

Федеральное государственное бюджетное учреждение науки
Институт проблем управления им. В.А. Трапезникова
Российской академии наук

На правах рукописи

Сергеев Владимир Александрович

**РАЗРАБОТКА МЕТОДОВ АНАЛИЗА И СИНТЕЗА
МЕХАНИЗМОВ КОМПЛЕКСНОГО ОЦЕНИВАНИЯ
НА ОСНОВЕ НАБОРОВ ДИСКРЕТНЫХ ДАННЫХ**

Специальность 2.3.4 – Управление в организационных системах

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель:
доктор техн. наук, доцент
Коргин Николай Андреевич

Москва – 2024

Оглавление

Оглавление.....	2
Введение.....	4
Глава 1. Формализация задачи синтеза МКО на основе обучающих наборов данных.....	11
1.1. Многокритериальные методы принятия решений	11
1.2. Методы вербального анализа решений	14
1.3. Основные подходы к экспертному синтезу МКО	16
1.4. Базовый подход к экспертному синтезу МКО.....	20
1.5. Выбор матриц свертки	22
1.6. Выбор структуры дихотомического дерева комплексного оценивания	24
1.7. Возможные подходы к решению задачи синтеза	26
1.8. Основные понятия и определения	31
1.9. Выводы по главе 1	35
Глава 2. Анализ и синтез структур МКО	36
2.1. Правила для кодирования МКО	36
2.2. Унитарное представление механизмов комплексного оценивания	37
2.3. Группы эквивалентности	38
2.4. Алгоритмы поиска числа групп эквивалентности	42
2.5. Таблица ветвей.....	44
2.6. Методы генерации структур.....	47
2.7. Выводы по главе 2	50
Глава 3. Анализ и синтез матриц МКО	52
3.1. Синтез матриц МКО на основе полного набора данных.....	52
3.2. Синтез полного набора матриц для заданной структуры МКО.....	55
3.3. Синтез матрицы для всех поддеревьев данного разбиения пространства входных переменных	60
3.4. Анализ матриц МКО	70
3.5. Выводы по главе 3	72
Глава 4. Прикладные задачи анализа и синтеза МКО	74
4.1. Модули программного комплекса по синтезу и анализу МКО	74
4.2. Общий алгоритм анализа и синтеза МКО.....	79
4.3. Примеры анализа и синтеза для полного набора данных – булевы функции	82
4.4. Примеры анализа и синтеза для неполных наборов данных.....	85
4.5. Прогнозная система основанная на моделях МКО	99

4.6. Выводы по главе 4	100
Заключение	102
Список литературы	103
Приложение А	114
Приложение Б	117
Приложение В.....	118
Приложение Г	119
Приложение Д.....	120

Введение

Актуальность и степень разработанности темы исследования.

Механизм комплексного оценивания (МКО) – это инструмент для оценки и ранжирования в организационных и производственных системах. МКО позволяет получать комплексную оценку сложного объекта путем свертки рассматриваемых показателей, характеризующих объект. Основные компоненты механизма комплексного оценивания – полное бинарное дерево на l именованных листьях и матрицы логической свертки. Механизмы комплексного оценивания были введены для управления и контроля в организационных и производственных системах в Советском Союзе в начале 80-х годов прошлого века, например, система АККОРД для электронной промышленности, (см., например, [19, 20, 28, 57]) и используются по сей день (см., например, [1, 4, 6, 15, 96]). В настоящее время МКО используются в целом спектре прикладных задач. Например, при оценке уровня инновационного развития отрасли, оценке проектов НИОКР, оценке программ регионального развития, в системах управления развитием организации, при оценке проектов в сфере малого и среднего предпринимательства, при оценке экологических рисков, оценке эффективности управления коммерческой и жилой недвижимостью, в механизмах поддержки принятия решений и других задачах организационной деятельности. МКО могут быть отнесены к классу так называемых методов вербального анализа решений (ВАР) для неструктурированных проблем (см., например, [41, 98]) и позволяют их структурировать, посредством синтеза агрегированного показателя оценки на основе качественно интерпретируемой структуры попарно агрегируемых исходных показателей разной природы (см., например, [26, 51, 64]). МКО предназначены для порядкового ранжирования или классификации с заранее определенным числом классов конечного набора многокритериальных альтернатив. Помимо выполнения самостоятельной роли агрегации оценки сложного объекта в сфере механизмов поддержки принятия решений (см., например, [41, 48, 49]). МКО может использоваться в цикле управления (см., например, [44, 46]) как элемент блока контроля. Также представляется актуальной задача управления входными параметрами для достижения необходимого значения комплексной оценки [36].

Основной подход к синтезу параметров МКО предполагает итеративное взаимодействие с лицами, принимающими решения (см., например, [6, 9, 15, 14, 27]). Однако в настоящее время существует запрос на разработку обучающих процедур для МКО, которые довольно обычны для методов синтеза в области искусственного интеллекта. Подобная задача исследовалась лишь для отдельных частных случаев (см., например, [14, 66, 33]). В этой работе приводится несколько подходов к синтезу структур и матриц – двух основных составляющих МКО. Характерной особенностью предлагаемых методов синтеза матриц является использование обучающего набора

примеров, в том числе и с применением довольно популярного в настоящее время подхода из области искусственного интеллекта – представление категориальных оценок на основе унитарного кодирования (см., например, [88, 134]). Говоря об обучающих наборах важно отметить, что в данном исследовании мы работаем с дискретными наборами данных. Это значит, что содержащаяся в наборе информация может принимать только конкретные значения из конечного набора значений, в отличие от непрерывных данных. Также рассматривается актуальный вопрос выбора структуры полного бинарного дерева МКО.

Объект исследования. Механизмы комплексного оценивания.

Предмет исследования. Методы анализа и синтеза механизмов комплексного оценивания.

Целью работы является разработка методов анализа и синтеза механизмов комплексного оценивания на основе наборов дискретных данных.

Для достижения цели требуется решить следующие **задачи**.

1. Провести анализ существующих методов анализа и синтеза механизмов комплексного оценивания.
2. Разработать методы анализа и синтеза множества рассматриваемых структур полных бинарных деревьев.
3. Разработать метод синтеза матриц механизма комплексного оценивания заданной структуры на основе наборов дискретных данных.
4. Разработать общий алгоритм и реализующее его программное обеспечение для анализа и синтеза механизмов комплексного оценивания на основе наборов дискретных данных.

Научная новизна.

1. Впервые предложены методы записи, синтеза и сокращения множества рассматриваемых структур полных бинарных деревьев, позволяющие выбирать перспективные структуры для задачи синтеза матриц.
2. Впервые предложены методы синтеза матриц МКО заданной размерности для случаев полного и неполного наборов дискретных данных.
3. Предложен общий алгоритм анализа и синтеза МКО. Алгоритм разработан с использованием предложенных методов анализа и синтеза, и впервые реализован в виде комплекса программ.

Достоверность полученных научных результатов. Достоверность полученных результатов обеспечивается использованием строгих математических моделей, сравнением аналитических результатов с результатами работы программного кода предложенных методов. Разработанные методы и алгоритмы, содержащиеся в методике анализа и синтеза механизма комплексного оценивания, успешно прошли апробацию, что подтверждается публикацией результатов работы в рецензируемых изданиях и актами о внедрении прикладных результатов работы.

Теоретическая и практическая значимость результатов диссертационной работы. Теоретическая значимость заключается в развитии методов анализа и синтеза МКО. Метод анализа групп эквивалентности позволил создать эффективный инструмент синтеза МКО на полных наборах данных. А в случае с неполными наборами данных перейти от рассмотрения всего множества полных двоичных деревьев на именованных листьях, число деревьев в котором растет как двойной факториал от числа входных параметров, к рассмотрению таблицы разбиений реализующихся ветвей. Предложены методы синтеза матриц МКО на основе таблицы обучающих примеров. Исследованы вопросы реализуемости булевых функций трех и четырех переменных на основе МКО.

Практическая значимость работы заключается в создании методов и программного обеспечения, которые позволяют синтезировать МКО на основе набора дискретных данных. Разработанные методы и программное обеспечение позволяют решать задачи синтеза матриц МКО, используя обучение на основе таблицы примеров. Они также позволяют автоматически создавать механизмы оценки без необходимости привлечения экспертов для выбора структуры МКО, формирования матриц МКО и согласования результатов работы экспертов. Разработанные инструменты анализа позволяют получать древовидное представление набора данных, а также производить: структурный анализ вклада отдельных переменных в итоговое значение, анализ влияния отдельных переменных и групп переменных на итоговое значение агрегированного показателя, анализ монотонности обучающего набора дискретных данных, выделение целевых примеров для дальнейшего синтеза модели.

Основные положения, выносимые на защиту.

1. Методы записи, синтеза и сокращения множества рассматриваемых структур полных бинарных деревьев: метод поиска числа групп эквивалентности, метод таблицы ветвей, методы генерации структур. Предложенные методы позволяют сформировать и выбрать структуры, перспективные для синтеза МКО при разработке математических моделей оценки организационных систем.

2. Методы синтеза матриц МКО для наборов дискретных данных: методы синтеза матриц рассматриваемой структуры МКО для случаев полного и неполного наборов данных, декомпозиционный метод синтеза для неполного набора данных. Предложенные методы используются для синтеза моделей оценки организационных систем, которые могут использоваться для расчета критериев эффективности, качества и надежности организационных систем.

3. Общий алгоритм и реализующее его программное обеспечение, позволяющее решать прикладные задачи анализа и синтеза МКО в организационных системах. Разработанное программное обеспечение может быть использовано в составе систем управления и механизмов принятия решений в организационных системах.

Соответствие пунктам паспорта специальности 2.3.4. Управление в организационных системах.

1. Положение 1 соответствует п. 5 «Разработка методов получения данных и идентификации моделей прогнозирования и управления организационными системами на основе ретроспективной, текущей и экспертной информации».

2. Положение 2 соответствует п. 2 «Разработка математических моделей и критериев эффективности, качества и надежности организационных систем».

3. Положение 3 соответствует п. 4 «Разработка информационного и программного обеспечения систем управления и механизмов принятия решений в организационных системах».

Методы исследования. Для решения задач, поставленных в диссертации, использовались методы комбинаторики, теории активных систем, дискретной оптимизации, машинного обучения. При разработке программного обеспечения использовались методы объектно-ориентированного программирования.

Апробация результатов. Основные научные и прикладные результаты докладывались и обсуждались на международных и всероссийских научных конференциях: IV Российский экономический конгресс (РЭК-2020 г., Москва), Artificial Intelligence for Sustainable and Resilient Production Systems (APMS 2021 г., Nantes, France, online), 20th IFAC Conference on Technology, Culture and International Stability (2021 г., ИПУ РАН, г. Москва), XVII Всероссийская школа-конференция молодых ученых «Управление большими системами» (2021 г., ИПУ РАН, г. Москва), XXII Апрельская международная научная конференция по проблемам развития экономики и общества (2021 г., ВШЭ, г. Москва, online), XVIII Всероссийская школа-конференция молодых ученых «Управление большими системами» (2022 г., ЮУрГУ, г. Челябинск), Математическая теория оптимизации и исследование операций (2022 г., Петрозаводск), Научно-практическая конференция «Технологическое развитие авиастроения: глобальные тенденции и национальные интересы России» 2022; на общемосковских семинарах: «Теория управления организационными системами» и «Математические методы анализа решений в экономике, бизнесе и политике»; на семинаре Пермского научно-образовательного центра проблем управления, а также на научных семинарах ИПУ РАН.

Связь с планами научных исследований.

Работа выполнялась при поддержке гранта Российского научного фонда 17-78-20047.

Внедрение результатов работы.

1. Методы синтеза механизмов комплексного оценивания внедрены в процедуру оценки дизайн-решений в рамках проекта «Арктический дизайн: методы технической эстетики в освоении и развитии территорий Российского Севера», ФГАОУ ВО «Национальный исследовательский Томский государственный университет».

2. Методы синтеза механизмов комплексного оценивания внедрены в процедуру синтеза МКО для контура системы управления робототехническими комплексами (экспериментального стенда) в рамках реализации научно-исследовательской работы «ВИАС-Модель-2022» ФГБУ «НИЦ «Институт имени Н.Е. Жуковского».

3. Методы синтеза МКО внедрены для синтеза системы комплексного оценивания платежеспособности российских строительных компаний в ООО «Пермский центр поддержки принятия решений».

Личный вклад. Все основные результаты диссертационной работы получены автором самостоятельно. Комплекс программ разработан автором.

Публикации. Основные положения и выводы диссертационного исследования нашли отражение в 9 научных работах. По результатам опубликованы две статьи в рецензируемых научных изданиях по специальности 2.3.4, относящиеся к категории К1 Перечня ВАК [16, 53], одна работа в журнале, индексируемом в международных базах данных, перечень которых определен в соответствии с рекомендацией ВАК [126], шесть статей опубликованы в прочих изданиях [36, 38, 52, 55, 79, 95], а также получено свидетельство о государственной регистрации программы для ЭВМ [54].

Статьи в журналах/сборниках из Перечня ВАК (категория К1)

- Бурков В.Н., Сергеев В.А., Коргин Н.А. Идентификация механизмов комплексного оценивания на основе унитарного кода // Управление большими системами: сборник трудов. – 2020. – № 87. – С. 67–85.
- Сергеев В.А. Синтез механизмов комплексного оценивания на основе разделительной декомпозиции // Проблемы управления. – 2022. – № 6. – С. 3–13.

Статьи в журналах, индексируемых в международных базах данных, перечень которых определен в соответствии с рекомендацией ВАК

- Sergeev V.A., Korgin N.A. Identification of Integrated Rating Mechanisms As An Approach To Discrete Data Analysis // IFAC-PapersOnLine. – 2021. – Vol. 54, Issue 13. – P. 134–139.

Статьи в других рецензируемых научных изданиях

- Korgin N., Sergeev V. Identification of Integrated Rating Mechanisms on Complete Data Sets / Proceedings of IFIP WG 5.7 International Conference "Advances in Production Management Systems" (APMS 2021) (Artificial Intelligence for Sustainable and Resilient Production Systems). – 2021. – Vol. 630. – P. 610–616.
- Burkov V.N., Korgin N.A., Sergeev V.A. Identification of Integrated Rating Mechanisms as Optimization Problem // Proceedings of 13th International Conference "Management of large-scale system development" (MLSD). – 2020. – P. 1–5.

В сборниках трудов конференций

- Сергеев В.А. Идентификация механизмов комплексного оценивания с применением разделительной декомпозиции // Труды 18-ой Всероссийской школы-конференции молодых ученых «Управление большими системами» (УБС'2022, Челябинск). – 2022. – С. 546–552.
- Коргин Н.А., Кравчук С.Г., Сергеев В.А. Проблемы согласования интересов в проектах развития инфраструктуры, обеспечивающей функционирование легкого внедорожного электротранспорта / Материалы 14-й Мультиконференции по проблемам управления (МКПУ-2021, Дивноморское, Геленджик). – 2021. – Т. 2. – С. 155–157.
- Коргин Н.А., Сергеев В.А. Выбор структур при решении задач идентификации механизмов комплексного оценивания для полных наборов данных / Материалы 14-й Мультиконференции по проблемам управления (МКПУ-2021, Дивноморское, Геленджик). – 2021. – Т. 2. – С. 158–160.
- Сергеев В.А. Коргин Н.А. Исследование чувствительности дискретной функции для решения задачи синтеза МКО на основе дискретных наборов данных / Материалы 16-й Мультиконференции по проблемам управления (МКПУ-2023, Волгоград). – 2023. – Т. 2. – С. 345–348.

Свидетельства о государственной регистрации программ для ЭВМ

- Сергеев В.А., Коргин Н.А. Сборщик полных бинарных деревьев на именованных листьях на основе данных анализа групп эквивалентности: Свидетельство о государственной регистрации программы для ЭВМ № 2023618561 РФ; Зарег. 26.04.2023.

Структура и объем работы.

Диссертация состоит из введения, четырех глав, заключения, библиографического списка из 143 наименований и пяти приложений. Основная часть изложена на 120 страницах, содержит 16 рисунков и 32 таблицы.

Первая глава посвящена рассмотрению механизма комплексного оценивания как одного из методов многокритериального оценивания. Делается обзор широко известных методов многокритериального оценивания, обозначается место механизмов комплексного оценивания в общей классификации. Освещается экспертный подход к синтезу механизмов комплексного оценивания, а также затрагиваются основные трудности, возникающие при комплексном оценивании. Дается определение механизма комплексного оценивания, а также через концепцию реализуемости заданного обучающего набора формулируется задача синтеза МКО на основе набора дискретных данных [16, 36, 53, 79].

Во второй главе исследуется вопрос анализа и синтеза структур механизмов комплексного оценивания. В этой главе предлагаются правила для кодирования структур МКО, а также вводится инструмент унитарного кодирования, который позволяет осуществлять операции

умножения на матрицах логической свертки как обычные матричные операции [38, 55]. Приводится пример формулирования задачи синтеза матриц МКО как оптимизационной проблемы [16, 79]. В разделе 2.3 предлагается метод анализа групп эквивалентности (АГЭ), приводятся утверждения о существовании МКО, реализующих рассматриваемый набор дискретных данных, в рамках заданной шкалы [53, 95]. В разделе 2.4 предлагаются алгоритмы поиска числа групп эквивалентности. В разделе 2.5 предлагается компактная, поуровневая запись комбинаций листьев МКО, названная «таблица ветвей» [53]. Раздел 2.6 посвящен методам генерации структур МКО.

В третьей главе описываются предложенные подходы по анализу и синтезу матриц МКО. В разделе 3.1. предлагается метод синтеза матриц МКО для заданной структуры полного двоичного дерева на полном наборе данных [95]. В разделах 3.2 – 3.3 предлагаются методы синтеза МКО для неполных наборов данных, приводятся необходимые утверждения и следствия. В разделе 3.2 приводится метод синтеза всех матриц для конкретной структуры МКО [16, 79], а в разделе 3.3 предлагается метод синтеза МКО с применением разделительной декомпозиции, позволяющей синтезировать матрицы, общие для целого подмножества полных бинарных деревьев, потенциальных структур агрегирования для синтезируемой МКО [52, 53]. Для предложенных методов рассматриваются примеры синтеза МКО. В разделе 3.4 предложен алгоритм анализа матриц МКО.

Четвертая глава посвящена прикладным аспектам применения полученных в главах 1-3 теоретических результатов. В разделе 4.1 приводится описание работы модулей программного комплекса для синтеза и анализа МКО. В разделе 4.2 приводится общий алгоритм анализа и синтеза МКО. В разделе 4.3 демонстрируется работа подхода по синтезу всех матриц заданной структуры МКО, на примерах синтеза МКО для булевых функций [79]. На примере решения прикладных задач синтеза МКО в разделе 4.4 демонстрируется подход к анализу и синтезу МКО [126]. Раздел 4.5 посвящен вопросу синтеза прогнозной системы на основе результатов предсказания успешно обученных моделей МКО.

В заключении приведены основные результаты работы.

В приложении А приводится неполный набор данных для задачи из главы 4. В приложениях Б, В, Г приводятся акты о внедрениях. Приложение Д содержит свидетельство о государственной регистрации программы для ЭВМ.

Глава 1. Формализация задачи синтеза МКО на основе обучающих наборов данных

1.1. Многокритериальные методы принятия решений

Проводя исследование, посвященное синтезу механизмов комплексного оценивания, следует обратить внимание на то, что уже сделано по данной теме, и в частности, как другие исследователи классифицируют МКО. Примерно с середины прошлого века [106, 94] получили развитие многие методы ранжирования, оценки и сортировки, связанные с разработкой вычислительных и математических инструментов для поддержки принятия решений. Многие из них со временем оформились в отдельные инструменты в рамках общего направления – методы принятия решений по множеству критериев, или, иначе говоря, мультикритериальное принятие решений MCDM (multiple criteria decision making). Как и в любой сформировавшейся области, по MCDM написано много монографий, посвященных теории и базовым методам, а также обзорных работ по методам MCDM [102, 74, 83, 138, 125, 98, 114, 42]. В книге М. Koksalan [94] представлена краткая история развития методов MCDM. Начиная с первых упоминаний о MCDM еще до 1960-х годов, а затем за каждые десять лет, дается обзор основных достижений, вплоть до 2010-х годов. J. Figueira [83] прекрасно рассказывает историю становления области MCDM с точки зрения связей со значимыми работами из других областей знаний. Например, он упоминает о функции полезности и теории социального выбора, указывает на связи метода Борда и подходом к ранжированию в MCDM, отмечает, что концепция доминирования используемая в MCDM связана с Парето, а также обращает внимание на связи MCDM и теории игр. Официальной точкой появления области MCDM он называет конференцию 1972 года “Multiple Criteria Decision Making”, организованную М. Zeleny в университете Южной Каролины. Несмотря на вклад в MCDM из множества областей знаний, принято считать, что исторически методы MCDM оформились в рамках дисциплины исследование операций (ИО) как методы принятия решений по множеству часто противоречивых критериев [83, 137].

Как было сказано выше, одной из отличительных черт методов многокритериального оценивания является наличие противоречивых параметров, по которым оцениваются альтернативы. Т.е. высокая оценка альтернативы по одному параметру сопровождается низкой оценкой по другому, что встречается в довольно большом количестве жизненных ситуаций. Дополнительно можно отметить, что многие методы MCDM ориентированы на проблемы с которыми лицо принимающее решение (ЛПР) сталкивается впервые, либо которые обладают определенными важными особенностями, относительно тех проблем, с которыми ЛПР уже приходилось работать

[41]. Помимо перечисленных выше особенностей, связанных с критериями, многокритериальные проблемы принятия решений отличаются от проблем исследования операций отсутствием четкой и объективной структуры рассматриваемой проблемы [98]. С точки зрения данного исследования также важно отметить, что в рамках MCDM существуют методы работающие как с количественными, так и с качественными критериями [98, 114, 42, 99, 78]. Нужно сделать оговорку, что представление данных в табличном виде само по себе такой структурой не является, хотя отдельные авторы и используют термин «структурированный» по отношению к табличным данным [122]. Поскольку очевидной лучшей альтернативы часто не существует, предпочтения ЛПР или группы экспертов обычно используются для сравнения и оценки преимуществ различных факторов и их возможных значений. Существует альтернативный подход к формированию правил для принятия решений, основанный на механизмах обучения по прецедентам, о нем речь пойдет в разделе, посвященном обучению.

Существует способ группировки методов, основанный на типе проблемы принятия решения, для которой они используются.

- Методы ранжирования. Эти методы используются для ранжирования альтернатив на основе их предпочтения по множеству критериев.
- Методы выбора. Эти методы используются для выбора наилучшей альтернативы или подмножества лучших альтернатив из набора альтернатив на основе нескольких критериев.
- Методы сортировки. Эти методы присваивают каждой из альтернатив какую-то из заранее определенных категорий. Если множество заданных классов может быть упорядочено, то речь о задаче упорядоченной сортировки, если нет, то говорим о проблеме классификации.

