированная поддержка принятия решений. e-mail: paster108@yandex.ru.

Стаття надійшла до редакції 01.03.2010