Помимо классификации по типу решаемой проблемы, можно разделить методы принятия решений по множеству критериев по типу переменных:

Один из способов группировки методов основан на типе информации, которую они используют.

- Количественные методы. Эти методы используют числовые значения для представления критериев и альтернатив и включают такие методы, как метод аналитической иерархии АНР (Analytic Hierarchy Process), многокритериальная теория полезности MAUT (multiple attribute utility theory), модель взвешенной суммы и метод предпочтения порядка по сходству с идеальным решением.
- Качественные методы: эти методы используют естественный язык или лингвистические переменные для представления критериев и альтернатив и включают такие подходы, как вербальный анализ решений или методы на основе нечеткой логики.

— Гибридные методы. Эти методы используют как количественную, так и качественную информацию для представления критериев и альтернатив. Здесь такие методы, как: нечеткий АНР, нечеткий МАУТ и нечеткий TOPSIS.

Некоторые из широко известных методов.

1. Метод аналитической иерархии АНР: используется иерархическая структура для организации и оценки различных альтернатив на основе множества критериев.
2. Многокритериальная теория полезности МАУТ: этот метод используется для оценки альтернатив на основе нескольких критериев путем присвоения числовых значений критериям и расчета общего значения полезности для каждой альтернативы.
3. Метод предпочтения порядка по сходству с идеальным решением TOPSIS: используется для определения альтернативы, которая наиболее близка к идеальному решению и наиболее далека от отрицательного идеального решения.
4. Разработка индексов парного сравнения альтернатив РИПСА или исключение и выбор, отражающие реальность: этот метод используется для оценки альтернатив на основе нескольких критериев путем сравнения каждой альтернативы с порогом «соответствия» и «несоответствия».
5. PROMETHEE: этот метод используется для оценки альтернатив на основе нескольких критериев путем ранжирования каждой альтернативы на основе ее «чистого потока» предпочтений.
6. Модель взвешенной суммы WSM, простое аддитивное взвешивание SAW и модель взвешенного продукта WPM: эти методы используются для оценки альтернатив на основе нескольких критериев путем взвешивания критериев и суммирования или перемножения, для SAW и WPM соответственно, взвешенных значений для каждой альтернативы.
7. Вербальный анализ решений ВАР. Этот подход касается проблем принятия решений, когда информация представлена на естественном языке, а не числовыми значениями.
8. Нечеткое многокритериальное принятие решений FMCDА: этот метод используется для оценки альтернатив на основе нескольких критериев при наличии неопределенности или неточности в данных.
9. Метод порогового агрегирования: для каждого критерия устанавливается порог, по которому объекты разделяются на положительные и отрицательные. Если значение критерия превышает установленный порог, то объект получает положительную оценку, в противном случае - отрицательную.

Существует литература, посвященная сравнению и критике различных методов, см. например, [124, 77, 91, 59, 50, 109, 108], а также по развитию уже широко применяемых методов [90,

110, 103]. К настоящему моменту сотни публикаций были изданы по теме MCDM, многие из них посвящены применению разработанных методов многокритериального анализа для решения актуальных проблем [133, 118, 85, 87, 117]. Например, методы MCDM применяются в таких областях, как: выбор путей для прокладки трубопроводов [85], оценка устойчивого развития городов [133], оценка состояния водных ресурсов [80], оценка влияния сельскохозяйственных практик на среду [87], оценка программных продуктов [110] и многие другие приложения. Применение методов многокритериального оценивания может быть актуально практически в любой сфере, где принимаются комплексные решения с противоречивыми критериями. Вместе с развитием вычислительных технологий появляются электронные продукты, ориентированные на помощь ЛПР [72]. Н. Weistroffer в своей работе [132], приводит обзор 69 пакетов для мультикритериального анализа решений, как например: 1000Minds, 4eMka2/jMAF, ACADEA, Accord, Analytica Optimizer и многие другие.

1.2. Методы вербального анализа решений

Среди методов MCDM для неструктурированных проблем выделяется группа, характерной особенностью которых является ориентированность на особенности взаимодействия с ЛПР [98, 89, 115, 78, 9, 26, 42]. Как изложено в [42], человек как биологический вид имеет характерные особенности, которые необходимо учитывать в процессе принятия решений. К ним относится факт, что люди обычно не имеют четко определенной политики или правила принятия решений: правила разрабатываются в процессе проб и ошибок. Кроме того, у человека ограниченный объем кратковременной памяти, что вызывает необходимость сосредотачиваться на рассмотрении некоторого подмножества вариантов, вместо рассмотрения всех вариантов одновременно. Эта особенность является причиной распространенной стратегии принятия решений – поиск доминирующей альтернативы или группы доминирующих альтернатив в области доступных альтернатив. Также распространен подход, при котором из рассмотрения исключаются заведомо проигрышные, доминируемые альтернативы, чтобы сократить количество альтернатив до обозримого числа, прежде чем использовать более сложные стратегии сравнения. Довольно распространен подход, когда ищут удовлетворительное, а не оптимальное решение, устойчивое к изменениям внешних факторов. Кроме того, многие люди склонны сводить к минимуму когнитивные усилия, необходимые для поиска решения, и могут менять свои стратегии на те, которые требуют меньше умственных усилий [41, 42]. Они также склонны полагаться на более простые когнитивные операции, такие как сложение и простое сравнение небольшого числа переменных.

Во многих ситуациях для ЛПР является довольно естественным формулировать свои оценки некоторыми словесными выражениями. L. Zadeh [136] предложил именовать словесные выражения значениями лингвистической переменной, что позволяет работать с критериями, оценки по шкалам которых являются словами, как лингвистическими или вербальными. Для таких проблем и предназначена методология вербального анализа решений [98, 14, 78].

Основные характеристики методов вербального анализа решений [41, 109].

- Во-первых, качественные измерения позволяют получить описание неструктурированной проблемы, близкое к реальному. Это существенно упрощает анализ для ЛПР, так как он может привычно работать с понятиями в рамках вербальной шкалы, не тратя усилия на попытку оценить численно стоимость того или иного варианта.
- Во-вторых, использование способов построения решающего правила, соответствующих возможностям человеческой системы переработки информации, позволяет обосновать методы с психологической точки зрения.
- В-третьих, специальные процедуры проверки информации на непротиворечивость обеспечивают надежность получаемой информации и создают для ЛПР возможности постепенной выработки решающего правила.
- В-четвертых, возможность получения объяснений того, как работает механизм, почему предпочтение отдано конкретной альтернативе, увеличивает шансы на успешное практическое применение.

Подытоживая особенности методов ВАР и не углубляясь в детали, можно сказать, что лицо, принимающее решения, является центральной фигурой в принятии решений на основе множества критериев. Выявление предпочтений ЛПР, должно учитывать особенности поведения человека в процессах принятия решений.

Существуют работы рассматривающие основные сходства и различия методов ВАР [98, 109] по сравнению с другими основными группами методов в многокритериальном анализе решений. И многокритериальная теория полезности, и ВАР могут использовать словесные описания для оценки альтернатив, чтобы получить порядок предпочтения. Конечная цель состоит в том, чтобы предоставить универсальное правило упорядочения, которое можно применять к любому набору реальных альтернатив. Например, метод аналитической иерархии хотя и может принимать информацию о предпочтениях в порядковой форме, но для формирования оценки преобразует ее в числовые значения для дальнейшего анализа. АНР, как и методы опережения, работает только с реальными альтернативами, и предпочтения, полученные как результат процесса оценивания, могут потребовать пересмотра с добавлением новых альтернатив.

Основные методы ВАР.

- Метод ЗАПРОС (замкнутые процедуры у опорных ситуаций) используется для ситуаций, когда необходимо ранжировать достаточно большое число альтернатив, причем набор альтернатив может меняться при сохранении решающих правил.
- Метод ОРКЛАСС (ординальная классификация) предназначен для задач определения соответствующего класса для каждой реальной альтернативы из K классов C_1, C_2, \dots, C_K . Классы являются порядковыми в том смысле, что все альтернативы в классе C_1 предпочтительнее, чем в классе C_2 , и т.д.
- Метод ПАКОМ, также называемый ПАРК (парная компенсация) или АСТРИД, ориентирован на выбор наилучшей альтернативы из небольшого их числа. Это метод, где процесс ориентирован на попарное сравнение реальных альтернатив по принципу компенсации, где недостатки одной альтернативы уравниваются недостатками других альтернатив.
- Метод комплексного оценивания. МКО предназначен для порядкового ранжирования или классификации с заранее определенным числом классов конечного набора многокритериальных альтернатив.

Существует литература [98], посвященная развитию указанных методов: так ОРКЛАСС III, а также ШАГ- ЗАПРОС развивают метод ЗАПРОС так, чтобы можно было сравнивать не только отдельные значения по разным критериям, но и пары значений критериев. UniCombos представляет собой компьютеризированную систему, которая основана на идеях ZAPROS и является более удобной в использовании. На основе ОРКЛАСС реализованы методы ЦИКЛ и ДИФКЛАСС. Метод СНОД разработан на основе ПАКОМ и призван сделать процесс, предложенный в ПАКОМ, более эффективным. Существуют исследования по сравнению методов ВАР [100, 101, 99, 24], а также посвященные применению методов ВАР [107, 101, 98]. Методы МКО будут более детально рассмотрены в следующем разделе.

1.3. Основные подходы к экспертному синтезу МКО

Итак, механизмы комплексного оценивания принято относить к методам вербального анализа решений. В русскоязычной литературе встречаются публикации, в которых под термином «механизм комплексного оценивания» имеются ввиду методы агрегации критериев с их весами в интегральную оценку, например с помощью линейной аддитивной свертки. Так что с точки зрения классификации в области методов MCDM такие методы ближе к методам WSM [139], чем к методам вербального анализа. В данной работе мы используем понятие МКО строго из теории управления организационными системами, см. определение 1.

МКО наследуют от методов ВАР основные особенности. Во-первых, это использование переменных в качественных шкалах, вызванное трудностями или невозможностью выражения параметров оцениваемого объекта в количественных шкалах. Во-вторых, ориентированность на работу с ЛПР или группой экспертов, опыт которых используется для формирования решающего правила. Задача комплексного оценивания (КО) близка по своей природе к многокритериальным задачам принятия решений [83, 106], однако имеет важные отличия, так как в работах по MCDM, критериальные свойства часто считаются заданными, а связи между ними считаются связями верного типа, т.е. все показатели агрегируются вместе. Что, по сути, равносильно отсутствию иерархии параметров объектов. На практике же критерии часто бывают взаимосвязаны, соподчинены или распадаются на группы [47, 26]. Проработка вопросов выявления и выработки критериальных свойств и установления типа их взаимосвязи в рамках задачи МКО выделяет ее среди многокритериальных задач принятия решений. Сложность выбора структуры указывает на важность процедур выявления существенных критериальных свойств. В начале 1970-х годов американский математик Томас Саати разработал процедуру поддержки принятия решений, которую назвал метод анализа иерархий АНР [83, 102]. Этот многокритериальный метод занимает особое место, благодаря исключительно широкому распространению и активному применению, особенно в США. Его отличительной особенностью является использование матриц парных сравнений критериев. При этом заполнение производится на основе количественного подхода. Такие матрицы заполняются экспертами или лицами, принимающими решения ЛПР, и формализуют относительную попарную предпочтительность значения критерия для рассматриваемых альтернатив. Считается [41, 42], что определение предпочтительности альтернатив путем парных сравнений вызывает меньше затруднений у ЛПР в сравнении с необходимостью напрямую задать вес для каждой из альтернатив, как, например, в методе простого многокритериального оценивания Эдвардса ПТМО [82]. Окончательное значение агрегированной оценки получается в результате взвешенной суммы значений критериев, что, в частности, исключает возможность учесть часто присутствующее взаимное влияние критериев друг на друга. Более подробно о недостатках АНР, см., например, в работах [50, 59]. Достоинством бинарной структуры, используемой в МКО, является то, что она позволяет, расположив матрицы свертки в своих узлах, получить решающее правило, учитывающее взаимосвязи параметров оцениваемого объекта. Использование матриц свертки для формирования КО, таким образом, представляет собой более гибкий инструмент, а также естественным образом позволяет передать существующую иерархию критериев в структуре дерева свертки. Сложность в необходимости задания матриц свертки меньше, чем сложность формирования таблиц попарных сравнений для всех возможных комбинаций критериев, но вопрос выбора структуры дерева, заслуживает отдельного рассмотрения.

Таким образом, еще одна отличительная черта МКО – использование принципа бинарности, т.е. редукция многокритериальной задачи к последовательности двухкритериальных задач, для которых ЛПР дает надежную (устойчивую) информацию [27, 25, 10].

Известно, что построенный механизм комплексного оценивания, представляет собой некоторую функцию, с количеством переменных, равным числу параметров оцениваемого объекта. Соответственно, допустимо представление бинарной структуры как некоторой декомпозиции целевой функции $f(l_1, l_2, \dots, l_n) = \varphi_1(l_1, \varphi_2(l_2, \dots, \varphi_{n-1}(l_{n-1}, l_n)))$ (см. например, [52, 53]). Вопрос декомпозируемости дискретных функций исследовался В.А. Готовым и В.В. Павельевым [26] с точки зрения освещения применения декомпозиции для построения критериально-целевой структуры. В.Н. Бурков и И.В. Буркова с коллегами исследовали дихотомическое представление функции [21, 12], применение метода разделительной декомпозиции на основе табличных данных для построения МКО исследовано в [52, 53].

Основные подходы к экспертному синтезу МКО.

1. Метод бинарной композиции. Основу метода (см., например, [24, 26]) составляют три шага: а) выявление критериальных свойств, б) попарное объединение критериальных свойств, в) построение матриц свертки. Предполагается, что эксперты ранжируют предлагаемые параметры из общего списка свойств, а затем на основе списков экспертов выделяются свойства, превосходящие некоторый порог по значимости и частоте включения в списки. Далее лицо принимающее решение производит объединение близких свойств в пары. Производя процедуру объединения, последовательно можно получить дихотомическую структуру дерева комплексного оценивания. При этом для объединенных в пары критериев строятся матрицы свертки – правила сопоставления рассматриваемых критериев. Таким образом, в результате последовательного выполнения шагов данного метода синтезируется МКО.

2. Метод последовательной дихотомии. Если метод бинарной композиции можно охарактеризовать схемой снизу вверх, то метод последовательной дихотомии (см., например, [24, 26]) – сверху вниз. В данном методе ЛПР из обобщающего критерия выделяет два равнозначных, например социокультурный критерий может быть разделен на два аспекта – социальный и культурный. Допустима схема с выделением главного критерия из обобщающего свойства, тогда от обобщающего свойства отделяется терминальная вершина бинарного дерева. Состав и структура критериев оценки объекта определяется не только целевым назначением объекта, но и совокупностью знаний в формате естественного языка, которыми располагает ЛПР. В процессе построения дихотомической структуры МКО происходит движение от абстрактных понятий к более конкретным. Как механизм сравнимости различных способов уточнения неопределенных понятий, в [26] предлагается следующая схема конкретизации понятий. Сложное понятие А представляется как объединение двух менее сложных понятий Б1 и Б2, каждое из менее сложных понятий,

в свою очередь, представляется как объединение пары еще менее сложных понятий. Таким образом, начиная с комплексного обобщающего свойства, происходит ступенчатое, дихотомическое деление пространства критериев.

Алгоритм получения комплексной оценки состоит из процедур попарных сверток критериев оцениваемого объекта, на матрицах расположенных в узлах бинарного дерева. Тогда результат оценивания будет получен в самой верхней матрице бинарного дерева.

Как было сказано ранее, когда решающее правило построено, лицо принимающее решение, участвовавшее в его построении, убежденно в том, что решающее правило сформировано на основе именно его предпочтений. Однако при решении задач многокритериального оценивания с опорой на правила оценки, формируемые ЛПР, закономерно задать вопрос об устойчивости информации сообщаемой ЛПР. Матрицы свертки используются в МКО согласно гипотезе ограниченности [26, 42]. Согласно гипотезе ограниченности, система переработки информации человека способна одновременно учитывать лишь ограниченное количество критериев [27, 42]. Поэтому попарные сравнения наиболее предпочтительны. Эта гипотеза проверялась экспериментально [27, 42]. В [27] утверждается, что при проведении повторной оценки лицом принимающим решение, некоторых объектов спустя 1,5 – 2 месяца, отличающихся оценками по двум или трем параметрам, различия при заполнении матриц свертки не превышают шести и тридцати процентов соответственно.

Часто задача формирования МКО не является ответственностью отдельного ЛПР, а ставится перед группой экспертов. Тогда решающее правило формируется в результате процесса согласования экспертных оценок. Подходы, реализующие данный принцип, носят название механизмов активной экспертизы. Важно учитывать, что привлеченные к оценке эксперты, могут быть заинтересованы в определенном результате экспертизы. Поэтому существует запрос на создание неманипулируемых механизмов многокритериального оценивания, см. например, [11, 13, 4].

Задачи комплексного оценивания, заключающиеся в переходе от детального к агрегированному описанию сложных систем, встречаются во многих областях хозяйственной деятельности. Ниже приведены некоторые примеры:

- оценка уровня инновационного развития отрасли [60];
- оценка проектов НИОКР [96];
- оценка программ регионального развития [5, 6, 43];
- в системе управления развитием организации [18, 15];
- оценка проектов в сфере малого и среднего предпринимательства [39];
- экологическая оценка [17, 61];
- оценка управления жилой недвижимостью [56, 2];
- оценка рисков [62];

— в механизмах поддержки принятия решений [63, 36].

МКО могут использоваться не только непосредственно для комплексного оценивания, но также и в механизмах при определении норм, квот, штрафов в экономической ответственности, при определении страховых взносов в механизмах страхования, при разработке планов формирования централизованных фондов и распределении бюджетных средств, наконец, при определении налоговой политики и политики льготного кредитования [15].

1.4. Базовый подход к экспертному синтезу МКО

Подход с опорой на экспертные знания при построении МКО в настоящее время достаточно хорошо проработан и изложен в ряде статей и монографий [15, 17, 96, 37]. Не углубляясь в детали, в данном разделе приведем основные моменты, связанные с составлением МКО, т.к. понимание канонической схемы синтеза МКО важно для понимания предложенных изменений на основе обучения, изложенных в рамках данной диссертации.

Основная идея при разработке механизма комплексного оценивания заключается в том, что каждая вершина структуры МКО, в том числе промежуточные и итоговая вершина, дезагрегируется ровно на две подвершины, то есть используется так называемый метод дихотомии. Это позволяет проводить агрегирование значений каждой пары вершин в оценку вышележащей вершины с помощью логических матриц свертки.

Реализация методологии оценивания состоит в последовательном выполнении ряда этапов. В данном подходе принципиальным моментом является тот факт, что для построения КО прежде всего формируется набор направлений или характеристик, которые дают общее представление об объекте и всех его особенностях с различных точек зрения. Для этих направлений строятся локальные оценки, которые затем сворачиваются в КО. Расчет КО – это выполнение последовательности этапов в соответствии с методологией комплексного оценивания. Ниже приведена последовательность основных этапов экспертного синтеза МКО (см., например, [15, 37]).

Первый этап – выделяются n направлений оценивания объекта или все характеристики объекта распределяются по n группам оценивания.

Второй этап – все эти направления разбиваются на две подгруппы.

- направления, где для построения оценки анализируются показатели, значения которых могут быть точно рассчитаны (например, финансовые, материальные, людские ресурсы или экономические показатели);

- направления, где для построения оценки анализируются показатели, значения которых объективно и точно рассчитать невозможно, и поэтому необходимо привлечение экспертов.
- направления, для построения оценки, по которым часть показателей могут быть точно рассчитаны, а для анализа другой части необходимо привлечение экспертов.

Таким образом, в результате такого разбиения может сформироваться одна, две, либо три подгруппы.

Третий этап – задать размер шкалы или определить максимальную балльную оценку, по которой можно оценить как отдельный показатель, так и направление в целом. При этом КО не может превысить этой максимальной оценки.

Четвертый этап – по результатам этапа 2 сформировать набор показателей, которые не могут быть точно рассчитаны, а оценены только экспертно. Сформировать группы экспертов по направлениям и определить балльные оценки этих показателей

Пятый этап – определяется множество показателей, которые характеризуют направления, входящие в первую подгруппу, то есть значения которых могут быть точно рассчитаны.

Шестой этап - определить для показателей, которые могут быть точно рассчитаны, максимальные и минимальные значения.

Седьмой этап – для всех показателей, выбранных на этапе 5, сформировать шкалы пересчета их значений в промежуточные балльные оценки.

Восьмой этап – определяются пары индикаторов, локальные оценки по которым будут сворачиваться в обобщенную оценку этих двух индикаторов. На данном этапе формируется так называемая *бинарная структура свертки*, которая наглядно иллюстрирует схему последовательного получения сначала обобщенных оценок, а затем и КО.

Девятый этап – формирование матриц свертки. Для получения КО локальные балльные оценки и обобщенные оценки попарно сворачиваются друг с другом, и формируются обобщенные оценки более высокого уровня иерархии. Формирование обобщенных оценок осуществляется при помощи матриц логической свертки.

Теперь с помощью сформированного МКО можно производить оценку рассматриваемого объекта. Для этого производится движение по построенному дереву от листьев (висячих вершин) к вершине. Как было сказано, основные элементы МКО – матрицы свертки и структура. Обсудим проблемы, связанные с ними в рамках базового подхода.

1.5. Выбор матриц свертки

Матрица свертки или матрица логической свертки – это таблица, номер строки которой соответствует балльной оценке одного критерия или обобщенной оценке нескольких критериев, а номер столбца – балльной оценке другого критерия или обобщенной оценке нескольких других критериев, см. рисунок 1. На пересечении этих строки и столбца как раз и находится обобщенная оценка. Отсчет строк и столбцов матрицы обычно ведется от ее верхнего левого угла. Однако можно встретить различные традиции нумерации. Например, пермские исследователи [62, 63] нумеруют строки и столбцы матриц свертки от правого нижнего угла, что обусловлено применяемой системой визуализации.

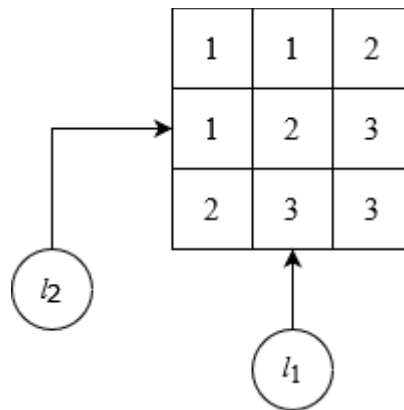


Рисунок 1 – Пример матрицы свертки.

В примере, изображенном на рисунке 1, значение l_1 одного индикатора соответствует номеру столбца, а значение другого l_2 - номеру строки.

Отметим, что для каждой пары сворачиваемых оценок выбирается своя матрица свертки. Для m -балльной шкалы матрицы имеют размерность $m \times m$. Для $m = 2$ можно выделить три базовые матрицы свертки [15], см. рисунок 2.

$$M_1 = \begin{array}{|c|c|} \hline 1 & 1 \\ \hline 1 & 2 \\ \hline \end{array} \quad M_2 = \begin{array}{|c|c|} \hline 1 & 1 \\ \hline 2 & 2 \\ \hline \end{array} \quad M_3 = \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 2 & 2 \\ \hline \end{array}$$

Рисунок 2 – Матрицы свертки для двухбалльной шкалы.

В [15] вопрос классификации матриц рассмотрен детально. Для данной работы важно, что выбор матрицы определяет характер агрегирования рассматриваемых оценок. Условно эти матрицы можно характеризовать следующим образом.

M_1 – Матрица наказания – из двух оценок всегда выбирается меньшая, то есть если одна из оценок, которые сворачиваются при помощи этой матрицы, равна 1, то обобщенная оценка также равна 1, независимо от того, чему равна другая сворачиваемая оценка.

M_2 – Матрица предпочтения. Здесь оценка, которая определяет номер строки матрицы, является предпочтительной по отношению к оценке, определяющей номер ее столбца.

M_3 – Матрица поощрения – из двух оценок всегда выбирается большая, то есть если одна из оценок, которые сворачиваются, равна 2, то независимо от того, какое значение имеет другая оценка, обобщенная оценка равна 2.

Для трехбалльной шкалы примеры базовых матриц свертки представлены на рисунке 3.

$$\begin{array}{ccc}
 M_1 = \begin{array}{|c|c|c|} \hline 1 & 2 & 3 \\ \hline 2 & 2 & 3 \\ \hline 3 & 3 & 3 \\ \hline \end{array} &
 M_2 = \begin{array}{|c|c|c|} \hline 1 & 2 & 2 \\ \hline 2 & 2 & 3 \\ \hline 2 & 3 & 3 \\ \hline \end{array} &
 M_3 = \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 2 & 2 \\ \hline 1 & 2 & 3 \\ \hline \end{array} \\
 \\
 M_4 = \begin{array}{|c|c|c|} \hline 1 & 1 & 2 \\ \hline 1 & 2 & 2 \\ \hline 2 & 2 & 3 \\ \hline \end{array} &
 M_5 = \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 2 & 2 & 2 \\ \hline 3 & 3 & 3 \\ \hline \end{array} &
 M_6 = \begin{array}{|c|c|c|} \hline 1 & 1 & 2 \\ \hline 2 & 2 & 2 \\ \hline 2 & 3 & 3 \\ \hline \end{array}
 \end{array}$$

Рисунок 3 – Матрицы свертки для трехбалльной шкалы.

По аналогии с матрицами свертки для двухбалльной шкалы матрицы трехбалльной шкалы можно характеризовать как

M_1 – Матрица максимального поощрения – из двух оценок всегда выбирается большая.

M_2 – Матрица умеренного поощрения.

M_3 – Матрица максимального наказания – из двух оценок всегда выбирается меньшая.

M_4 – Матрица умеренного наказания.

M_5 – Матрица абсолютного предпочтения одного показателя.

M_6 – Матрица предпочтения одного показателя.

В рамках базового подхода к построению МКО основное внимание при формировании матриц свертки должно быть уделено их непротиворечивости, то есть значение каждого элемента матрицы не должно противоречить логике ее построения.

Если обозначить через a_{ij} элементы матрицы свертки, то матрица будет непротиворечивой, см. [15], если выполняются следующие условия (считается, что нумерация столбцов и строк матриц свертки ведется от верхнего левого угла):

1. $a_{ii} = i$;
2. $\min(i,j) \leq a_{ij} \leq \max(i,j)$;
3. $a_{ij} \leq a_{ij+1}$;
4. $a_{ij} \leq a_{i+1j}$.

В качестве примера на рисунке 4 приведены противоречивые матрицы.

M_1	M_2																		
<table border="1" style="border-collapse: collapse; width: 100%; height: 100%;"> <tr><td>1</td><td>2</td><td>3</td></tr> <tr><td>2</td><td style="background-color: #f8d7da;">1</td><td>3</td></tr> <tr><td>3</td><td>3</td><td>3</td></tr> </table>	1	2	3	2	1	3	3	3	3	<table border="1" style="border-collapse: collapse; width: 100%; height: 100%;"> <tr><td>1</td><td>2</td><td>3</td></tr> <tr><td>2</td><td>2</td><td>3</td></tr> <tr><td style="background-color: #f8d7da;">3</td><td style="background-color: #f8d7da;">2</td><td>3</td></tr> </table>	1	2	3	2	2	3	3	2	3
1	2	3																	
2	1	3																	
3	3	3																	
1	2	3																	
2	2	3																	
3	2	3																	

Рисунок 4 – Противоречивые матрицы.

Здесь выделены элементы матриц, которые противоречат логике их построения. Так в матрице M_1 выделенный диагональный элемент не может равняться 1. Он находится на пересечении третьей строки и третьего столбца, что соответствует сворачиванию двух оценок, равным 2, а в этом случае обобщенная оценка может быть только 2. Что касается матрицы M_2 , то значения выделенных элементов также противоречат друг другу. Дело в том, что первый выделенный элемент является результатом свертки локальных оценок 3 и 1 и может принимать значение равно 3, но тогда второй выделенный элемент, являющийся результатом свертки локальных оценок 3 и 2, должен быть больше или равен свертки локальных оценок 3 и 1, то есть принимать значение 3.

1.6. Выбор структуры дихотомического дерева комплексного оценивания

Структура дихотомического дерева комплексного оценивания, также именуемая как бинарная структура свертки или структура полного бинарного дерева на именованных листьях, наглядно иллюстрирует схему последовательного получения сначала обобщенных оценок, а затем и КО (см., например, [15, 17]).

Простейшая бинарная структура строится для двух индикаторов. Естественно, что эта структура единственная, и для двух локальных оценок существует только один вариант свертки оценок. Поэтому если для построения КО выделено только два индикатора, то привлечение

экспертов для формирования бинарной структуры не требуется. Бинарная структура для простейшего случая изображена на рисунке 5.

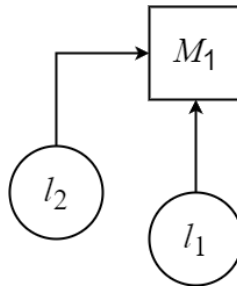


Рисунок 5 – Простейшая бинарная структура.

Для трех индикаторов и, соответственно, для трех локальных оценок (l_1 , l_2 и l_3), как и для двух, бинарная структура также может быть построена только одним способом. Эта структура представлена на рисунке 6.

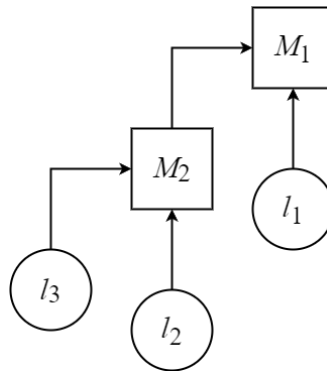


Рисунок 6 – Бинарная структура для трех локальных оценок.

Однако для этой единственной структуры уже возможно построить три варианта свертки. Все возможные варианты получения обобщенной (l_{ij}) и КО оценок для трех направлений представлены в таблице 1.

Таблица 1 – Варианты свертки 3х вершин.

№ варианта	i	j	k
1	1	2	3
2	1	3	2
3	2	3	1

Четыре индикатора уже позволяют построить две бинарные структуры свертки. Первая структура основана на параллельном сворачивании локальных и обобщенных оценок – параллельная структура. При этом сначала из двух пар оценок строятся две обобщенные оценки, а затем уже из полученных обобщенных оценок формируется КО. Второй способ построения

бинарной структуры для четырех индикаторов – это последовательное агрегирование локальных оценок индикаторов – последовательная структура. Эту структуру можно назвать «ветка дерева». При этом на первом уровне иерархии находятся локальные оценки только двух индикаторов, на втором уровне иерархии находится одна обобщенная оценка, построенная на основе свертки двух локальных оценок первого уровня, и одна локальная оценка третьего индикатора. Обобщенная оценка третьего уровня получается путем свертки обобщенной оценки второго уровня и локальной оценки третьего индикатора. И, наконец, на следующем уровне, агрегируя обобщенную оценку третьего уровня и локальную оценку четвертого индикатора, получаем КО. На рисунке 7 представлены примеры дихотомических деревьев на четырех индикаторах.

Число возможных структур со всеми вариантами сверток на них описывается формулой $n = (2l - 3)!!$, где n – общее число возможных структур, l – число критериев оценки объекта.

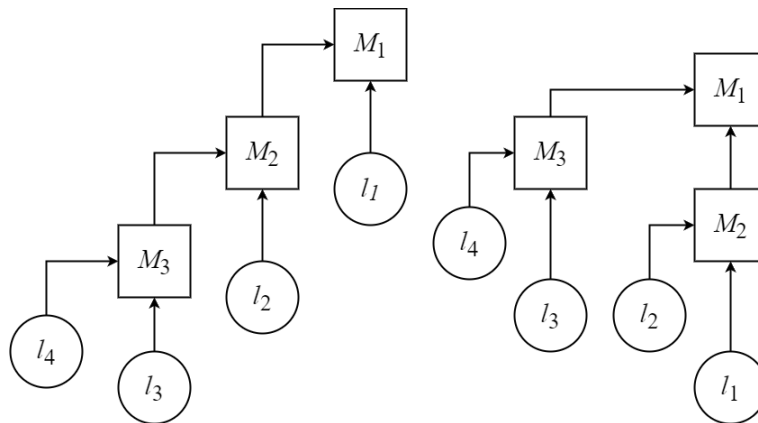


Рисунок 7 – Примеры дихотомических деревьев на четырех индикаторах.

Из приведенных формул видно, что с ростом числа параметров для экспертов задача выбора структуры существенно усложняется. Примеры содержательных правил построения структуры можно найти в [29]. Задачам выбора структур и синтеза матриц МКО посвящены соответствующие главы работы.

1.7. Возможные подходы к решению задачи синтеза

В области методов машинного обучения существует целое направление – обучение по прецедентам [45, 122]. В рамках которого, выделяют обучение на размеченных данных, называемое обучение с учителем. Задачу синтеза или идентификации МКО на основе табличных данных можно рассматривать как задачу обучения с учителем. В отличие от широко практикуемых в настоящее время методов классификации, основанных на нейронных сетях, рассматриваемый в данной диссертации подход обладает тем преимуществом, что имеет открытое внутреннее устройство. Это значит, что лицо принимающее решение имеет возможность получить полное объяснение по каждому результату, сформированному системой комплексного оценивания.

Существует несколько основных типов объяснений: на основе текста, визуальные, в локальной области, на примере, через упрощение, на основе сравнения влияния параметров [73]. Например, работу МКО можно объяснять через влияние изменения каждого из параметров на результат. Для формирования более полной картины области можно обратить внимание на ряд работ посвященных освещению и сравнению основных методов интерпретируемых механизмов машинного обучения [119, 73, 86]. Тема интерпретируемых механизмов машинного обучения [65, 73, 121, 120, 128, 127] все активнее обсуждается в академическом сообществе в последнее время: это видно по росту числа публикаций по данной теме [65]. В этой связи развитие методов обучения МКО представляется актуальной темой исследования.

Далее освещены некоторые работы, связанные с вопросом обучения и представляющиеся актуальными для данной диссертации. Преимущественно это работы в рамках двух направлений – обучение булевой функции и функциональная декомпозиция в рамках машинного обучения.

В настоящее время существует целое направление исследования булевых функций, называемое точным обучением по запросам [69, 75, 130, 131, 135], в рамках которого традиционно рассматриваются вопросы, связанные со сложностью обучения (идентификации) некоторой булевой функции. В силу наличия связи между формульным и древовидным представлением булевой функции, например, см. [135], рассмотрение вопросов связанных с обучаемостью булевых функций, прежде всего неповторных, вполне актуально для задач синтеза булевых функций с помощью МКО. Основополагающей в рамках данного направления является работа L. Valiant [130] из области обучения булевых функций. Для построения теории L. Valiant ввел такие источники информации о неизвестной концепции, как оракулы; в реальности в качестве оракула может выступать эксперт или база данных. Он определил несколько типов оракулов, три основных он назвал так: необходимость, соответствующая вероятность (ОСВ) и соответствующее сопровождение (ОСС). L. Valiant показал, что монотонные дизъюнктивные формулы в нормальной форме могут быть приближенно идентифицированы за полиномиальное время из случайных примеров и оракула необходимости. Он также показал, что неповторные формулы, когда каждая из переменных входит в формулу только один раз, могут быть идентифицированы полиномиальным алгоритмом, использующим все три типа оракулов. В работе [131] L. Valiant описал понятие вероятно приближенно корректное обучения (ВПК) и показал, что любая булева функция может быть изучена с произвольным уровнем точности на достаточном количестве примеров. L. Valiant представил теоретическую основу для понимания обучаемости булевых функций, которая широко используется в области машинного обучения и привела к разработке множества алгоритмов для изучения булевых функций. Важно отметить, что под термином «обучение булевой функции» обычно понимается получение формулы, описывающей исследуемую табличную функцию.

Идеи L. Valiant были развиты такими исследователями, как D. Angluin, A. Blum, L. Hellerstein, M. Karpinski, В.А. Гурвичем и другими [31, 69, 75, 135] представителями направления идентификации функции по запросам. Например, M. Kearns и др. [93] показывают, что монотонные неповторные формулы не легче предсказывать в модели без распределения, чем общие булевы формулы. D. Angluin [69, 68] предложила изучать понятие точной идентифицируемости понятий за полиномиальное время с использованием различных типов запросов – оракулов, в терминологии L. Valiant. Два типа запросов кажутся особенно интересными: запросы на принадлежность и запросы на эквивалентность. В запросе на членство алгоритм обучения предлагает конкретный пример, а ответом является правильная классификация примера в соответствии с неизвестной концепцией. В запросе на эквивалентность алгоритм обучения предлагает гипотезу на указанном языке гипотез, и ответом является либо «да», либо контрпример.

Проблема с обучением булевой функции в общем виде, состоит в том, что для идентификации булевой функции n переменных необходимо задать 2^n вопросов. В случаях, когда мы знаем некоторую дополнительную информацию о рассматриваемой функции, можно уменьшить необходимое количество вопросов. Например, широко практикуется подход исследования монотонных функций, что позволяет уменьшить объем вопросов к оракулу [135].

В [69] рассматривается сложность точной идентификации как неповторных формул, так и монотонных неповторных формул. Оказывается, что неповторные формулы общего вида, т. е. не обязательно монотонные, не могут быть точно идентифицированы за полиномиальное время только с помощью запросов на принадлежность. Более того, D. Angluin предъявляет полиномиальный алгоритм для точной идентификации монотонных неповторных формул с использованием запросов на принадлежность и доказывает, что не существует алгоритма с полиномиальным временем для обучения монотонным неповторным формулам только с запросами эквивалентности. Также представляется полиномиальный алгоритм, являющийся усовершенствованием алгоритма L. Valiant, для точной идентификации общих неповторных формул с использованием оракула соответствующей вероятности. Вопрос ограниченности предложенных методов идентификации исследован D. Angluin в [70]. В [135] V. Gurvich и M. Golombic проводится дальнейшее развитие области методов точного обучения. Вводятся необходимые понятия с привлечением теории графов и предлагаются алгоритмы для решения задач распознавания и точного обучения функции. Приводятся рассуждения по поводу булевых выражений, в которых каждая переменная появляется ровно t раз.

Как обозначено выше, в рамках дисциплины точного обучения по запросам занимаются булевыми функциями, в то время как для задач синтеза МКО бинарная шкала – это лишь частный случай. Однако при разработке методов обучения МКО мы не требуем монотонности синтезируемой функции. Так как устройство МКО предполагает использование каждого параметра лишь

единожды, то из всей области наиболее актуальными являются работы посвященные неповторным функциям [34]. Исследование вопроса представимости функции в неповторном виде имеет важные практические следствия. Например, в области электроники существует вопрос представления булевых функций неповторными формулами [92, 40, 58, 97]. Необходимость разделения формул на повторные и неповторные вызвана тем, что от этого существенным образом зависит сложность их программной или аппаратурной реализации. При аппаратурной реализации неповторных формул трассировка данных от места хранения переменной до места ее обработки осуществляется к единственному входу логического элемента. В то время как для повторных формул такая трассировка требуется ко многим элементам. При программной реализации повторные формулы приводят к многократному обращению к ячейкам памяти, где хранятся переменные. Вычисление же неповторной формулы приводит к единственной выборке данных из входных ячеек. У следующего направления исследований, тесно связанного с темой данной диссертационной работы и рассматриваемого в данном разделе, также можно найти корни в электронике.

В рамках дисциплины машинного обучения развивалось множество методов, работающих на основе табличных данных, например, линейные модели, деревья решений и случайные леса, нейронные сети, градиентный бустинг и другие. В рамках данной диссертационной работы хочется обратить внимание на методы функциональной декомпозиции, которые так же, как и МКО используют подход дихотомического деления, но, в отличие от НС, являются открытыми моделями. Согласно [141, 142] основы подхода функциональной декомпозиции были обозначены в области проектирования коммутационных схем [71]. Н. Curtis в [81] сообщает, что интерес к данной теме был проявлен еще в 1940-х и 1950-х годах со стороны теоретиков коммутационных схем. Позже, в [71] R. Ashenhurst сообщил о единой теории декомпозиции функций переключения. Отличительной особенностью начального этапа была работа с полностью определенными и в основном булевыми дискретными функциями. Т. Luba, Н. Selvaraj, М. Perkowski и другие [104, 116] развили подход R. Ashenhurst: был предложен метод работы с не полностью заданными функциями, на основе эвристики раскраски графа. Представленный подход был ориентирован на область проектирования функций для ПЛИС [105]. На основе методов декомпозиции булевых функций образовались методы декомпозиции функций больших размерностей [140].

Важной вехой является работа [123], в которой А. Samuel использует иерархический подход к табличным данным на примере составления алгоритма для игры в шашки. Он приходит к необходимости иерархического представления игры в виде ее разбиения на три фазы – дебют, миттельшпиль и эндшпиль. Соответственно разбивается и пространство параметров. На каждом из этапов параметры агрегируются в таблицах относительно небольшой размерности – от 68 до 343 записей (в зависимости от иерархической схемы агрегации).

Позже вводятся понятия параллельной и последовательной декомпозиции [104, 76]. До некоторого времени, вышеупомянутые подходы декомпозиции обычно имеют дело с бесшумными данными и булевыми функциями. Различные расширения для работы с номинальными признаками были предложены в [140, 84].

Наконец, на основе декомпозиционного подхода появляются методы работы с качественными данными для неполностью определенных функций [143], что уже можно отнести к области ВАР методов, хотя В. Zupan и его соавторы позиционируют свои работы как работы из области машинного обучения. Например, в [143] предлагается метод, основанный на функциональной декомпозиции, который идентифицирует типичные группы признаков для ревматических заболеваний. Результаты сравниваются с оценками, предоставленными медицинским экспертом.

В части развития теории декомпозиции алгебраических методов хочется также отметить вклад В.С. Выхованца [22, 23]. Он рассматривает задачу идентификации дискретной функции, заданной в табличной форме. Табличная функция с помощью метода декомпозиции, представляется в виде формул, состоящих из входных переменных и операций над ними. Характерно, что не накладывается ограничений на размерность исследуемых данных. Автор предлагает механизм нахождения формул, и механизм редукции операций бесповторной формы функции; исследует вопрос получения бесповторных формул, что важно как с точки зрения их программной, так и аппаратной реализации. Также он рассматривает возможность бесскобочного представления формул, т.к. при реализации скобочных формул порядок вычислений отличается от естественного. Хотя для бесскобочного варианта и требуется сохранение промежуточных результатов вычисления подформул, заключенных в скобки. Кроме того, освещен вопрос оценок сложности синтезируемых формул.

В.Н. Бурков, И.В. Буркова и А.В. Щепкин в [14] предложили метод синтеза матриц МКО, при заданной структуре МКО. Матрицы формируются на основе заданного множества примеров, с определенными экспертами оценками. Ставится ряд ограничений на формируемые матрицы, в частности требуется одинаковая размерность и монотонности всех матриц. В данной работе мы также работаем с матрицами в единой размерности, однако не вводим ограничения на монотонность матриц, что позволяет синтезировать немонотонные функции. Важная особенность, затронутая В.Н. Бурковым и коллегами – требование минимального числа примеров для синтеза МКО, что сокращает затраты времени экспертов, хотя при таком подходе и нельзя утверждать, что построенный механизм будет удовлетворять предпочтениям экспертов на нерассмотренных примерах (вариантах). Приводятся оценки необходимого числа экспертных примеров (вариантов) для построения МКО.

А.О. Алексеев в [66] предлагает подход к синтезу МКО, который условно предлагается называть табличным. Он приводит метод синтеза, основанный на непосредственной работе с

таблицей размеченных данных, для полного набора данных, которая состоит из четырех шагов, которые можно представить как процесс формирования четырех таблиц. Ключевым отличием метода синтеза, предлагаемого А.О. Алексеевым, является гибкость в формировании размерности матриц МКО. В данной диссертационной работе рассматриваются квадратные матрицы и размерность матриц МКО ограничивается размерностью сворачиваемых на них параметров, что можно трактовать как более экономную форму представления МКО. Стоит отметить, что предлагаемый Алексеевым подход, в силу отсутствия требования на размерность матриц МКО, не имеет ограничений, связанных с количеством групп эквивалентности [95], на формирование МКО реализующий предложенный набор входных данных.

Многие рассмотренные в данном разделе подходы берут начало из области электроники. Но стартовав из прикладных задач электроники, исследуют важные теоретические вопросы обучаемости и декомпозируемости функций и открывают путь к современным задачам интерпретируемых механизмов [120, 119, 128], как например МКО. Дополнительно хочется отметить особенность методов МКО по работе с небольшими наборами данных. Существует точка зрения [122], согласно которой основа эффективности глубокого обучения находится в возможности изучения глубоких иерархических представлений данных. МКО используют преимущества от работы с иерархическим представлением, но способны эффективно обучаться на датасетах с малым количеством обучающих примеров. О конкретных подходах к обучению МКО речь пойдет в следующих главах данной работы. Но, в начале введем формализм необходимый для постановки задачи обучения МКО.

1.8. Основные понятия и определения

Для формальной постановки задачи синтеза МКО требуется ввести обозначения для используемых терминов и дать более строгое определение механизма комплексного оценивания, чем существующие варианты (см., например, [33, 5, 4, 46, 15, 10]). Пусть задан конечный набор индикаторов, показателей или переменных – в дальнейшем эти три термина мы будем считать синонимами $L \subset \mathbb{N}$, $|L| = l$, на основе их значений должна быть произведена порядковая оценка некоторого объекта или ранжирование нескольких объектов. Отдельный индикатор отражает определенное свойство рассматриваемого сложного объекта. Значения индикаторов задаются в дискретной шкале, например, 1 балл – плохо, 2 балла – удовлетворительно, 3 – хорошо. Для задачи синтеза МКО будем считать, что для каждого индикатора $i \in L$ задан конечный набор $K_i \subset \mathbb{N}$ его возможных значений, $k_i \in K_i$ – оценка, дискретная величина. Вектор $k = (k_1, \dots, k_l)^T$ – совокупность оценок на основе оценок отдельных индикаторов. Считаем, что совокупность всех возможных комбинаций оценок, описывает любое возможное состояние оцениваемого объекта. Также

есть конечный набор $K_L \subset \mathbb{N}$ возможных интегральных значений (рангов или классов) $k_L \in K_L$ для любого k .

Методы комплексного оценивания, работе с которыми посвящена данная диссертация, более корректно именовать как матричные методы комплексного оценивания ММКО [66] из-за использования матриц логической свертки при формировании оценки. Наряду с термином «механизм комплексного оценивания» существует ряд работ [26, 47] в которых используется термин «метод векторной стратификации», но в рамках данной работы будем придерживаться устоявшегося термина «механизм комплексного оценивания». Несмотря на широкое употребление термина «механизм комплексного оценивания» и разработанную теорию экспертного синтеза МКО [15, 14, 33, 46], для теоретического исследования проблемы синтеза МКО на основе обучающих наборов данных, необходимо дать более формальное определение МКО как математического объекта, позволяющее детализировать и адресовать основные компоненты МКО. Предложенные ранее определения не обладают необходимым нам уровнем формализации, поэтому нами предлагается следующее определение МКО.

Определение 1. Механизм комплексного оценивания – это отображение $w(\cdot): K \rightarrow K_L$, для которого индикаторы L являются листьями полного бинарного дерева – ориентированного графа $G = (V, E)$:

1. $V = L \cup \hat{L}$, $\hat{L} = \{l+1, \dots, 2l-1\}$.
2. $E = \{e_{ij}\} \subseteq V \times V$,
 - а. $\forall i \in V \setminus \{2l-1\} \exists! j \in \hat{L} \setminus \{i\}: e_{ij} = 1, \forall t \in V \setminus j e_{it} = 0$;
 - б. $\forall j \in L \forall i \in V e_{ij} = 0$;
 - в. $\forall j \in \hat{L} \exists! \{r, c\} \in V \setminus \{j\} \times V \setminus \{j\}: e_{rj} = 1, e_{cj} = 1$;

и $\forall j \in \hat{L}$ заданы:

- 1) конечный набор $K_j \subset \mathbb{N}$ с возможными значениями $k_j \in K_j$, $K_{2l-1} = K_L$;
- 2) матрица свертки $M_j = [m_{jgp} \in K_j]_{g \in \{0, \dots, |K_r|-1\}, p \in \{0, \dots, |K_c|-1\}}$, и $\{r, c\} \in V \setminus \{j\} \times V \setminus \{j\}: e_{rj} = 1, e_{cj} = 1$

определены. \square

Здесь $K = \prod_{i \in L} K_i$ обозначает декартово произведение $K_1 \times K_2 \times \dots \times K_l$, V – множество всех вершин G , \hat{L} – множество внутренних вершин G , E – множество ребер G заданное матрицей смежности, g – строка матрицы свертки, p – столбец матрицы свертки. Потенциально данное определение может быть расширено на нечеткие [3, 64, 68] или непрерывные шкалы [26]; в рамках данной работы акцент сделан на работу именно с дискретными шкалами как индикаторов (показателей), так и значений, получаемых в узлах дерева свертки [19]. Пример механизма

комплексного оценивания на трех индикаторах, построенный для функции $f(l_1, l_2, l_3) = l_1 \wedge (l_2 \vee l_3)$, представлен на рисунке 8.

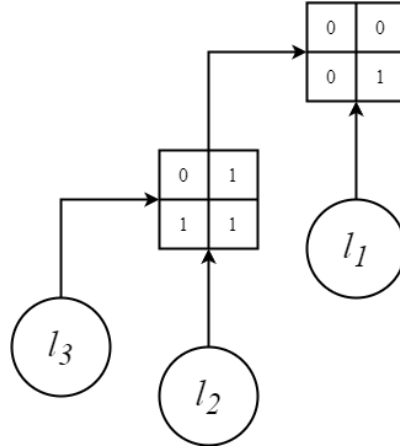


Рисунок 8 – Пример механизма комплексного оценивания на трех индикаторах.

В рамках данной работы рассматриваются МКО с единой шкалой, такие что $\forall j \in V K_j = K_L$.

Для $L \subset \mathbb{N}$ обозначим через $\Gamma_2(L)$ – набор всех бинарных деревьев с листьями из L ; $IRM_{L,2}$ – набор всех МКО для любого конкретного бинарного дерева $G \in \Gamma_2(L)$; $IRM_{L,G} \subseteq IRM_{L,2}$ – набор всех МКО с таким деревом. Как формально показано в определении 1, в данной работе под определением полного бинарного дерева понимается дерево, в котором каждый узел имеет либо ни одного, либо два дочерних элемента.

Имея некоторый МКО $w(\cdot)$, обозначим набор всех его матриц свертки через $M_w = \{M_j\}_{j \in L}$.

Тогда используя определение 1, каждому МКО можно сопоставить кортеж $\langle K, M_w, G \rangle$. Например, МКО, изображенный на рисунке 8, можно записать как: $\langle \{0,1\}^3, \{M_p, M_s\}, G \rangle$, где M_p , M_s – матрица наказания и матрица поощрения соответственно. $G = (V, E)$, где $V = \{l_1, l_2, l_3, M_1, M_2\}$,

$$E = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Для формализации задачи обучения обозначим через $q = (k, k_L)$ отдельный обучающий пример, через $Q \subset K \otimes K_L$ обучающий набор (из предоставленных примеров), состоящий из значений оценок по каждому из индикаторов и интегральной оценки для данной совокупности значений индикаторов.

Обучающий набор является *согласованным*, если $\forall \{q, \tilde{q}\} \subseteq Q$ имеем $k \neq \tilde{k}$. Обучающий набор является *полным*, если $\forall k \in K \exists q \in Q: q = (k, k_L)$. Обучающий набор задан в *единой шкале*,

если $\forall i \in L$ верно $K_i = K_L$. Для произвольных $\{k, \tilde{k}\} \subseteq K$ обозначим $k \succ \tilde{k}$, если $\forall i \in L$ выполнено $\tilde{k}_i \leq k_i$.

Считаем, что обучающий набор Q называется монотонным когда $\forall \{q, \tilde{q}\} \subseteq Q$: если $k \succ \tilde{k}$, то $\tilde{k}_L \leq k_L$. В таком случае $q \succ \tilde{q}$.

Для некоторого произвольного $Q \subset K \otimes K_L$ можно определить следующие ключевые обозначения, касающиеся проблемы синтеза (см., например, [16, 79]). Во-первых, можно формализовать задачу *реализуемости* обучающего набора.

Определение 2. $w(\cdot) \in IRM_{L,2}$ реализует Q тогда и только тогда, когда $\forall q \in Q$ имеет место равенство $w(k) = k_L$. \square

Обозначим через $IRM_{L,2}(Q)$ множество всех МКО, которые реализуют Q , через $IRM_{L,G}(Q)$ множество всех МКО, которые реализуют Q и построены на основе двоичного дерева $G \in \Gamma_2(L)$. Тогда если $IRM_{L,2}(Q) \neq \emptyset$, то Q реализуема на основе МКО, если $IRM_{L,G}(Q) \neq \emptyset$, тогда Q является реализуемой на основе МКО со структурой G .

Таким образом, задача синтеза МКО состоит в поиске матриц $M_j, j \in \{l+1, \dots, 2l-1\}$, либо на наборе деревьев $\Gamma_2(L)$: $w(\cdot) \in IRM_{L,2}(Q)$, либо $w(\cdot) \in IRM_{L,G}(Q)$ на некотором определенном дереве $G \in \Gamma_2(L)$.

Определение 2 также может быть сужено до одного конкретного обучающего примера: $w(\cdot) \in IRM_{L,2}$ реализует некоторый пример $q \in Q$ тогда и только тогда, когда $w(k) = k_L$; так же определяются $IRM_{L,2}(q)$ и $IRM_{L,G}(q)$.

Если $IRM_{L,2}(Q) = \emptyset$ или $IRM_{L,G}(Q) = \emptyset$, то возможно поставить задачу *аппроксимации*. Для конкретного $w(\cdot) \in IRM_{L,2}$ обозначим $Q_w = \{q \subseteq Q : w(k) = k_L\}$. Для любого произвольного $w(\cdot) \in IRM_{L,2}$ обозначим через $U_Q(w) = |Q_w| / |Q|$ качество аппроксимации с максимумом в 1, если $w(\cdot) \in IRM_{L,2}(Q)$. Тогда задача аппроксимации заключается в поиске $w^*(\cdot) \in \underset{w \in IRM_{L,2}}{Arg \max} U_Q(w)$. Для

отдельного бинарного дерева $G \in \Gamma_2(L)$ может быть сформулирована аналогичная задача – найти $w^*(\cdot) \in \underset{w \in IRM_{L,G}}{Arg \max} U_Q(w)$.

Вышеуказанные проблемы носят комбинаторный характер, например, $|\Gamma_2(L)| = (2l-3)!!$, то есть количество полных бинарных деревьев на l именованных листьях растет как двойной факториал. Но для определенного $G \in \Gamma_2(L)$ число переменных – количество ячеек всех матриц свертки – минимально по сравнению с любым сетевым представлением функции, которая реализует Q (см. например, [16]); для МКО с единой шкалой это $(l-1)(|K_L|+1)^2$, в то время как полное число ячеек таблицы представления отображения $\prod_{i \in L} K_i \rightarrow K_L$ равно $(|K_L|+1)^l$. Поэтому МКО

может рассматриваться как самое экономное приближение некоторого дискретного отображения

$$\prod_{i \in L} K_i \rightarrow K_L.$$

1.9. Выводы по главе 1

В первой главе представлен обзор области методов мультикритериального оценивания, на основе которого обозначено место механизма комплексного оценивания как метода из области вербального анализа. Рассмотрение работ посвященных экспертному синтезу позволило обратить внимание на ключевые проблемы задачи синтеза МКО – выбор матриц и структур МКО. На основе проведенного обзора исследований по существующим методам синтеза МКО и методам из смежных областей можно сделать вывод об актуальности разработки методов синтеза МКО на основе обучающих наборов данных. Более того, было обнаружено, что отсутствует достаточно строгая система определений и понятий, позволяющая строго формулировать данную задачу, в следствии чего, в завершение главы 1, была предложена такая оригинальная система понятий и определений.

В следующих главах на основе введенного формализма будут рассмотрены подходы позволяющие упростить работу над задачами синтеза структур и матриц МКО. Так, для задачи выбора структур будут предложены методы отсева неперспективных деревьев, а для матриц будут предложены подходы к синтезу, основанные на обучающих наборах данных.

Глава 2. Анализ и синтез структур МКО

Настоящая глава посвящена анализу и синтезу структур механизмов комплексного оценивания. Проблема исследования множества допустимых структур МКО имеет комбинаторный характер, т.к. число структур для рассмотрения равно $|\Gamma_2(L)| = (2l - 3)!!$. Поэтому, крайне важной является задача определения перспективных полных бинарных деревьев для рассмотрения. В разделе 2.1 предлагаются правила для кодирования МКО, позволяющие работать с множеством структур полных бинарных деревьев как с набором записей. В разделе 2.2 вводится инструмент унитарного кодирования, на основе которого демонстрируется пример формулирования задачи синтеза матриц МКО как оптимизационной проблемы. В разделе 2.3 предлагается метод анализа групп эквивалентности (АГЭ), приводятся утверждения о существовании МКО, реализующих рассматриваемый набор дискретных данных в рамках заданной шкалы. В разделе 2.4 предлагаются алгоритмы поиска числа групп эквивалентности. В разделе 2.5 предлагается компактная, поуровневая запись комбинаций листьев МКО, названная «таблица ветвей». Раздел 2.6 посвящен методам генерации структур МКО.

2.1. Правила для кодирования МКО

Для некоторого конечного набора $K \subset \mathbb{N}$ и, для некоторой матрицы свертки $M = [m_{rc} \in K]_{\{r,c\} \in (|K|-1)^2}$ результат свертки для любой пары $\{l_1, l_2\} \in \{0, \dots, |K| - 1\}^2$, где l_1 определяет выбор столбца матрицы и l_2 – строки, описывается матричным уравнением $l_2^T M l_1$. Далее мы также будем использовать так называемую квадратичную форму представления $(M l_1, l_2)$, упрощая ее до $M l_1 l_2$.

Принимая во внимание тот факт, что любое полное двоичное дерево с l листьями должно иметь $l - 1$ внутренних узлов, включая корень, любой МКО может быть закодирован в форме «слова» с $2l - 1$ буквами в упрощенном квадратичном представлении; $l - 1$ буквами должны обозначаться матрицы свертки (внутренние узлы) и l – листья (см., например [16]).

Предлагаются следующие правила для кодирования МКО.

1. Первая буква всегда должна обозначать матрицу свертки для корневого узла.
2. Если после буквы матрицы в слове есть только одна буква листа или другая буква матрицы, то запись описывает «спуск» вдоль ветви дерева. Под ветвью полного бинарного дерева мы понимаем любое поддерево, исходящее из одной из вершин рассматриваемого дерева, за исключением поддеревьев содержащих только листья. Если после некоторой буквы матрицы

расположены две буквы листьев, то это «терминальная» матрица свертки в этой ветви дерева, после которой запись «поднимается» вдоль уже описанной ветви в поисках следующей ветви.

3. После любой буквы матрицы сперва должно быть закодировано «низкое» поддерево (с меньшей высотой) из двух, которые исходили из этого узла, а затем второе, «высокое». Таким образом, любая структура полного бинарного дерева на именованных листьях может быть записана в предложенной мл-нотации, для обозначения будем также пользоваться терминами: структура, запись, мл-запись. Например, структура, изображенная на рисунке 6, может быть записана как: $M_1 l_1 M_2 l_2 l_3$.

2.2. Унитарное представление механизмов комплексного оценивания

В результате исследования работ посвященных синтезу МКО оказалось, что многие исследователи ограничиваются графическим представлением матричных операций МКО [5, 15, 46], а некоторые записывают операцию матричной свертки с помощью квадратичной формы [8, 32]. В данном разделе вводится инструмент унитарного кодирования, (см., например [16, 79, 113]) который позволяет введенную в разделе 2.1 форму записи трактовать как последовательность матричных операций с унитарно закодированными значениями листьев в соответствии со «словом», которое описывает соответствующий МКО. Результатом такой последовательности операций является унитарно закодированный результат МКО \tilde{w} .

Для некоторого конечного набора $K \subset \mathbb{N}$ введем его *нормализованное* представление $\bar{K} = \{0, \dots, |K| - 1\}$. Тогда $\forall l_i \in \bar{K}$, $i \in \{1, \dots, |K|\}$, введем унитарное представление $\tilde{l}_i = (0, \dots, 0, 1, 0, \dots, 0)^T$. В этой работе мы ограничимся рассмотрением МКО с единой шкалой.

Пример 1. Рассмотрим $|L| = 3$. Любой МКО, на трех листьях и с единой шкалой может быть представлен как $(M_1 \cdot, \tilde{b})$, $b = (M_2 \cdot \cdot, \dots)$, где $\{l_1, l_2, l_3\} \in \bar{K}^3$, $\tilde{l}_1, \tilde{l}_2, \tilde{l}_3$ расположены на позициях \cdot, \cdot, \dots произвольным образом, без повторов, и $M_1 = [m1_{rc} \in \bar{K}]_{\{r,c\} \in \bar{K}^2}$, $M_2 = [m2_{rc} \in \bar{K}]_{\{r,c\} \in \bar{K}^2}$ – некоторые квадратные матрицы. □

Мы также применяем операцию унитарного кодирования к матрицам свертки. Дана некоторая матрица свертки $M = [m_{rc} \in K]_{\{r,c\} \in \bar{K}^2}$, обозначим $\tilde{M} = [\tilde{m}_{rc}]_{\{r,c\} \in \bar{K}^2}$, где \tilde{m}_{rc} – унитарное представление m_{rc} в наборе K . Унитарная матрица – это матрица, в каждой ячейке которой стоит унитарный вектор. Тогда операция $\tilde{l}_2^T \tilde{M} \tilde{l}_1$ даст унитарный результат для операции $\tilde{l}_2^T M \tilde{l}_1$, описанной выше. В упрощенном квадратичном представлении – $\tilde{M} \tilde{l}_2$.

Пример 2. Пусть $|K_L| = 2$, что можно трактовать как классическую бинарную логику. Тогда

$$\tilde{0} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \tilde{1} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \tilde{M} = \begin{bmatrix} \begin{pmatrix} m_{00}^0 \\ m_{00}^1 \end{pmatrix} & \begin{pmatrix} m_{01}^0 \\ m_{01}^1 \end{pmatrix} \\ \begin{pmatrix} m_{10}^0 \\ m_{10}^1 \end{pmatrix} & \begin{pmatrix} m_{11}^0 \\ m_{11}^1 \end{pmatrix} \end{bmatrix}$$

с дополнительными унитарными ограничениями:

$$\forall \{r, c\} \in \{0, 1\}^2 \text{ имеем } m_{rc}^0 + m_{rc}^1 = 1, \forall t \in \{0, 1\} m_{rc}^t \in \{0, 1\}.$$

Помня о том, что на позициях внутри матриц находятся унитарные векторы, можем выполнить операции матричного умножения, например для операции $\tilde{M}\tilde{1}\tilde{0}$:

$$(1, 0) \begin{bmatrix} \begin{pmatrix} m_{00}^0 \\ m_{00}^1 \end{pmatrix} & \begin{pmatrix} m_{01}^0 \\ m_{01}^1 \end{pmatrix} \\ \begin{pmatrix} m_{10}^0 \\ m_{10}^1 \end{pmatrix} & \begin{pmatrix} m_{11}^0 \\ m_{11}^1 \end{pmatrix} \end{bmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = (1, 0) \begin{pmatrix} \begin{pmatrix} m_{01}^0 \\ m_{01}^1 \end{pmatrix} \\ \begin{pmatrix} m_{11}^0 \\ m_{11}^1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} m_{01}^0 \\ m_{01}^1 \end{pmatrix}. \square$$

В результате получили унитарно закодированный вектор. Можно заметить, что операция умножения матрицы на унитарно закодированный вектор, может быть интерпретирована как операция фильтрации. При умножении строки на матрицу $(1, 0)\tilde{M}$ из матрицы выделяется строка с номером равным номеру позиции единицы в унитарно закодированной строке. При умножении матрицы на столбец $\tilde{M}\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ из матрицы выделяется некоторый столбец, соответствующий позиции единицы в унитарно закодированном столбце.

2.3. Группы эквивалентности

Теперь давайте представим идею о чувствительности обучающего набора Q или функции, описываемой этим набором f_Q , к некоторому подмножеству его индикаторов (см., например [38, 55, 95]).

Дано некоторое подмножество индикаторов $\tilde{L} \subseteq L$, обозначим через $k = (k_{(\tilde{L})}, k_{(L \setminus \tilde{L})})$ разбиение кортежа индикаторов k для некоторого обучающего примера $q = (k, k_L)$ на два кортежа и разделим весь набор возможных значений индикаторов из этого набора $K_{(\tilde{L})} = \prod_{i \in \tilde{L}} K_i$ в набор не пересекающихся подмножеств $K_{\tilde{L}}(Q) = \{\mu \subseteq K_{(\tilde{L})}\}$ такой, что $\forall \{q, \tilde{q}\} \subseteq Q$ если $k_{(\tilde{L})} \in \mu$ и $\tilde{k}_{(\tilde{L})} \in \mu$, тогда $k_L = \tilde{k}_L$ ($f_Q(k) = f_Q(\tilde{k})$) и для любых $\{\mu, \tilde{\mu}\} \subseteq K_{\tilde{L}} \exists \{q, \tilde{q}\} \subseteq Q$ так, что $k_{(\tilde{L})} \in \mu$, $\tilde{k}_{(\tilde{L})} \in \tilde{\mu}$ и $k_L \neq \tilde{k}_L$ ($f_Q(k) \neq f_Q(\tilde{k})$). Обозначим каждое такое множество μ как группу

эквивалентности. Конструкция $K_{\tilde{L}}(Q)$ – композиция множеств эквивалентности для подмножества индикаторов \tilde{L} .

Очевидно, что $|K_{\tilde{L}}(Q)| \in \{1, \dots, \prod_{i \in \tilde{L}} |K_i|\}$. Также видно, что если для некоторого Q и $\tilde{L} \subseteq L$ верно $|K_{\tilde{L}}(Q)| = 1$, то любая переменная из этого подмножества несущественна для Q (или f_Q). И чем выше $|K_{\tilde{L}}(Q)|$, тем более чувствителен Q к данному набору индикаторов. Также стоит отметить, что $|K_L(Q)| = K_L$ в то время как для некоторого $\tilde{L} \subseteq L$ может быть $|K_{\tilde{L}}(Q)| > K_L$.

Рассмотрим несколько примеров подсчета групп эквивалентности. Пример обучающего набора Q на трех индикаторах:

Таблица 2 – Полный обучающий набор на трех индикаторах.

q	Параметры			Оценка
	l_1	l_2	l_3	
1	0	0	0	0
2	1	0	0	1
3	0	1	0	1
4	1	1	0	0
5	0	0	1	0
6	1	0	1	0
7	0	1	1	0
8	1	1	1	0

В таблице 3 приведены множества эквивалентности для этого примера.

Таблица 3 – Группы эквивалентности для полного набора на трех индикаторах.

\tilde{L}	l_1	l_2	l_3	$l_1 l_2$	$l_1 l_3$	$l_2 l_3$
$ K_{\tilde{L}}(Q) $	2	2	2	2	3	3

На основе таблицы 4 рассмотрим обучающий набор Q с четырьмя индикаторами:

Таблица 4 – Полный обучающий набор на четырех индикаторах.

q	Параметры				Оценка
	l_1	l_2	l_3	l_4	
1	0	0	0	0	0
2	1	0	0	0	1
3	0	1	0	0	1
4	1	1	0	0	0
5	0	0	1	0	0

Продолжение таблицы 4

q	l_1	l_2	l_3	l_4	Оценка
6	1	0	1	0	0
7	0	1	1	0	0
8	1	1	1	0	0
9	0	0	0	1	0
10	1	0	0	1	0
11	0	1	0	1	0
12	1	1	0	1	0
13	0	0	1	1	0
14	1	0	1	1	0
15	0	1	1	1	0
16	1	1	1	1	0

Для этого примера у нас есть следующие группы эквивалентности:

Таблица 5 – Группы эквивалентности для полного набора на четырех индикаторах.

\tilde{l}_i	l_1	l_2	l_3	l_4	l_1l_2	l_1l_3	l_1l_4
$ K_{\tilde{l}_i}(Q) $	2	2	2	2	2	3	3
\tilde{l}_i	l_2l_3	l_2l_4	l_3l_4	$l_1l_2l_3$	$l_1l_2l_4$	$l_1l_3l_4$	$l_2l_3l_4$
$ K_{\tilde{l}_i}(Q) $	3	3	2	2	2	3	3

Как можно видеть в таблице выше, возможно, что $|K_{l_1}(Q)| = 2$ и $|K_{l_3}(Q)| = 2$, но $|K_{l_1l_3}(Q)| = 3$. Также $|K_{l_1l_2}(Q)| = 2$, $|K_{l_3l_4}(Q)| = 2$ и $|K_{l_2}(Q)| = 2$, $|K_{l_4}(Q)| = 2$, но, $|K_{l_2l_3l_4}(Q)| = 3$, когда $|K_{l_1l_2l_3}(Q)| = 2$, $|K_{l_1l_2l_4}(Q)| = 2$. На рисунке 9 мы представляем полученные группы через реализованные полные двоичные деревья на множестве Q из таблицы 5.

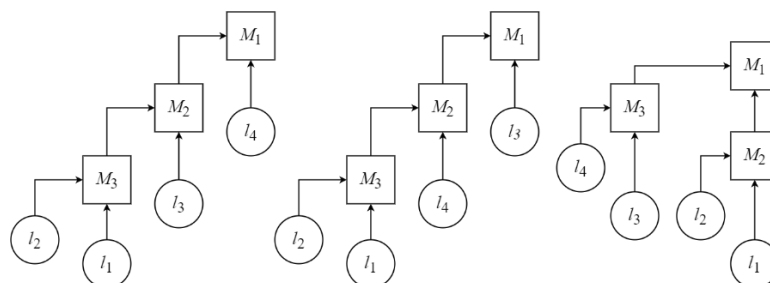


Рисунок 9 – Полные бинарные деревья, реализующие рассматриваемый обучающий набор Q на четырех индикаторах.

Теперь применим введенные выше понятия к проблеме реализации МКО.

Для любого произвольного $G \in \Gamma_2(L)$ обозначим его структуру декомпозиции показателей через $\Lambda(G) = \{L_i\}_{i \in \{1, \dots, l-1\}}$ так, что $\forall i \in \{1, \dots, l-1\} L_i \subseteq L$ набор листьев (индикаторов) поддерева с корнем в i . Тогда $L_1 = L$, и $\forall i \in \{1, \dots, l-1\} |L_i| \geq 2$.

Очевидно, что нет явной монотонности $K_{\tilde{L}}$ групп показателей по вложенности. Но об эквивалентности можно с уверенностью сказать, что если $\tilde{L} \subseteq L |K_{\tilde{L}}| = 1$, то $\forall \hat{L} \subseteq \tilde{L} |K_{\hat{L}}| = 1$. Кроме того, верны следующие утверждения.

Утверждение 1. Пусть дан некоторый полный набор Q и $G \in \Gamma_2(L)$. Тогда для любого $L_i \in \Lambda(G)$ такого что $|L_i| \geq 2$ и его подгруппы $\{L_{ir}; L_{ic}\} \subset \Lambda(G): L_{ir} \cup L_{ic} = L_i$. Для любых допустимых значений индикаторов из $L_{ir} - k_{(L_{ir})}$, $\tilde{k}_{(L_{ir})}$ и из $L_{ic} - k_{(L_{ic})}$, $\tilde{k}_{(L_{ic})}$ таких что $\{k_{(L_{ir})}, \tilde{k}_{(L_{ir})}\} \in \mu$, где $\mu \in K_{L_{ir}}(Q)$, и $\{k_{(L_{ic})}, \tilde{k}_{(L_{ic})}\} \in \hat{\mu}$, где $\hat{\mu} \in K_{L_{ic}}(Q)$ следует, что значения соответствующих показателей из L_i $k_{(L_i)} = (k_{(L_{ic})}, k_{(L_{ir})})$ и $\tilde{k}_{(L_i)} = (\tilde{k}_{(L_{ic})}, \tilde{k}_{(L_{ir})})$ принадлежат одним и тем же группам эквивалентности $\{k_{(L_i)}, \tilde{k}_{(L_i)}\} \in \mu'$. \square

Утверждение 2. В случае единой шкалы для полного обучающего набора Q выполнено $IRM_{L,G}(Q) \neq \emptyset$ тогда и только тогда, когда $\forall i \in \{1, \dots, l-1\}$ верно $|K_{L_i}(Q)| \leq |K_L|$. \square

Утверждение 2 позволяет построить такой алгоритм выбора $G \in \Gamma_2(L)$, что $IRM_{L,G}(Q) \neq \emptyset$, в случае если Q реализуем. \square

Прежде всего, проведение анализа групп эквивалентности позволяет нам удалить из рассмотрения все структуры $G \in \Gamma_2(L)$, в которые входят комбинации листьев с количеством групп эквивалентности, превышающим шкалу $|K_{\tilde{L}}(Q)| \geq K_L$ (см., например, [95]). Это позволяет нам резко сократить количество структур, которые необходимо учитывать при любом алгоритме синтеза МКО для конкретных $G \in \Gamma_2(L)$. Проведение проверки на эквивалентность одного подмножества показателей \tilde{L} пропорционально $(K_L)^l$. Во-вторых, основываясь на группах эквивалентности, удовлетворяющих шкале $|K_{\tilde{L}}(Q)| \leq K_L$, мы можем использовать алгоритм отбора заведомо реализуемых структур для случая полностью определенной функции. В области декомпозиции функции, заданной в табличном виде, для задач типа проверки количества групп эквивалентности широко распространена практика использования эвристики на основе задачи раскраски графа, предложенной М. Perkowski [116]. В рамках данного исследования были предложены два метода подсчета числа групп эквивалентности: эвристика и метод точного решения, изложенные в разделе 2.4. Численные эксперименты показали, что метод основанный на переформулировке задачи анализа групп эквивалентности как задачи целочисленного программирования, для исследованных практических задач и использования решателя Gurobi, выполняется за удовлетворительное время. Время расчета числа групп эквивалентности для каждой из подгрупп листьев на примерах, приведенных в главе 4, не превышало нескольких секунд, что является приемлемой скоростью вычисления на настольном ПК.

2.4. Алгоритмы поиска числа групп эквивалентности

Таблица данных, с помощью которой задана рассматриваемая дискретная функция, для заданного разбиения пространства входных параметров, преобразуется к представлению, называемому карта Карно (см., например, [71]). Основа метода получения групп эквивалентности заключается в получении числа групп поэлементно различных столбцов или строк (в зависимости от исследуемого разбиения) для составленной карты Карно (см., например [55]). Например, для примера из таблицы 2 карта Карно представлена в таблице 6. По строкам размещены возможные значения переменной l_1 , по столбцам – возможные значения группы переменных l_2l_3 , а в ячейках размещены соответствующие значения переменных значения функции. Как показано в таблице 3, для данного разбиения число групп эквивалентности равно 3. Ниже представлены разработанные алгоритмы для поиска числа групп эквивалентности.

Таблица 6 – Карта Карно для полного набора данных, заданного таблицей 2.

F		l_2l_3			
		00	01	10	11
l_1	0	0	0	1	0
	1	1	0	0	0

Первый предложенный алгоритм представляет собой жадный алгоритм поиска числа групп эквивалентности. В рамках метода вопрос поиска числа групп эквивалентности рассматривается с точки зрения склеиваемости столбцов, но может рассматриваться и склейка строк в зависимости от исследуемого разбиения. Численный алгоритм получения групп эквивалентности на основе жадной эвристики, описан ниже. Для полных наборов данных склеиваемость столбцов означает их идентичность. Для неполных – можно приклеивать рассматриваемый столбец к группе, если он не имеет ни в одной своей ячейке числа, отличающегося от чисел в соответствующих ячейках столбцов группы. Данный алгоритм имеет полиномиальную временную сложность от числа столбцов, однако, полученное число групп эквивалентности может зависеть от порядка столбцов карты Карно.

Опишем алгоритм.

Имеем: карту Карно (КК) для некоторого разбиения входных параметров, с числом столбцов n ; диапазон значений порядковой шкалы $scale$.

Получаем: количество групп эквивалентности.

1. Создается список групп.
2. Для каждого из n столбцов КК:
 - а. Если в нем больше одного символа отличного от пустого значения:

- i. Для каждого значения порядковой шкалы *scale*:
 - 1. Отмечаются позиции, на которых расположены группы одинаковых символов порядковой шкалы. Пустые позиции игнорируются.
 - 2. Создается вектор-карта позиций занятых рассматриваемыми значениями порядковой шкалы.
- ii. Если рассматриваемый вектор-карта склеивается/совпадает с каким-то из имеющихся в базе, то рассматриваемый столбец относится к имеющейся в базе группе.
- iii. Иначе, ему присваивается новая группа в списке групп.

b. Иначе переходим к следующему столбцу.

3. Возвращаем список групп.

Основа второго алгоритма в переформулировке задачи склейки столбцов в виде задачи раскраски графа. Каждый столбец представляется в виде вершины графа, а связи между вершинами обозначают несклеиваемость столбцов. Таким образом, теперь задача состоит в поиске минимального числа цветов, в которое можно раскрасить полученный граф при условии, что двум смежным вершинам соответствуют разные цвета. Для данной задачи существуют подходы, основанные на эвристиках и на точных методах. Ниже описывается точный алгоритм, предложенный в рамках данной работы. Для данного алгоритма в худшем случае имеем экспоненциальную временную сложность.

Имеем: карту Карно (КК) для некоторого разбиения входных параметров с числом столбцов n ; диапазон значений порядковой шкалы dim .

Получаем: количество групп эквивалентности.

1. Создается список групп, пустая матрица смежности размерности $n \times l$.
2. Для каждого из столбцов КК:
 - a. Ищем столбцы КК, с которыми рассматриваемый столбец не склеивается по определенным ячейкам.
 - b. Если нашли, то ставим 1 на пересечении соответствующих строки и столбца матрицы смежности. Таким образом составляется граф конфликтов.
3. Получаем k – максимальную степень вершины графа конфликтов.
4. Формируем вектор u длины k использованных цветов.
5. Формируется матрица X раскраски вершин, размерности $n \times k$, x_{ij} – элемент матрицы.
6. Формируются ограничения:
 - a. Каждая вершина окрашивается только в один цвет: $\sum_0^{n-1} x_i == 1$.
 - b. Отслеживания использованных цветов:
 - i. Для i в диапазоне от 0 до $n-1$:

1. Для j в диапазоне от 0 до $k-1$:
 - а. $x_{ij} \leq y_j$.
- с. Смежные вершины имеют различные цвета:
 - і. Для u в диапазоне от 0 до $n-1$:
 1. Для v в диапазоне смежных с u вершин:
 - а. Если $v > u$:
 - і. Для j в диапазоне от 0 до $k-1$:
 1. $x_{uj} + x_{vj} \leq 1$.
7. Целевая функция: $\min \sum_0^{k-1} y_i$.
8. Полученная задача линейного программирования решается одним из точных методов, например симплекс методом.

2.5. Таблица ветвей

В случае если стоит задача поиска всех структур на l листьях из $\Gamma_2(L)$, удобно воспользоваться механизмом проверки групп эквивалентности, описанным в разделе 2.3. На основе результатов анализа, число разбиений листьев на каждом уровне можно снизить, отсеив комбинации листьев, заведомо не реализующиеся в заданной шкале k_L (см., например, [53]). Полученные результаты удобно свести в таблицу ветвей. Таблица ветвей – это компактная, поуровневая запись комбинаций листьев для исследования на реализуемость в рамках метода декомпозиции. В верхней части таблицы располагаются названия столбцов, соответствующие группам листьев L_i для заданного L . Рассматриваются группы, состоящие от двух до $|L|-1$ листьев. Столбцы пронумерованы по порядку. Поля таблицы λ_i - содержат все возможные разбиения для группы листьев данного столбца. Для заполнения таблицы ветвей нужно создать столбец, для каждой из рассматриваемых групп листьев, на основе L . Затем для каждого из столбцов, заполнить поля таблицы значениями допустимых, в соответствии с правилами записи из раздела 2.1, разбиений листьев.

Если в результате анализа групп эквивалентности, в рамках заданной шкалы K_L , оказались допущенными некоторые комбинации листьев, то сборку таблицы ветвей начинаем с групп состоящих из двух листьев $|L_i| = 2$. Использование только разрешенных комбинаций листьев сокращает количество рассматриваемых структур. Таким образом, мы составляем список проверенных групп из трех листьев. Если групп состоящих из двух листьев нет среди разрешенных, то проводить дальнейший синтез не имеет смысла, т.к. это означает, что в рамках заданной шкалы невозможен синтез ни одной терминальной матрицы, т.е. матрицы, принимающей значения двух листьев. Допущенные группы заносим в таблицу ветвей. На основе разрешенных групп,

состоящих из двух и отдельных листьев, составляем комбинации, состоящие из трех листьев $|L_i| = 3$, полученные ветви заносим в таблицу, в колонку, соответствующую названию группы, которую составляют рассмотренные ветви. Подобный подход встречается в литературе по названию снизу-вверх или bottom-up (см. например [30]). На основе разрешенных групп, состоящих из трех, двух и отдельных листьев, составляем комбинации с $|L_i| = 4$, так получаем синтезированные ветви, состоящие из четырех листьев $|L_i| = 4$. Важно отметить, что начиная с $|L_i| = 4$, в рассмотрение попадают те ветви, которые попали в результаты анализа групп эквивалентности, и были внесены в таблицу ветвей, т.е. (если рассматривать ветви $|L_i| > 2$) были синтезированы на основе допущенных ветвей с меньшим количеством листьев. Полученные ветви заносим в колонку таблицы, соответствующую названию группы, которую составляют рассмотренные ветви. Процедура продолжается до достижения корня дерева $|L_i| = l$. В результате любая такая структура $G \in \Gamma_2(L)$, что ее $\Lambda(G)$ входит в список разрешенных групп листьев, должна рассматриваться в задаче синтеза МКО для этого обучающего множества \mathcal{Q} .

Для демонстрации того, как работает таблица ветвей вместе с методом анализа групп эквивалентности, введем данные, представленные в таблице 7. Данный набор также называется «О Зените» [45] и в рамках примера 10 будет рассмотрен подробнее. В задаче рассматривается вопрос предсказания результата выступления футбольной команды в зависимости от четырех параметров: расположение в турнирной таблице относительно соперника, дома ли играется матч, пропускает ли матч кто-то из лидеров, идет ли дождь, см. таблицу 7. Всего имеется 8 из 16 возможных примеров. Далее, для формирования механизма, исходные данные были закодированы в бинарной шкале.

Таблица 7 – Исходные данные для задачи «О Зените».

Соперник	Играем	Лидеры	Дождь	Победа
выше	дома	на месте	да	нет
выше	дома	на месте	нет	да
выше	дома	пропускают	нет	нет
ниже	дома	пропускают	нет	да
ниже	в гостях	пропускают	нет	нет
ниже	дома	пропускают	да	да
выше	в гостях	на месте	да	нет
ниже	в гостях	на месте	нет	да

Синтезируем таблицу ветвей для набора данных из таблицы 7. На основе полученной таблицы 8 можно получить все структуры полного бинарного дерева на четырех листьях.

2.6. Методы генерации структур

Ранее мы ввели буквенно-цифровую мл-нотацию для представления структур полных бинарных деревьев. Теперь представим методы для генерации записей структур МКО, соответствующих правилам, введенным в разделе 2.1.

1. Метод генерации на основе базовой записи. Метод состоит из двух частей: а) алгоритм получения всех структур на неименованных листьях и б) алгоритм получения всех структур на именованных листьях. В первой части используется механизм перестановок матриц в рамках записи структуры, начиная с базовой записи, в которой номера матриц расположены по возрастанию, начиная слева, например: $M_1l_1M_2l_2M_3l_3l_4$. Алгоритм получения всех структур на именованных листьях применяется к результатам алгоритма получения всех структур алгоритм получения всех структур на неименованных листьях.

А. Алгоритм получения всех структур на неименованных листьях.

Алгоритм получения всех структур на неименованных листьях основывается на работе со словом, кодирующим МКО. В данном алгоритме производится кодирование матриц заглавными буквами, а листов прописными: M и l соответственно. К буквам добавляются номера соответствующих матриц и листов: $M_i, i \in [1; |L| - 1]$ и $l_j, j \in [1; |L|]$. Начальное слово – это слово, соответствующее последовательной структуре, в котором номера матриц и листьев возрастают от начала к концу слова. Например, $M_1l_1M_2l_2l_3$ – такие позиции матриц считаются исходными относительно любых изменений слова. Активная матрица – это матрица, с которой на данной итерации производится перестановка.

Подход построен на последовательном обходе множества всех возможных бинарных деревьев для заданного количества матриц и листов, и выбора из него структур, удовлетворяющих требованиям для полных бинарных деревьев.

Для заданного слова на каждой итерации осуществляется перестановка активной матрицы с листом, находящимся на позиции перед ней, т.е. ближе к началу слова, либо выбор новой активной матрицы, если перестановка невозможна.

Первоначально активной назначается матрица с наибольшим номером.

Перестановка делается если выполняются два условия:

1. Перед активной матрицей расположен лист.
2. За активной матрицей находятся меньше двух листов подряд.

Например, в слове $M_1l_1M_2l_2M_3l_3l_4$ для активной матрицы M_2 доступна перестановка.

Если перестановка произведена, то активной снова назначается матрица с наибольшим номером.

Если перестановка невозможна, то активной назначается матрица с меньшим номером, чем текущая и происходит переход к следующей итерации.

После каждой перестановки производится проверка: удовлетворяет ли полученное слово свойствам кодирования структур из раздела 2.1. Для этого с помощью рекурсивной функции производится обход дерева по полученному слову и сравнение длин ветвей поддеревьев. Если в одной из ветвей или подветвей обнаружится несоблюдение критериев структур, то функция вернет ошибку и данное слово не будет сохранено в результат работы алгоритма.

После проверки происходит переход к следующей итерации. Процесс завершается, когда активной выбирается матрица с номером 1.

Для повышения скорости выполнения можно перекодировать слово в вектор, при этом матрицам сопоставить 1, а листам 0.

К результатам описанного алгоритма применяется проверка на наличие зеркальных подветвей, чтобы избежать повторов в результатах. Например, структуры $M_1M_2l_1M_3l_2M_4l_3l_4M_5M_6l_5l_6M_7l_7l_8$ и $M_1M_2M_3l_1l_2M_4l_3l_4M_5l_5M_6l_6M_7l_7l_8$ – зеркальны.

Б. Алгоритм получения всех полных бинарных деревьев на именованных листьях.

Алгоритм получения всех полных бинарных деревьев на именованных листьях также основывается на работе со словом кодирующим МКО, он работает с результатами алгоритма получения всех структур. В данном алгоритме производится перестановка листьев, в рамках каждой из допустимых структур. А также производится кодирование матриц заглавными буквами, а листов прописными: M и l соответственно. К буквам добавляются номера соответствующих матриц и листов: $M_i, i \in [1; |L| - 1]$ и $l_j, j \in [1; |L|]$.

В данном алгоритме в рамках каждой рассматриваемой структуры позиции матриц и листьев в слове фиксированы. Суть алгоритма состоит в рассмотрении всех размещений листьев по позициям листьев данного слова и отборе размещений удовлетворяющих критериям полных бинарных деревьев на именованных листьях. Для этого вводится понятие пула: пул – это набор всех допустимых листьев для размещения на определенной позиции листа в слове. Пулы создаются только для позиций листьев, порядок матриц в слове остается неизменным. Метка – текущая позиция в слове, соответствующая выбранному пулу. В начале алгоритма метка ставится в начало слова и инициализируется пул первого листа. Далее она перемещается по позициям листьев по следующему правилу:

1. Если активный пул не пуст, то значение метки увеличивается.
2. Иначе – уменьшается.

Для позиции активного пула производится выбор значения из пула, которое размещается в текущей позиции, а также инициализация следующего пула. Выбор значения производится в

соответствии с правилами размещения листьев полных бинарных деревьев на именованных листьях:

1. На несимметричной позиции может располагаться лист с любым номером.

2. Номера листьев, расположенных на симметричных позициях, должны возрастать. Например, в слове $M_1M_2l_1l_2M_3l_3l_4$ листья в парах l_1l_2 и l_3l_4 верно располагаются с возрастанием номеров, т.к. они находятся на симметричных позициях относительно матриц M_2 и M_3 , но также и номера листьев под матрицей M_3 должны быть больше чем номера листьев под матрицей M_2 , т.к. эти пары позиций симметричны относительно матрицы M_1 . Для этого производится предварительная разметка симметричных матриц в слове, а последующая работа с пулами производится в зависимости от позиции метки в слове. Выбранное значение листа удаляется из пула и помещается в слово. Пул следующей позиции инициализируется листьями, которые еще не задействованы в текущем слове.

Слово продолжает обрабатываться до тех пор, пока первый пул не станет пустым. В результате слово сохраняется после выбора листа из последнего пула.

В результате на основе двух описанных этапов получаем все допустимые, в соответствии с правилами из раздела 2.1, структуры полных бинарных деревьев на именованных листьях, для заданного L . Описанный алгоритм опробован для генерации структур МКО с $|L| < 11$. В задачах, с которыми нам приходилось сталкиваться в процессе написания данной работы количество индикаторов не превышало данное ограничение.

В качестве развития, был предложен алгоритм генерации структур на основе результатов анализа групп эквивалентности.

2. Метод генерации структур полных бинарных деревьев на основе результатов анализа групп эквивалентности. Данный метод предназначен для формирования списка структур полных бинарных деревьев на именованных листьях, каждая из ветвей сформированных структур удовлетворяет заданной шкале. В качестве входных данных используются результаты анализа групп эквивалентности, представленные в виде таблицы, например таблица 3 и таблица 5. Формат таблицы результатов анализа групп эквивалентности: первый столбец содержит имена групп, второй содержит шкалу, в которой соответствующая группа реализуется. Данные в столбцах должны быть представлены в числовом виде. Таблица должна содержать названия столбцов. Ниже представлен алгоритм, лежащий в основе метода.

Имеем: таблица результатов анализа групп эквивалентности *table*, с числом строк n ; шкала *scale*.

Получаем: список структур полных бинарных деревьев, удовлетворяющий заданной шкале по каждой из подгрупп листьев.

1. Формируем из первого столбца *table* список *eqglist*. Оставляем строки подгрупп со шкалой, не превышающей заданную *scale*.
2. Создаем *listoflists* список списков для хранения сгенерированных подструктур.
3. Находим первую строку *a* списка *eqglist* с числом символов больше 2.
4. Для каждой из *i* строк таблицы *eqglist*, в диапазоне от *a* до *n-1*:
 - a. *Pattern = eqglist[i]*.
 - b. *j=0*.
 - c. **Пока** количество символов *eqglist[j] <* количества символов *pattern*:
 - i. **Если** *eqglist[j]* пересекается с *pattern*:
 1. Вычисляем разницу *remainder* между ними.
 2. **Если** длина *remainder >* 2:
 - a. Ищем вхождение *remainder* в другие строки *eqglist*.
 - b. **Если** нашли, составляем из *remainder* и *pattern* новую разрешенную подгруппу. Добавляем ее в *listoflists*.
 3. **Иначе**, составляем из *remainder* и *pattern* новую разрешенную подгруппу. Добавляем ее в *listoflists*.
 - ii. *j += 1*.
5. Возвращаем *listoflists* содержащий список всех разрешенных структур.

В результате выполнения данного метода, как и для первого метода, получаем допустимые, в соответствии с правилами из раздела 2.1, структуры полных бинарных деревьев на именованных листьях, для заданного *L*. Однако для данного метода синтезируются только те структуры, подгруппы листьев которых, на основе анализа групп эквивалентности, удовлетворяют заданной шкале.

2.7. Выводы по главе 2

В данной главе введен ряд инструментов необходимых для решения задачи синтеза МКО, таких как форма записи структуры полного бинарного дерева или унитарное кодирование. Однако центральной его частью являются методы работы со структурами полных бинарных деревьев. Предлагаемый метод анализа групп эквивалентности позволяет предсказать, возможен ли в принципе синтез МКО для некоторого полного обучающего набора, т.е. представима ли описываемая им дискретная многомерная функция в виде бинарного дерева, что позволяет осуществлять ее представление в компактной табличной форме и получать решения оптимизационных задач на основе полиномиальных (по размерам шкал и числу входных переменных) алгоритмов.

Вопрос синтеза МКО на основе полного набора данных относится к третьей главе. Однако утверждение 2 верно лишь в случае полного набора данных. Это приводит к проблемам при оценке количества групп эквивалентности на неполных наборах данных. Тем не менее, для неполных наборов данных метод анализа групп эквивалентности позволяет решить проблему факториального роста числа полных бинарных деревьев на именованных листьях за счет отсева структур, заведомо не реализуемых в рамках заданной шкалы. Как будет показано в главе 4, это может быть существенная доля от всех возможных структур.

Глава 3. Анализ и синтез матриц МКО

В данной главе описываются предложенные подходы к анализу и синтезу матриц МКО. Сперва предлагается метод синтеза матриц МКО для заданной структуры полного двоичного дерева на полном наборе данных. В разделах 3.2 – 3.3 предлагаются методы синтеза МКО для неполных наборов данных, приводятся необходимые утверждения и следствия. В разделе 3.2 описывается метод синтеза всех матриц заданной структуры МКО, а в разделе 3.3 предлагается подход к синтезу МКО с применением разделительной декомпозиции. Для предложенных методов рассматриваются примеры синтеза МКО. В разделе 3.4 предложены подходы к анализу свойств матриц МКО, синтезированных в процессе идентификации.

3.1. Синтез матриц МКО на основе полного набора данных

Применение метода анализа групп эквивалентности, из раздела 2.3, к полному набору данных обеспечивает однозначно заполняемые карты Карно для любого допустимого разбиения входных параметров. Это автоматически дает заполнение матриц МКО соответственно исследуемым группам листьев (см., например, [95]). Таким образом, если результаты анализа групп эквивалентности показывают реализуемость подгрупп, необходимых для построения рассматриваемой структуры, то на основе соответствующих карт Карно можно автоматически получить искомыми матрицы МКО. Процедура синтеза основана на последовательной модификации таблицы дискретных данных *table_data* и записи структуры *note* и состоит из двух основных шагов: на основе заданной записи структуры выделяются разбиения групп листьев, для которых проводится анализ групп эквивалентности. Результаты анализа представляются как столбцы значений в *table_data* обозначаемые M_i и соответствующие рассматриваемым разбиениям групп листьев. Полученные значения могут быть представлены как матрицы синтезируемого МКО. Ниже приведен алгоритм синтеза матриц на основе полного набора данных.

Имеем: заполненную таблицу дискретных данных на l параметрах *table_data*, запись структуры полного бинарного дерева на именованных листьях *note* (в соответствии с правилами из раздела 2.1).

Получаем: значения для $l-1$ матрицы рассматриваемой структуры.

1. Шкала значений функции *scale* извлекается из таблицы исходных данных.
2. **Пока** в *note* больше одного элемента:

- a. Для последней матрицы из *note*, с номером i , выбираем соответствующие ей листья на основе *note*. Для выбранных листьев на основе значений из *table_data* производим анализ групп эквивалентности, см. раздел 2.4.
- b. Результаты анализа групп эквивалентности заносим в таблицу данных *table_data* в качестве столбца с названием M_i . Значения в этой колонке являются значениями искомой матрицы M_i . Столбцы с использованными листьями удаляются.
- c. Удаляем использованные в предыдущем пункте матрицу и листья из *note*. Вместо них помещаем лист « l » с номером i . Значения этого листа берутся из столбца M_i таблицы *table_data*.
- d. Если количество групп эквивалентности меньше или равно *scale*:
 - i. Продолжаем.
- e. Иначе:
 - i. Выводится сообщение о невозможности синтеза МКО в заданной шкале.

3. Возвращаем синтезированные значения матриц для заданной структуры.

Пример 3 Рассмотрим синтез матриц для набора данных, заданного в таблице 11, на основе структуры $M_1 M_2 l_1 l_2 M_3 l_3 l_4$

Таблица 11 – Полный набор данных в двоичной шкале на четырех индикаторах, для задачи синтеза матриц.

q	Параметры				Оценка
	l_1	l_2	l_3	l_4	
1	0	0	0	0	0
2	1	0	0	0	0
3	0	1	0	0	1
4	1	1	0	0	0
5	0	0	1	0	1
6	1	0	1	0	1
7	0	1	1	0	1
8	1	1	1	0	1
9	0	0	0	1	0
10	1	0	0	1	0
11	0	1	0	1	1
12	1	1	0	1	0
13	0	0	1	1	0

Продолжение таблицы 11

q	l_1	l_2	l_3	l_4	Оценка
14	1	0	1	1	0
15	0	1	1	1	1
16	1	1	1	1	0

В соответствии с записью $M_3l_3l_4$ из $M_1M_2l_1l_2M_3l_3l_4$, на основе анализа групп эквивалентности для подгруппы листьев l_3l_4 синтезируется столбец M_3 . На основе анализа групп эквивалентности для подгруппы l_1l_2 в соответствии с $M_2l_1l_2$ синтезируется столбец M_2 . После этих двух шагов, запись структуры *note* преобразуется к виду: $M_1l_2l_3$. Столбец M_1 генерируется аналогично, но в качестве значений листьев l_2l_3 используются полученные значения M_3 и M_2 . В таблице 12, для наглядности, вместе с синтезированными значениями матриц, приведем также и использованные в процессе синтеза столбцы со значениями листьев несмотря на то, что в алгоритме предполагается удаление использованных столбцов. □

Таблица 12 – Результаты синтеза матриц на полном наборе данных, заданным таблицей 11.

q	Параметры				M_3	M_2	M_1	Оценка
	l_1	l_2	l_3	l_4				
1	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0
3	0	1	0	0	0	1	1	1
4	1	1	0	0	0	0	0	0
5	0	0	1	0	1	0	1	1
6	1	0	1	0	1	0	1	1
7	0	1	1	0	1	1	1	1
8	1	1	1	0	1	0	1	1
9	0	0	0	1	0	0	0	0
10	1	0	0	1	0	0	0	0
11	0	1	0	1	0	1	1	1
12	1	1	0	1	0	0	0	0
13	0	0	1	1	0	0	0	0
14	1	0	1	1	0	0	0	0
15	0	1	1	1	0	1	1	1
16	1	1	1	1	0	0	0	0

3.2. Синтез полного набора матриц для заданной структуры МКО

Унитарное кодирование может применяться к обучающим примерам так же, как это было описано разделе 2.2. Для некоторого обучающего примера $q = (k, k_L)$ мы будем обозначать унитарную версию через $\tilde{q} = (\tilde{k}, \tilde{k}_L)$, где \tilde{k}_L – унитарное представление k_L и \tilde{k} – кортеж k в унитарном представлении.

Если $w(\cdot) \in IRM_{L,2}$ реализует некоторый пример $q \in Q$, то $\tilde{w}(\tilde{k}) = \tilde{k}_L$. Из этого следует, что для того, чтобы найти МКО, который реализует некоторый обучающий пример, необходимо решить систему уравнений, полученных на основе унитарного представления этого МКО, с дополнительными ограничениями, вытекающими из унитарного представления (см., например, [16, 79]).

Пример 4. Рассмотрим $|L| = 3$, $|K_L| = 2$ и обучающий пример $q = ((0, 0, 0), 0)$, унитарная форма представлена в таблице 13.

Таблица 13 – Унитарное представление обучающего примера для примера 3.

	\tilde{l}_1	\tilde{l}_2	\tilde{l}_3	f
\tilde{q}	$\begin{pmatrix} 1 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 0 \end{pmatrix}$

Если МКО имеет структуру $\tilde{w}(\tilde{k}) = \tilde{M}_1 \tilde{l}_1 \tilde{M}_2 \tilde{l}_2 \tilde{l}_3$, то необходимо решить систему уравнений

$$(1) \begin{pmatrix} 1 \\ 0 \end{pmatrix}^T \begin{pmatrix} \begin{pmatrix} m2_{00}^0 & m2_{01}^0 \\ m2_{00}^1 & m2_{01}^1 \end{pmatrix} \\ \begin{pmatrix} m2_{10}^0 & m2_{11}^0 \\ m2_{10}^1 & m2_{11}^1 \end{pmatrix} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} \begin{pmatrix} m1_{00}^0 & m1_{01}^0 \\ m1_{10}^0 & m1_{11}^0 \end{pmatrix} \\ \begin{pmatrix} m1_{00}^1 & m1_{01}^1 \\ m1_{10}^1 & m1_{11}^1 \end{pmatrix} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

с унитарными условиями: $\forall \{i, j\} \in \{0, 1\}^2 \quad m1_{ij}^0 + m1_{ij}^1 = 1, \quad m2_{ij}^0 + m2_{ij}^1 = 1, \quad \forall t \in \{0, 1\} \quad m1_{ij}^t \in \{0, 1\}, \quad m2_{ij}^t \in \{0, 1\}$.

Аналогично операциям, описанным в примере 2, система уравнений (1) может быть последовательно упрощена:

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix}^T \begin{pmatrix} \begin{pmatrix} m2_{00}^0 & m2_{01}^0 \\ m2_{00}^1 & m2_{01}^1 \end{pmatrix} \\ \begin{pmatrix} m2_{10}^0 & m2_{11}^0 \\ m2_{10}^1 & m2_{11}^1 \end{pmatrix} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} \begin{pmatrix} m1_{00}^0 \\ m1_{00}^1 \end{pmatrix} \\ \begin{pmatrix} m1_{10}^0 \\ m1_{10}^1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$

$$\begin{pmatrix} m2_{00}^0 \\ m2_{00}^1 \end{pmatrix}^T \begin{bmatrix} \begin{pmatrix} m1_{00}^0 \\ m1_{00}^1 \end{pmatrix} \\ \begin{pmatrix} m1_{10}^0 \\ m1_{10}^1 \end{pmatrix} \end{bmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$

$$m2_{00}^0 \begin{pmatrix} m1_{00}^0 \\ m1_{00}^1 \end{pmatrix} + m2_{00}^1 \begin{pmatrix} m1_{10}^0 \\ m1_{10}^1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

И наконец, раскладывая по унитарным компонентам, имеем систему уравнений:

$$m2_{00}^0 m1_{00}^0 + m2_{00}^1 m1_{10}^0 = 1,$$

$$m2_{00}^0 m1_{00}^1 + m2_{00}^1 m1_{10}^1 = 0.$$

Из полученной системы уравнений видно, что решением являются любые матрицы M_1 и M_2 такие, что $\tilde{m}1_{00} = \tilde{m}2_{00} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$. Это означает, что будет достаточно определить $m1_{00} = m2_{00} = 0$ для реализации $q = ((0, 0, 0), 0)$ в МКО с унитарным представлением $\tilde{w}(\tilde{k}) = \tilde{M}_1 \tilde{l}_1 \tilde{M}_2 \tilde{l}_2 \tilde{l}_3$. \square

Приведенный выше пример также иллюстрирует приведенное ниже утверждение. Для произвольного $w(\cdot) \in IRM_{L,2}(Q)$ обозначим через $P(w, q)$ – функцию, которая находится в левой части уравнения с правой частью, равной 1. Вследствие унитарного подхода есть только одна такая функция для любого q .

Утверждение 3. Для любых $L \subset \mathbb{N}$, $K \subset \mathbb{N}$, МКО с единой шкалой $w(\cdot) \in IRM_{L,2}(Q)$ и любого возможного q в единой шкале: $P(w, q)$ – однородный полином степени $l-1$, который может быть представлен как сумма κ^{l-2} уникальных компонент:

$$P(w, q) = \sum_{j=1}^{\kappa^{l-2}} p_j, \quad \forall j \in \{1, \dots, \kappa^{l-2}\} \quad p_j = \prod_{i=1}^{l-1} m_i, \quad \forall i \in \{1, \dots, l-1\},$$

m_i – одна компонента кортежа в некоторой ячейке унитарно закодированной матрицы \tilde{M}_i ; $\{\tilde{M}_i\}_{i \in \{1, \dots, l-1\}}$ – набор всех унитарно закодированных матриц свертки МКО w ; $\kappa = |K|$ – шкала значений индикатора;

$$P(w, q) \in \{0, 1\};$$

$$P(w, q) = \tilde{w}(\tilde{k})^T \tilde{k}_L;$$

$$IRM \ w \text{ реализует } q \Leftrightarrow P(w, q) = 1. \quad \square$$

Доказательство утверждения 3.

В рамках введенной унитарной записи, операция в рамках МКО с отдельным набором значений индикаторов $\tilde{w}(\tilde{k})$ представляет из себя последовательность $l-1$ матричных операций

$\tilde{y}^T \tilde{M} \tilde{x}$, где векторы \tilde{x} , \tilde{y} имеют размерность $k + 1 \in \mathbb{N}$, матрица \tilde{M} – размерность $(k + 1) \times (k + 1)$.

Очевидно, что каждая такая операция представима в виде полинома

$$\tilde{y}^T \tilde{M} \tilde{x} = \sum_{r \in \{0, \dots, k\}} \tilde{y}^r \sum_{c \in \{0, \dots, k\}} \tilde{x}^c \tilde{m}_{rc}.$$

Соответственно, если во всех ячейках матрицы стоят векторы размерности $k + 1$, то результатом операции будет также вектор размерности $k + 1$, а любая его компонента $i \in \{0, \dots, k\}$ будет записываться как

$$\sum_{r \in \{0, \dots, k\}} \tilde{y}^r \sum_{c \in \{0, \dots, k\}} \tilde{x}^c \tilde{m}_{rc}^i.$$

То есть итоговым значением операции $\tilde{w}(\tilde{k})$ будет также вектор размерности $k + 1$. При этом каждая компонента этого вектора будет представима однородным полиномом степени $2l - 1$, определяемым $2l - 2$ операциями типа $\tilde{y}^T \tilde{x}$, т.е. иметь k^{2l-2} . Учитывая, что значения всех листьев заданы унитарными векторами, то итоговая степень полинома будет $l - 1$ – в нем сохранятся только те слагаемые, которые не умножаются на нулевые значения компонент векторов листьев, и каждое слагаемое будет иметь вид $\prod_{i=1}^{l-1} m_i$, m_i – одна компонента кортежа в некоторой ячейке унитарно закодированной матрицы \tilde{M}_i . При этом количество ненулевых слагаемых будет уже определяться $2l - 2 - 1$ операциями типа $\tilde{y}^T \tilde{x}$, т.е. их будет всего k^{l-2} . Уникальность каждого слагаемого также следует из сути описанной операции.

Учитывая, что в ячейках всех матриц должны стоять унитарные векторы, получаем, что каждая компонента вектора, определяемого операцией $\tilde{w}(\tilde{k})$, может принимать значения либо 0, либо 1.

Соответственно, то, что МКО реализует отдельно взятый пример, означает, что $\tilde{w}(\tilde{k}) = \tilde{k}_L$. Так как Вектор \tilde{k}_L является унитарным, то у вектора $\tilde{w}(\tilde{k})$ только одна компонента должна быть равна 1 – та же, что и у вектора \tilde{k}_L , все остальные должны быть равны 0. То есть $\tilde{w}(\tilde{k})^T \tilde{k}_L = 1$.

Поэтому, обозначив через $P(w, q)$ полином, соответствующий компоненте вектора, определяемой $\tilde{w}(\tilde{k})$ и которая должна равняться 1, получаем все совокупность пунктов данного утверждения. \square

Следствие 1. Для любых $L \subset \mathbb{N}$, $K \subset \mathbb{N}$ и любого возможного q в единой шкале $IRM_{L,2}(q) = Arg \max_{w \in IRM_{L,2}} P(w, q)$. \square

Доказательство следствия 1.

Из утверждения 3 следует, что для любых $L \subset \mathbb{N}$, $K \subset \mathbb{N}$ и любого возможного q в единой шкале для любой МКО w $P(w, q) \in \{0; 1\}$, а то, что некоторый МКО \tilde{w} реализует q , означает, что $P(\tilde{w}, q) = 1$. Отсюда следует, что $IRM_{L,2}(q) = \underset{w \in IRM_{L,2}}{Arg \max} P(w, q)$. \square

В дополнение к следствию, указанному выше, из утверждения 1 следует, что для синтеза $IRM_{L,G}(q)$ для некоторого бинарного дерева $G \in \Gamma_2(L)$ достаточно рассмотреть релаксированную проблему оптимизации:

$$(2) \max_{w \in IRM_{L,G}} P(w, q)$$

при ограничениях:

$$(3) \forall \{\tilde{M}_i\}_{i \in \{1, \dots, l-1\}}, \forall \tilde{m} \in \tilde{M}_i \tilde{m} \in [0; 1]^l, |m| = 1,$$

и любое решение этой задачи должно обеспечивать значение $P(w, q) = 1$.

Чтобы аппроксимировать некоторый обучающий набор $Q \subset K \otimes K_L$ по критерию, представленному в разделе 2, необходимо найти такой МКО, который реализует максимальное число обучающих примеров из этого набора. С учетом (2) и (3) можно сформулировать следующее утверждение.

Утверждение 4. Для любых $L \subset \mathbb{N}$, $K \subset \mathbb{N}$ и любого возможного $Q \subset K^{l+1}$ с единой шкалой решение задачи аппроксимации МКО с единой шкалой состоит в решении задачи максимизации:

$$(4) \max_{w \in IPM_{L,G}} \sum_{q \in Q} P(w, q)$$

при ограничениях:

$$(5) \forall \{\tilde{M}_i\}_{i \in \{1, \dots, l-1\}}, \forall \tilde{m} \in \tilde{M}_i \tilde{m} \in [0; 1]^l, |m| = 1. \square$$

Доказательство утверждения 4.

Так как задача аппроксимации в рамках данной статьи определена как реализация как можно большего числа примеров из обучающего набора, то из записи оптимизационной задачи для реализации отдельного примера в виде (2), (3) следует, что запись (4), (5) соответствует данной формулировке задачи аппроксимации. \square

Следствие 2. Для любых $L \subset \mathbb{N}$, $K \subset \mathbb{N}$ и любой $Q \subset K^{l+1}$ с единой шкалой может быть реализован с помощью МКО с единой шкалой тогда и только тогда, когда существует $w \in IRM_{L,2}(Q)$, такое что $\sum_{q \in Q} P(w, q) = |Q|$. \square

Доказательство следствия 2 вытекает очевидным образом из определений задач реализуемости и аппроксимации.

Проиллюстрируем, как результаты разработанного подхода могут быть реализованы в отдельных случаях. Как уже упоминалось ранее, мы ограничимся рассмотрением примеров

с двоичной логикой. Начнем с типичного примера проверки подходов к обучению ИИ – реализации *XOR* (см., например, [111]).

Пример 5. Булева функция *XOR* $f(x_1, x_2) = (x_1 \wedge \bar{x}_2) \vee (\bar{x}_1 \wedge x_2)$. Имеем $|I| = 2$ и $|K| = 2$, полный набор обучающих примеров (его таблица истинности) в унитарной форме представлен в таблице 14.

Таблица 14 – Обучающий набор в унитарном представлении для примера 4.

q	\tilde{x}_1^T	\tilde{x}_2^T	\tilde{f}^T
1	(1, 0)	(1, 0)	(1, 0)
2	(0, 1)	(0, 1)	(1, 0)
3	(1, 0)	(0, 1)	(0, 1)
4	(0, 1)	(1, 0)	(0, 1)

Существует только одна допустимая структура МКО – $\tilde{M}\tilde{x}_1\tilde{x}_2$. Необходимо решить систему уравнений в матричной форме $\forall q \in Q$:

$$\begin{pmatrix} \tilde{x}_2^0(q) \\ \tilde{x}_2^1(q) \end{pmatrix}^T \begin{bmatrix} \begin{pmatrix} m_{00}^0 \\ m_{01}^1 \end{pmatrix} & \begin{pmatrix} m_{01}^0 \\ m_{11}^1 \end{pmatrix} \\ \begin{pmatrix} m_{10}^0 \\ m_{11}^1 \end{pmatrix} & \begin{pmatrix} m_{11}^0 \\ m_{11}^1 \end{pmatrix} \end{bmatrix} \begin{pmatrix} \tilde{x}_1^0(q) \\ \tilde{x}_1^1(q) \end{pmatrix} = \begin{pmatrix} \tilde{f}^0(q) \\ \tilde{f}^1(q) \end{pmatrix}.$$

Из нее следует, что $\forall w(\cdot) \in IRM_{L,2}$ верны равенства $P(w, q^1) = m_{00}^0$, $P(w, q^2) = m_{11}^0$, $P(w, q^3) = m_{10}^1$, $P(w, q^4) = m_{01}^1$. Поэтому существует только одно допустимое решение с $\sum_{q \in Q} P(w, q) = 4$, для него

$$\tilde{M} = \begin{bmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \begin{pmatrix} 0 \\ 1 \end{pmatrix} & \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{bmatrix} \Rightarrow M = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \square$$

Такая матрица M соответствует функции *XOR*.

На основании приведенного выше примера можно сформулировать алгоритм синтеза всех матриц заданной структуры.

Входные данные: предварительно подготовленная таблица дискретных данных на l параметрах и структура или список структур, шкала значений функции (опционально).

Выходные данные: значения для $l-1$ матрицы рассматриваемой структуры.

1. **Пока** есть структуры в списке структур:
 - a. В соответствии с утверждением 3, на основе таблицы данных и текущей структуры, генерируются однородные полиномы, для каждого из примеров входных данных.
 - b. В соответствии с утверждением 4, производится формирование единого оптимизационного функционала (ОФ) на основе полученных полиномов.
 - c. На основе составленного ОФ и ограничений унитарности формулируется задача для решателя.
2. Вывод результатов.

3.3. Синтез матрицы для всех поддеревьев данного разбиения пространства входных переменных

В случаях, когда параметров в предоставленном наборе данных слишком много, для синтеза полного набора матриц на основе некоторой структуры МКО можно прибегнуть к методу декомпозиции. Многие исследователи проявляли интерес к возможности декомпозиции функции. Например, возможность разложения любой непрерывной функции n переменных в суперпозицию непрерывных функций меньшего числа переменных, исследовалась А.Н. Колмогоровым и В.И. Арнольдом, в частности для двух, доказана в работах [7, 35]. Со стороны дискретной функции В.С. Выхованец [22] исследовал построение декомпозиции алгебраических функций, а также задачу синтеза дискретной системы с помощью спектрального разложения (см, например [23]). В [40] А.В. Кузнецовым исследован вопрос неповторных булевых функций. В области МКО важно отметить работу из области многокритериальной оценки В.А. Глотова и В.В. Павельева [26], в которой освещается использование декомпозиции для построения критериально-целевой структуры. Более того, в [26] доказывается теорема о представимости дискретной функции n переменных в бинарном (разделительном) виде. В.Н. Бурков и И.В. Буркова с коллегами исследовали дихотомическое представление функции [21, 12] с точки зрения решения задач дискретной оптимизации, в том числе и применительно к задаче комплексного оценивания.

3.3.1. Декомпозиция дискретной функции

В разделе 3.2 был предложен подход к синтезу МКО с использованием механизма обучения на дискретных данных, в центре которого находится механизм составления оптимизационного функционала на основе входных данных и предложенной структуры полного бинарного дерева. Такой подход к задаче синтеза сопряжен с трудностями в решении оптимизационной задачи в

случае большой размерности входных данных. Степень оптимизационного полинома растет линейно в зависимости от количества входных параметров, а число общих ограничений оптимизационной задачи растет экспоненциально: $Cn = \kappa^{l-2} ex_num$, где Cn – число общих ограничений, $\kappa = |K|$ – шкала значений индикатора, для параметров в единой шкале k_L , ex_num – количество примеров в наборе Q , а l – число параметров.

В качестве альтернативы была исследована возможность решения задачи синтеза по шагам с помощью разделительной декомпозиции функции (см., например, [52, 53]) $f(X) = \Sigma(X_1, a(X_2))$, где Σ, a – некоторые функции, а $X_1 \cap X_2 = \emptyset$ – подмножества получающиеся в результате разделения X , см. [22]. Для случая МКО, т.е. с использованием полных бинарных деревьев легко показать, что при условии отсутствия фиксирования шкал значений $k_i \in K_i$ декомпозиция заданной функции возможна в любой структуре полного бинарного дерева, т.к. в силу конечного k_L шкала значений функций на которые раскладывается φ_i , тоже конечна, $\forall i \in \{1, \dots, l-1\}$. Таким образом, имеет место последовательный подход к составлению и решению задач синтеза для декомпозиции в каждом из узлов рассматриваемого дерева из $\Gamma_2(L)$.

Доказательство представимости некоторой дискретной функции l переменных в разделительном виде при условии отсутствия фиксирования шкал значений параметров.

В некоторой структуре полного бинарного дерева, МКО определяется через набор матриц свертки $M_f = \{M_j\}_{j \in L}$. В рамках рассматриваемой структуры МКО, на основе каждой M_i выполняется операция свертки: $\tilde{M}_i \tilde{x}_i \tilde{y}_j$. Каждая отдельная M_i реализует некоторую дискретную функцию $\varphi_i(k_{(L_i)})$. Для любого произвольного $G \in \Gamma_2(L)$, структура декомпозиции его показателей $\Lambda(G) = \{L_i\}_{i \in \{1, \dots, l-1\}}$, так что $\forall i \in \{1, \dots, l-1\} L_i \subseteq L$ набор листьев (индикаторов) поддерева с корнем в i . L_i имеет некоторые подгруппы $\{L_{ir}; L_{ic}\} \subset \Lambda(G): L_{ir} \cup L_{ic} = L_i$. В общем случае, неизвестно какая размерность \tilde{M}_i потребуется для реализации $\varphi_i(k_{(L_i)})$ на разбиении $\lambda_i = (k_{(L_{ir})}, k_{(L_{ic})})$. Основываясь на методе анализа групп эквивалентности, число групп эквивалентности для разбиения λ_i , а значит и соответствующая размерность \tilde{M}_i , не может быть больше, чем число комбинаций кодируемое индикаторами подмножества $K_{(L_{ir})} = \prod_{j \in L_{ir}} K_j$, $K_{(L_{ic})} = \prod_{j \in L_{ic}} K_j$ – для подмножеств индикаторов строк и столбцов соответственно. Может потребоваться разместить в M_i по различным ячейкам примеры, каждый из которых соответствует отдельной группе эквивалентности данного шага декомпозиции. Т.е. в худшем случае, для каждого из обучающих примеров необходимо предусмотреть отдельную ячейку матрицы. Иначе говоря, любое подмножество переменных для дискретной функции содержит конечное число комбинаций значений переменных, это число является оценкой максимального значения числа строк или столбцов матрицы.

Учитывая сказанное, размерность некоторой $\varphi_i(k_{(L_i)})$ будет не больше чем $K_{(L_i)}, L_{ir} \cup L_{ic} = L_i$. Таким образом, на каждой из ступеней декомпозиции f размерность матрицы $\tilde{M}_i, i \in l-1$, достаточна для реализации $\varphi_i(k_{(L_i)})$, по построению. Такой подход к синтезу матриц гарантирует, что МКО, построенная на основе любой из разрешенных структур полных двоичных деревьев, будет реализовывать заданную функцию f .

Пользуясь обозначениями введенными в разделе 2.3, для некоторого подмножества индикаторов $\tilde{L} \subseteq L$, обозначим через $\lambda = (k_{(\tilde{L})}, k_{(L \setminus \tilde{L})})$ разбиение кортежа индикаторов k для некоторого обучающего примера $q = (k, k_L)$ на два кортежа, $\lambda_i = (k_{(L_{ir})}, k_{(L_{ic})})$ разбиение кортежа индикаторов в i -м узле дерева G . В каждом из рассматриваемых узлов дерева будем называть функции компоненты, на которые раскладывается $\varphi_i(k_{(L_i)})$ в соответствии с разбиением и в соответствии с нумерацией матриц, расположенных в узлах структуры рассматриваемого дерева, на которых реализуются функции составляющие $\varphi_i(k_{(L_i)})$. Например, для разбиения $\lambda = (k_{(1,2)}, k_{(3,4)})$ названия компонент двух подфункций будут записываться, так: $\varphi_{i+1} - k_1 k_2$ и $\varphi_{i+2} - k_3 k_4$. Также для удобства изложения, можем именовать компоненты $\varphi_i(k_{(L_i)})$ в соответствии с λ_i как $\varphi_r(k_{(L_r)})$ и $\varphi_c(k_{(L_c)})$. В случае если составляется разбиение, состоящее из отдельного листа и группы: $\lambda = (k_{(1)}, k_{(3,4)})$, кодируем только $\varphi_{i+1} - k_3 k_4$, где $i \in \{1, \dots, l-1\}$. Рассмотрим пример в контексте предлагаемого подхода. \square

Пример 6. Рассмотрим $|L| = 3, |K_L| = 2$ и обучающий пример $q = ((0, 0, 0), 0)$, унитарное представление: $\tilde{q} = ((\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}), \begin{pmatrix} 1 \\ 0 \end{pmatrix})$.

Рассмотрим в первую очередь реализуемость некоторой функции $f(\tilde{l}_1, \tilde{l}_2, \tilde{l}_3) = \varphi_1(\tilde{l}_3, \varphi_2(\tilde{l}_1, \tilde{l}_2))$. Для именования функций декомпозиции используем $\lambda_1 = (k_{(1)}, k_{(2,3)})$, тогда система уравнений в матричной выглядит как:

$$\begin{pmatrix} \left(\begin{pmatrix} \varphi_{2-00}^0 \\ \varphi_{2-00}^1 \end{pmatrix} \right)^T \left[\begin{array}{cc} \begin{pmatrix} m_{00}^0 \\ m_{00}^1 \end{pmatrix} & \begin{pmatrix} m_{01}^0 \\ m_{01}^1 \end{pmatrix} \\ \begin{pmatrix} m_{10}^0 \\ m_{10}^1 \end{pmatrix} & \begin{pmatrix} m_{11}^0 \\ m_{11}^1 \end{pmatrix} \end{array} \right] \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

с унитарными условиями: $\forall \{i, j\} \in \{0, 1\}^2 \quad m_{ij}^0 + m_{ij}^1 = 1 \quad m_{ij}^t \in \{0, 1\}, \quad \varphi_{2-00}^0 + \varphi_{2-00}^1 = 1, \quad \varphi_{2-00}^t \in \{0, 1\}, \quad \forall t \in \{0, 1\}$. В результате несложных преобразований получаем:

$$\begin{aligned} \varphi_{2-00}^0 m_{00}^0 + \varphi_{2-00}^1 m_{10}^0 &= 1, \\ \varphi_{2-00}^0 m_{00}^1 + \varphi_{2-00}^1 m_{10}^1 &= 0. \end{aligned}$$

Решение данной системы уравнений легко найти используя ограничения бинарности, например: $\varphi_{2_{-00}}^0 = 1$, $m_{00}^0 = 1$. Соответственно имеем часть оптимизационной задачи, соответствующую примеру $q = ((0, 0, 0), 0)$. \square

Обозначим через $P(\lambda_i, q)$ функцию, которая находится в левой части уравнения с правой частью, равной 1. Вследствие унитарного подхода есть только одна такая функция для любого q . Через Q_i обозначен набор данных полученный из исходного Q путем выделения только тех столбцов, на которых определена $\varphi_i(k_{(L_i)})$.

Утверждение 5. Для любых $L \subset \mathbb{N}$, $K \subset \mathbb{N}$, и любого возможного q в единой шкале: $P(\lambda_i, q)$ – однородный полиномом степени меньше или равной 3, который может быть представлен как сумма κ^{φ_num} уникальных компонент:

$$P(\lambda_i, q) = \sum_{j=1}^{\kappa^{\varphi_num}} p_j, \quad \forall j \in \{1, \dots, \kappa^{\varphi_num}\}, \quad p_j = m_j \prod_{d=1}^{\varphi_num} \varphi_d, \quad \forall d \in \{1, \dots, \varphi_num\},$$

где φ_d – обозначение компонент функций, на основе которых декомпозируется φ_i , m_j – одна компонента кортежа в некоторой ячейке унитарно закодированной матрицы \tilde{M}_i , q – пример из Q_i , $\varphi_num = 1$ при подсоединении к матрице ветви и листа, $\varphi_num = 2$ при подсоединении к матрице пары ветвей;

$$P(\lambda_i, q) \in \{0, 1\};$$

$$\varphi_i(k_{(L_i)}) \text{ реализует } q \Leftrightarrow P(\lambda_i, q) = 1. \quad \square$$

Доказательство утверждения 5.

В рамках введенной унитарной записи операция в рамках МКО с отдельным набором значений индикаторов представляется как некоторая пошагово декомпозируемая, в соответствии с некоторым набором разбиений $\Lambda(G)$ функция $f(\tilde{l}_1, \dots, \tilde{l}_{|L|})$. На каждом шаге декомпозиции, т.е. в каждом из узлов рассматриваемого дерева G , $\varphi_i(k_{(L_i)})$ определена на подмножестве листьев L_i , в соответствии с разбиением λ_i , на основе набора примеров Q_i , составленного из Q , путем отбора листьев соответствующих L_i . Соответственно, размерность $k_{(L_i)}$ определяется исходя из L_i , на которой определена $\varphi_i(k_{(L_i)})$. $\varphi_i(k_{(L_i)})$ представляет из себя матричную операцию на отдельно взятой матрице, $\tilde{y}^T \tilde{M} \tilde{x}$, где векторы \tilde{x} , \tilde{y} имеют размерность κ , матрица \tilde{M} – размерность $\kappa \times \kappa$. Очевидно, что каждая такая операция в качестве результата возвращает вектор компонент отдельной ячейки матрицы размерности $\kappa \in \mathbb{N}$. Пример операции на матрице M_i :

$$\begin{pmatrix} y^0(q^i) \\ y^1(q^i) \end{pmatrix}^T \begin{bmatrix} \begin{pmatrix} m_{00}^0 \\ m_{00}^1 \end{pmatrix} & \begin{pmatrix} m_{01}^0 \\ m_{01}^1 \end{pmatrix} \\ \begin{pmatrix} m_{10}^0 \\ m_{10}^1 \end{pmatrix} & \begin{pmatrix} m_{11}^0 \\ m_{11}^1 \end{pmatrix} \end{bmatrix} \begin{pmatrix} x^0(q^i) \\ x^1(q^i) \end{pmatrix}$$

То есть, итоговым значением операции $\tilde{y}^T \tilde{M} \tilde{x}$ будет также вектор размерности k . При этом каждая компонента этого вектора будет представима однородным полиномом степени 3, т.к. на каждом шаге рассматривается присоединение к матрице двух ветвей. Учитывая, что \tilde{x}, \tilde{y} могут быть как некоторыми функциями компонентами $\varphi_i(k_{(L)})$, так и листьями, а значения всех листьев заданы унитарными векторами, то итоговая степень полинома будет не больше 3 – в нем сохранятся только те слагаемые, которые не умножаются на нулевые значения компонент векторов листьев. Каждое слагаемое полинома будет иметь вид $m_j \prod_{d=1}^{\varphi_num} \varphi_d$, m_j – одна компонента кортежа в некоторой ячейке унитарно закодированной матрицы \tilde{M} . Уникальность каждого слагаемого также следует из сути описанной операции.

Учитывая, что в ячейках всех матриц должны стоять унитарные вектора, получаем, что каждая компонента вектора, определяемого операцией $\tilde{y}^T \tilde{M} \tilde{x}$, может принимать значения либо 0, либо 1. На основе схемы $\tilde{y}^T \tilde{M} \tilde{x}$ и заданных значений функции f , получаем уравнения:

$$\begin{pmatrix} y^0(q^i) \\ y^1(q^i) \end{pmatrix}^T \begin{bmatrix} \begin{pmatrix} m_{00}^0 \\ m_{00}^1 \end{pmatrix} & \begin{pmatrix} m_{01}^0 \\ m_{01}^1 \end{pmatrix} \\ \begin{pmatrix} m_{10}^0 \\ m_{10}^1 \end{pmatrix} & \begin{pmatrix} m_{11}^0 \\ m_{11}^1 \end{pmatrix} \end{bmatrix} \begin{pmatrix} x^0(q^i) \\ x^1(q^i) \end{pmatrix} = \begin{pmatrix} K^0(q^i) \\ K^1(q^i) \end{pmatrix}$$

То, что $\varphi_i(k_{(L)})$ реализует отдельно взятый пример q из набора Q_i , означает, что $\tilde{y}^T \tilde{M} \tilde{x} = \varphi_i(q)$. Так как, вектор $\varphi_i(q)$ является унитарным, то у вектора результата операции $\tilde{y}^T \tilde{M} \tilde{x}$ только одна компонента должна быть равна 1 – та же, что и у вектора $\varphi_i(q)$, все остальные должны быть равны 0. То есть $\varphi_i(q)^T \tilde{y}^T \tilde{M} \tilde{x} = 1$.

Поэтому, обозначив через $P(\lambda_i, q)$ полином, соответствующий компоненте вектора, определяемой $\varphi_i(q)$ и которая должна равняться 1, получаем всю совокупность пунктов данного утверждения. \square

Для каждой $\varphi_i(k_{(L)})$, на каждом шаге декомпозиции формируем из Q соответствующий ей набор Q_i , основанный на листьях, соответствующих λ_i .

Пример 7. Добавим к рассмотренному в примере 5 еще один обучающий пример $q_2 = ((0, 1, 0), 1)$ на основе того же разбиения $\lambda_1 = (k_{(1)}, k_{(2,3)})$; унитарное представление:

$\tilde{q}_2 = ((\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}), \begin{pmatrix} 0 \\ 1 \end{pmatrix})$. Тогда имеем следующий набор операций:

$$\left(\begin{pmatrix} \varphi_{2_{-00}}^0 \\ \varphi_{2_{-00}}^1 \end{pmatrix}^T \begin{bmatrix} \begin{pmatrix} m1_{00}^0 \\ m1_{00}^1 \end{pmatrix} & \begin{pmatrix} m1_{01}^0 \\ m1_{01}^1 \end{pmatrix} \\ \begin{pmatrix} m1_{10}^0 \\ m1_{10}^1 \end{pmatrix} & \begin{pmatrix} m1_{11}^0 \\ m1_{11}^1 \end{pmatrix} \end{bmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \left(\begin{pmatrix} \varphi_{2_{-00}}^0 \\ \varphi_{2_{-00}}^1 \end{pmatrix}^T \begin{bmatrix} \begin{pmatrix} m1_{00}^0 \\ m1_{00}^1 \end{pmatrix} & \begin{pmatrix} m1_{01}^0 \\ m1_{01}^1 \end{pmatrix} \\ \begin{pmatrix} m1_{10}^0 \\ m1_{10}^1 \end{pmatrix} & \begin{pmatrix} m1_{11}^0 \\ m1_{11}^1 \end{pmatrix} \end{bmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

с унитарными условиями: $\forall \{i, j\} \in \{0, 1\}^2 \quad m1_{ij}^0 + m1_{ij}^1 = 1, m1_{ij}^i \in \{0, 1\}, \varphi_{2_{-00}}^0 + \varphi_{2_{-00}}^1 = 1$, Из сопоставления уравнений составленных для первого и второго обучающих примеров, следует необходимость согласования значений $\varphi_{2_{-00}}$ и $\varphi_{2_{-01}}$. Поэтому, при составлении $\varphi_i(k_{(L_i)})$, реализующей одновременно q_1 и q_2 , требуем $\varphi_{2_{-00}}^T \varphi_{2_{-01}} = 0$. \square

Следствие 3. Для любых $L \subset \mathbb{N}, K \subset \mathbb{N}$ и любого возможного $Q_i \subset K^{l+1}$ с единой шкалой $\varphi_i(k_{(L_i)})$ реализует Q_i , когда $\sum_{q \in Q_i} P(\lambda_i, q) = |Q_i|$, для любого возможного q в единой шкале, с учетом ограничений согласования значений $\varphi_r(k_{(L_r)})$ и $\varphi_c(k_{(L_c)})$ для всех примеров из Q_i . \square

Доказательство следствия 3.

Как показано в доказательстве утверждения 5 $\varphi_i(k_{(L_i)})$ реализует некоторый обучающий пример $q \Leftrightarrow P(\lambda_i, q) = 1$. Следовательно, если реализуются все $|Q_i|$ примеров, то $\sum_{q \in Q_i} P(\lambda_i, q) = |Q_i|$. \square

Если к рассматриваемой матрице присоединяются две ветви, то $\varphi_{-num} = 2$, и следует рассмотреть согласованность функций $\varphi_r(k_{(L_r)})$ и $\varphi_c(k_{(L_c)})$ для каждой из ветвей. \square

Утверждение 6. Для любых $L \subset \mathbb{N}, K \subset \mathbb{N}$, $IRM_{G,2}(q)$, представленный как декомпозиция некоторой заданной в табличном виде функции f , реализует $Q \subset K^{l+1}$, если $\varphi_i(k_{(L_i)})$ для некоторой последовательности разбиений $\Lambda(G)$, соответствующей дереву G , реализует Q_i для $i \in \{1, \dots, l-1\}$. \square

Доказательство утверждения 6.

Для любых $L \subset \mathbb{N}, K \subset \mathbb{N}$, любой $Q \subset K^{l+1}$, в соответствии с некоторой последовательности разбиений $\Lambda(G)$ соответствующей дереву G из множества полных бинарных деревьев $\Gamma_2(L)$, $f(\tilde{l}_1, \dots, \tilde{l}_L)$ может быть представлена как суперпозиция функций меньшего числа переменных $\varphi_i(k_{(L_i)}), \forall i \in \{1, \dots, l-1\}$, определенных на наборах данных Q_i , полученных из Q , так как, все

компоненты $\varphi_i(k_{(L_i)})$: $\varphi_r(k_{(L_r)})$ и $\varphi_c(k_{(L_c)})$ формируется на основе $\varphi_i(k_{(L_i)})$, $\forall i \in \{1, \dots, l-1\}$ по построению, а значения $\varphi_1(k_{(L_1)})$ задаются через Q . \square

Следствие 4. Если $\sum_{q \in Q_i} P(\lambda_i, q) < |Q_i|$, $\forall i \in \{1, \dots, l-1\}$, то $IRM_{G,2}(Q) = \emptyset$. \square

Доказательство следствия 4.

Из утверждения 6 следует, что задача получения $f(\tilde{l}_1, \dots, \tilde{l}_{|L|})$ как суперпозиции функций меньшего числа переменных $\varphi_i(k_{(L_i)})$, $\forall i \in \{1, \dots, l-1\}$, может быть выражена последовательностью шагов $\max_{q \in Q_i} \sum P(\lambda_i, q)$, с нахождением $\varphi_i(k_{(L_i)})$, $\forall i \in \{1, \dots, l-1\}$. $\varphi_1(k_{(L_1)})$ имеем из Q , а $\varphi_r(k_{(L_r)})$ и $\varphi_c(k_{(L_c)})$, $\forall i \in \{2, \dots, l-1\}$ находим для каждой из декомпозируемых функций на основе $\varphi_i(k_{(L_i)})$. Соответственно, если задача не решается на каком-то шаге i , для некоторых из $|Q_i|$ примеров, значит, что она не решается и для компонент $\varphi_i(k_{(L_i)})$. Следовательно $IRM_{G,2}(Q) = \emptyset$. \square

Таким образом, если для матрицы верхнего уровня удастся найти решение оптимизационной задачи

$$(6) \text{ Arg max}_{m_r, \varphi_r, \varphi_c} \sum_{q \in Q_1} P(\lambda_1, q),$$

такое что $\sum_{q \in Q_1} P(\lambda_1, q) = |Q_1|$, можно продолжить декомпозицию на следующий узел структуры, в качестве функции теперь используя найденные значения компонент $\varphi_1(k_{(L_1)})$. При этом $\varphi_1(k_{(L_1)})$ и найденные значения M_I можно использовать для декомпозиции на поддеревьях $\{L_{1r}; L_{1c}\} \subset L(G)$ на значениях индикаторов $k_{(L_{1r})}$, $k_{(L_{1c})}$ соответственно. Применяя данный подход, последовательно для $L(G)$ можно найти $\varphi_i(k_{(L_i)})$ и M_i , $\forall i \in \{1, \dots, l-1\}$.

Пример 8. Рассмотрим обучающий набор данных, представленный в таблице 15.

Таблица 15 – Исходные данные задачи декомпозиции.

q	l_1	l_2	l_3	l_4	f
1	0	0	0	0	0
2	0	1	0	0	1
3	1	1	0	0	0
4	0	0	1	0	1
5	1	1	1	0	1
6	0	1	0	1	1

Продолжение таблицы 15

q	l_1	l_2	l_3	l_4	f
7	1	0	1	1	0
8	1	1	1	1	0

Исследуем, в первую очередь, реализуемость функции $f = \varphi_1(\tilde{l}_1, \varphi_2(\tilde{l}_2, \tilde{l}_3, \tilde{l}_4))$, и в случае если $\varphi_1(k_{(L)})$ доступна в заданной шкале k_L , то перейдем к поиску реализации $\varphi_2(k_{(L)})$. Положим разбиение первого уровня: $\lambda_1 = (k_{(1)}, k_{(2,3,4)})$. Составим уравнения для первого примера первой ступени:

$$\left(\begin{array}{c} \left(\varphi_{2_000}^0 \right)^T \\ \left(\varphi_{2_000}^1 \right) \end{array} \right)^T \left[\begin{array}{cc} \left(\begin{array}{c} m_{00}^0 \\ m_{00}^1 \end{array} \right) & \left(\begin{array}{c} m_{01}^0 \\ m_{01}^1 \end{array} \right) \\ \left(\begin{array}{c} m_{10}^0 \\ m_{10}^1 \end{array} \right) & \left(\begin{array}{c} m_{11}^0 \\ m_{11}^1 \end{array} \right) \end{array} \right] \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{с унитарными условиями: } \forall \{i, j\} \in \{0, 1\}^2$$

$m_{ij}^0 + m_{ij}^1 = 1$, $m_{ij}^t \in \{0, 1\}$, $\varphi_{2_000}^0 + \varphi_{2_000}^1 = 1$, $\varphi_{2_000}^t \in \{0, 1\}$, $\forall t \in \{0, 1\}$. Имеем

$$\varphi_{2_000}^0 m_{00}^0 + \varphi_{2_000}^1 m_{10}^0 = 1,$$

$$\varphi_{2_000}^0 m_{00}^1 + \varphi_{2_000}^1 m_{10}^1 = 0.$$

Далее, по схеме, описанной в примере 6, получаем набор уравнений для всех примеров:

$$\varphi_{2_000}^0 m_{00}^0 + \varphi_{2_000}^1 m_{10}^0 = 1, \quad \varphi_{2_100}^0 m_{00}^1 + \varphi_{2_100}^1 m_{10}^1 = 1, \quad \varphi_{2_100}^0 m_{01}^0 + \varphi_{2_100}^1 m_{11}^0 = 1,$$

$$\varphi_{2_010}^0 m_{00}^1 + \varphi_{2_010}^1 m_{10}^1 = 1, \quad \varphi_{2_110}^0 m_{10}^1 + \varphi_{2_110}^1 m_{11}^1 = 1, \quad \varphi_{2_101}^0 m_{00}^1 + \varphi_{2_101}^1 m_{10}^1 = 1,$$

$$\varphi_{2_011}^0 m_{01}^0 + \varphi_{2_011}^1 m_{11}^0 = 1, \quad \varphi_{2_111}^0 m_{01}^0 + \varphi_{2_111}^1 m_{11}^0 = 1,$$

из которых составляем оптимизационную задачу:

$$\varphi_{2_000}^0 m_{00}^0 + \varphi_{2_000}^1 m_{10}^0 + \varphi_{2_100}^0 m_{00}^1 + \varphi_{2_100}^1 m_{10}^1 + \varphi_{2_100}^0 m_{01}^0 + \varphi_{2_100}^1 m_{11}^0 + \varphi_{2_010}^0 m_{00}^1 + \varphi_{2_010}^1 m_{10}^1 + \varphi_{2_110}^0 m_{10}^1 + \varphi_{2_110}^1 m_{11}^1 + \varphi_{2_101}^0 m_{00}^1 + \varphi_{2_101}^1 m_{10}^1 + \varphi_{2_011}^0 m_{01}^0 + \varphi_{2_011}^1 m_{11}^0 + \varphi_{2_111}^0 m_{01}^0 + \varphi_{2_111}^1 m_{11}^0 \xrightarrow{M, \varphi_2} \max.$$

Также добавляются ограничения, выписанные на основе анализа конфликтов между уравнениями, составленными для разных ступеней. Например, из

$$\left(\begin{array}{c} \left(\varphi_{2_000}^0 \right)^T \\ \left(\varphi_{2_000}^1 \right) \end{array} \right)^T \left[\begin{array}{cc} \left(\begin{array}{c} m_{00}^0 \\ m_{00}^1 \end{array} \right) & \left(\begin{array}{c} m_{01}^0 \\ m_{01}^1 \end{array} \right) \\ \left(\begin{array}{c} m_{10}^0 \\ m_{10}^1 \end{array} \right) & \left(\begin{array}{c} m_{11}^0 \\ m_{11}^1 \end{array} \right) \end{array} \right] \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{для первого примера и}$$

$$\left(\begin{array}{c} \left(\varphi_{2_100}^0 \right)^T \\ \left(\varphi_{2_100}^1 \right)^T \end{array} \right)^T \left[\begin{array}{cc} \left(m1_{00}^0 \right) & \left(m1_{01}^0 \right) \\ \left(m1_{00}^1 \right) & \left(m1_{01}^1 \right) \\ \left(m1_{10}^0 \right) & \left(m1_{11}^0 \right) \\ \left(m1_{10}^1 \right) & \left(m1_{11}^1 \right) \end{array} \right] \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \text{ для второго выпишем } \left(\varphi_{2_000}^0 \right)^T \left(\varphi_{2_100}^0 \right) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \text{ или}$$

$$\varphi_{2_000} \varphi_{2_100} = 0; \varphi_{2_000} \varphi_{2_010} = 0; \varphi_{2_110} \varphi_{2_100} = 0; \varphi_{2_000} \varphi_{2_101} = 0; \varphi_{2_110} \varphi_{2_011} = 0; \varphi_{2_110} \varphi_{2_111} = 0.$$

$$\text{Решение этой системы: матрица } \tilde{M}_1 = \begin{bmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \begin{pmatrix} 0 \\ 1 \end{pmatrix} & \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{bmatrix} \text{ и вектор } \tilde{\varphi}_2 = \left(\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \right.$$

$\left. \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right)$. На основе полученных значений $\varphi_2(k_{(L_2)})$ составляем таблицу 16, данных для второй

ступени декомпозиции $f = \varphi_2(\tilde{l}_3, \varphi_3(\tilde{l}_1, \tilde{l}_2))$, в столбец \tilde{k}_L помещаем значения $\varphi_2(k_{(L_2)})$:

Таблица 16 – Данные на второй степени декомпозиции для задачи декомпозиции.

q	l_2	l_3	l_4	φ_2
1	0	0	0	0
2	1	0	0	1
3	0	1	0	1
4	1	1	0	0
5	1	0	1	1
6	0	1	1	1
7	1	1	1	1

Используем разбиение второго шага в виде $\lambda_2 = (k_{(4)}, k_{(2,3)})$. Составим для него уравнение:

$$\left(\begin{array}{c} \left(\varphi_{3_00}^0 \right)^T \\ \left(\varphi_{3_00}^1 \right)^T \end{array} \right)^T \left[\begin{array}{cc} \left(m2_{00}^0 \right) & \left(m2_{01}^0 \right) \\ \left(m2_{00}^1 \right) & \left(m2_{01}^1 \right) \\ \left(m2_{10}^0 \right) & \left(m2_{11}^0 \right) \\ \left(m2_{10}^1 \right) & \left(m2_{11}^1 \right) \end{array} \right] \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

с унитарными условиями: $\forall \{i, j\} \in \{0,1\}^2 \quad m2_{ij}^0 + m2_{ij}^1 = 1, \quad m2_{ij}^i \in \{0,1\}, \quad \varphi_{3_00}^0 + \varphi_{3_00}^1 = 1, \quad \varphi_{3_00}^i \in \{0,1\}, \quad \forall t \in \{0,1\}.$

$$\varphi_{3_00}^0 m2_{00}^0 + \varphi_{3_00}^1 m2_{10}^0 = 1, \quad \varphi_{3_00}^0 b_{00}^1 + \varphi_{3_00}^1 b_{10}^1 = 0.$$

Далее, получаем набор уравнений:

$$\begin{aligned} \varphi_{3_{-00}}^0 m2_{00}^0 + \varphi_{3_{-00}}^1 m2_{10}^0 &= 1, & \varphi_{3_{-10}}^0 m2_{00}^1 + \varphi_{3_{-10}}^1 m2_{10}^1 &= 1, & \varphi_{3_{-01}}^0 m2_{00}^1 + \varphi_{3_{-01}}^1 m2_{10}^1 &= 1, \\ \varphi_{3_{-11}}^0 m2_{00}^0 + \varphi_{3_{-11}}^1 m2_{10}^0 &= 1, & \varphi_{3_{-10}}^0 m2_{10}^1 + \varphi_{3_{-10}}^1 m2_{11}^1 &= 1, & \varphi_{3_{-01}}^0 m2_{10}^1 + \varphi_{3_{-01}}^1 m2_{11}^1 &= 1, \\ \varphi_{3_{-11}}^0 m2_{01}^1 + \varphi_{3_{-11}}^1 m2_{11}^1 &= 1. \end{aligned}$$

Из них составляем оптимизационную задачу:

$$\begin{aligned} &\varphi_{3_{-00}}^0 m2_{00}^0 + \varphi_{3_{-00}}^1 m2_{10}^0 + \varphi_{3_{-10}}^0 m2_{00}^1 + \varphi_{3_{-10}}^1 m2_{10}^1 + \varphi_{3_{-01}}^0 m2_{00}^1 + \varphi_{3_{-01}}^1 m2_{10}^1 + \varphi_{3_{-11}}^0 m2_{00}^0 + \varphi_{3_{-11}}^1 m2_{10}^0 + \varphi_{3_{-10}}^0 m2_{10}^1 + \\ &+ \varphi_{3_{-10}}^1 m2_{11}^1 + \varphi_{3_{-01}}^0 m2_{10}^1 + \varphi_{3_{-01}}^1 m2_{11}^1 + \varphi_{3_{-11}}^0 m2_{01}^1 + \varphi_{3_{-11}}^1 m2_{11}^1 \xrightarrow{M_2, \varphi_3} \max. \end{aligned}$$

Добавим ограничения, выписанные на основе анализа конфликтов между уравнениями, составленными для разных примеров:

$$\varphi_{3_{-10}} \varphi_{3_{-00}} = 0; \quad \varphi_{3_{-01}} \varphi_{3_{-00}} = 0; \quad \varphi_{3_{-00}} = \varphi_{3_{-11}}.$$

$$\text{Решение: } \tilde{M}_2 = \begin{bmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \begin{pmatrix} 0 \\ 1 \end{pmatrix} & \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{bmatrix} \text{ и вектор } \tilde{\varphi}_3 = \left(\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right).$$

Очевидно, что φ_3 в дальнейшей декомпозиции не нуждается, из него получаем

$$\tilde{M}_3 = \begin{bmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \begin{pmatrix} 0 \\ 1 \end{pmatrix} & \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{bmatrix}. \square$$

Рассмотрение некоторой $G \in \Gamma_2(L)$, состоящей из допущенных подветвей, производится на основе таблицы ветвей следующим образом. Из столбца таблицы с наибольшим номером последовательно берутся λ_i . На основе Q и каждого λ_i составляется оптимизационная задача

$$(7) \text{ Arg max}_{m_{rc}, \varphi_r, \varphi_c} \sum_{q \in Q_i} P(\lambda_i, q);$$

в случае если удастся найти решение в рамках допустимой шкалы, M_i сохраняется, а полученные значения $\varphi_r(k_{(L_r)})$ и $\varphi_c(k_{(L_c)})$ используются для поиска решения для ветвей, на которые разбиваются рассмотренные λ_i . После рассмотрения всех λ_i в столбце с наибольшим номером рассмотрение переходит в столбец с меньшим номером.

Предложенный подход обладает тем преимуществом, что в случае, когда не удалось найти решение $\text{Arg max}_{m_{rc}, \varphi_r, \varphi_c} \sum_{q \in Q_i} P(\lambda_i, q)$, можно исключить из дальнейшего рассмотрения целое семейство поддеревьев, порождаемых из λ_i , и, как следствие, всех $G \in \Gamma_2(L)$, содержащих такие поддеревья. Например, если нет решения в шкале k_L для $\lambda_1 = (k_{(1)}, k_{(2.3.4)})$, см. табл. 9, то значит, что

из рассмотрения можно удалить структуры $M_{1l_1}M_{2l_4}M_{3l_2}l_3$ и $M_{1l_1}M_{2l_2}M_{3l_3}l_4$, см. [95], т.е. не рассматривать декомпозиции $\lambda_2 = (k_{(3)}, k_{(1,2)})$ и $\lambda_2 = (k_{(1)}, k_{(2,3)})$. Однако может оказаться и так, что у рассматриваемой задачи (7) существует сразу несколько решений $\varphi_r(k_{(L_r)})$ и $\varphi_c(k_{(L_c)})$ с соответствующими M_i , тогда возникает задача отбора из них более перспективных, для использования на следующих уровнях декомпозиции.

3.3.2. Численное решение задачи

Так как в каждом из рассматриваемых узлов $G \in \Gamma_2(L)$ степень полинома не превышает 3, то сложность составляемых оптимизационных задач невелика и удается их быстро решать средствами оптимизатора. Например, Gurobi 9.5.0 решает оптимизационную задачу (8 квадратичных ограничений и 16 общих ограничений) первой ступени второго примера за 30 мсек на AMD Ryzen 7 4800H, 16 гб оперативной памяти. Как видно из формулы $P(\lambda_i, q) = \sum_{j=1}^{k^{\varphi_num}} p_j$ число мономов оптимизационного полинома, а значит и сложность задачи оптимизации, в предлагаемом подходе не зависит от числа входных параметров, что позволяет работать с наборами дискретных данных с большим числом параметров.

3.4. Анализ матриц МКО

В случае, если для некоторого обучающего набора данных задача идентификации МКО, его реализующего, была решена, то актуальной становятся вопросы анализа их свойств. Например, нередкой является ситуация, когда в матрицах синтезированного механизма присутствуют неиспользуемые ячейки. В результате решения задачи оптимизации для синтеза МКО инициализируются все ячейки матриц синтезируемого МКО. Хотя может быть так, что какие-то из ячеек реально не используются в процессе получения комплексной оценки на базе синтезированного МКО, т.е. их значения не запрашиваются алгоритмом оценивания. Поэтому был предложен алгоритм проверки ячеек матриц МКО на предмет используемости.

Имеем: набор заполненных матриц МКО $list_M$, запись структуры МКО $note$, таблицу обучающих примеров Q , единую шкалу оценок K , количество обучающих примеров $leaves_combinations$.

Получаем: набор матриц МКО, в котором неиспользуемые ячейки матриц обозначены n .

Создаем набор матриц $list_M_filled$, в которые будем заносить значения используемых ячеек. Значения матриц инициализируем NA (в тексте диссертации для краткости используется x).

1. Для каждого из i обучающих примеров $leaves_combinations$:
 - a. Создаем список $list_l$ содержащий унитарные вектора параметров рассматриваемого примера i .
 - b. В соответствии с заданной структурой, вычисляем значение комплексной оценки, для рассматриваемого примера.
 - c. Пока в $note$ есть элементы:
 - i. Находим номер последней матрицы содержащейся в $note$, берем соответствующую матрицу $M_{current}$ из $list_M$. Запоминаем номера относящихся к ней листьев из $list_l$.
 - ii. Производим операцию умножения матрицы $M_{current}$ на относящиеся к ней листья из $list_l$. Удаляем использованные в операции умножения листья. Сохраняем результат в качестве последнего листа в $list_l$.
 - iii. Удаляем из $note$ использованные элементы.
 - iv. Запоминаем в $list_M_filled$ ту ячейку, на которую указывают значения листьев из $list_l$.
2. Возвращаем $list_M_filled$, представляющий собой набор матриц МКО, в которых неиспользуемые ячейки отмечены как x .

Например, рассмотрим результат синтеза МКО для данных, представленных в таблице 7, на основе структуры $M_1M_2l_1l_2M_3l_3l_4$. Построенный МКО реализует все восемь обучающих примеров предоставленного обучающего набора данных в шкале два. Значения матриц МКО, полученные от алгоритма синтеза представлены в таблице 17.

Таблица 17 – Решение задачи синтеза для данных из таблицы 7 на основе структуры $M_1M_2l_1l_2M_3l_3l_4$.

M_1	M_2	M_3
0 1	0 1	0 0
1 1	0 0	1 0

После поиска неиспользуемых ячеек, для данных матриц, получаем результаты, представленные в таблице 18.

Таблица 18 – Решение задачи синтеза для данных из таблицы 7 на основе структуры $M_1M_2l_1l_2M_3l_3l_4$, после проверки используемости ячеек.

M_1	M_2	M_3
0 1	0 1	0 0
1 x	0 0	1 0

Символ x обозначает неиспользуемую, для данного обучающего набора ячейку. Это значит, что данная ячейка матрицы не принимает участия в формировании комплексной оценки ни для одного из примеров обучающего набора данных. Более подробно результаты синтеза МКО для набора данных из таблицы 7 будут рассмотрены в главе 4. Такие неиспользуемые ячейки можно интерпретировать как степени свободы МКО в том смысле, что наличие незаполненной ячейки означает возможность расширения количества обучающих примеров, которые могут быть реализованы с помощью рассматриваемого МКО. Так, набор входных примеров из таблицы 7 можно расширить примером («выше», «в гостях», «пропускают», «да», «да») или («выше», «в гостях», «пропускают», «да», «нет»).

В п. 1.6 первой главы была введена классификация матриц согласно [13]. С этой точки зрения, можно рассмотреть синтезированные матрицы. Наличие неиспользуемых ячеек дает свободу в выборе характера матриц. Так, поставив «1» в ячейку с x , мы получим матрицу поощрения

$$M_1 = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}.$$

То есть из результатов свертки матриц M_2 и M_3 всегда выбирается большая оценка. А поставив «0», получим матрицу, соответствующую функции XOR , т.е. M_1 будет возвращать «1» только когда либо оценка M_2 , либо оценка M_3 равна «1»:

$$M_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

3.5. Выводы по главе 3

В главе представлены подходы к решению задачи синтеза МКО на основе унитарного кодирования. Алгоритм синтеза всех матриц заданной структуры позволяет синтезировать сразу все матрицы для рассматриваемой структуры МКО, однако степень составляемого в процессе синтеза, оптимизационного полинома растет линейно в зависимости от количества входных параметров, а число общих ограничений оптимизационной задачи растет экспоненциально. Рассмотрен подход к синтезу МКО с применением разделительной декомпозиции, позволяющий существенно снизить вычислительную сложность составляемых оптимизационных задач путем декомпозиции исходной задачи на более простые. В качестве схемы декомпозиции предлагается использовать таблицу ветвей, введенную во второй главе. Основа уменьшения сложности оптимизационной задачи - избавление от зависимости от количества входных параметров для каждого отдельного узла МКО. Вторым плюсом предлагаемого подхода состоит в том, что в случае отсутствия решений на некотором шаге декомпозиции $\varphi_i(k_{(L)})$ в заданной шкале K_L нет

необходимости рассматривать продолжение декомпозиции, в виду отсутствия решения задачи в заданной шкале. Развитие данного метода возможно в сторону отбора более перспективных решений, на каждом шаге декомпозиции. Приведен пример анализа матриц синтезированного МКО, с учетом незаполненных ячеек некоторой матрицы МКО. Также необходимо упомянуть, что иллюстрирующие примеры в настоящей главе были основаны на бинарной логике, но, как будет показано далее в главе 4, переход на многозначную логику не вызывает дополнительных трудностей при реализации предложенного подхода.

Глава 4. Прикладные задачи анализа и синтеза МКО

После рассмотрения вопросов, связанных с синтезом и отбором перспективных структур полных бинарных деревьев, рассмотренных во второй главе, а также, после рассмотрения методов синтеза и анализа матриц МКО, перейдем к описанию программной части реализующей предложенные методы, и применению реализованного программного кода к прикладным задачам. В разделах 4.1 – 4.2 данной главы приводится общий алгоритм работы реализованного в рамках данной работы программного комплекса по анализу и синтезу МКО, а также описание работы его модулей. В разделе 4.3 демонстрируется работа подхода по синтезу всех матриц заданной структуры МКО на примерах синтеза МКО для булевых функций. На примере решения прикладных задач синтеза МКО в разделе 4.4 демонстрируются подход к анализу и синтезу МКО. Раздел 4.5 посвящен вопросу синтеза прогнозной системы на основе результатов предсказания успешно обученных моделей МКО.

В разделе 1.3 были приведены примеры применения МКО в различных областях хозяйственной деятельности. В данной главе, в качестве примеров рассматриваются как демонстрационные примеры, так и характерные задачи синтеза МКО, такие как: оценка успешности компаний и оценка уровня эффективности обеспечения безопасности полетов. Разработанный программный комплекс реализует общий алгоритм синтеза и анализа МКО и представляет собой набор модулей, каждый из которых реализует функциональность описанных в главах 2 и 3 методов. Программный код написан на языках программирования *R* и *Python*. Для создания веб-версии модуля анализа групп эквивалентности была использована библиотека с открытым исходным кодом *voila*.

4.1. Модули программного комплекса по синтезу и анализу МКО

4.1.1. Модуль потоковой генерации записей структур

В данном подразделе описывается модуль, работающий на основе метода потоковой генерации структур. Данный модуль работает в соответствии с алгоритмами изложенными в разделе 2.6. Потоковая генерация позволяет формировать запись структуры полного бинарного дерева для модулей синтеза или анализа групп эквивалентности без формирования всего списка возможных структур, который может быть довольно обширным, учитывая, что число структур полных бинарных деревьев на именованных листьях растет как двойной факториал от числа параметров.

Наравне с потоковой генерацией данных, модуль может быть использован для формирования списка структур полных бинарных деревьев.

На вход модуль получает число индикаторов l и мл-запись начальной структуры *note*. Метод генерации состоит из двух этапов. На первом, осуществляется изменение структуры полного бинарного дерева. Для этого в пространстве всех допустимых структур полных бинарных деревьев производится последовательный переход к следующей структуре. На втором этапе происходит работа с результатами первого этапа. На основе каждой из синтезированных структур первого уровня производится генерация всех допустимых структур на именованных листьях, согласно правилам, изложенным в разделе 2.1. Для этого, последовательно, в рамках пространства допустимых структур полных бинарных деревьев на именованных листьях осуществляются перестановки для пар листьев, как изложено в пункте Б раздела 2.6. Чтобы избежать повторов в результатах к результатам описанного алгоритма первого уровня применяется проверка на симметричность и зеркальность подветвей. Предложенный метод опробован для генерации структур МКО с $|L| < 11$. В задачах, с которыми нам приходилось сталкиваться в процессе написания данной работы количество индикаторов не превышало данное ограничение.

4.1.2. Модуль анализа групп эквивалентности

Как было сказано в разделе 2.3, получение групп эквивалентности позволяет удалить из рассмотрения все структуры $G \in \Gamma_2(L)$, которые включают в себя комбинации с количеством групп эквивалентности, превышающим шкалу $|K_{\tilde{L}}(Q)| \geq K_L$. Это позволяет сократить количество структур, которые необходимо учитывать при любом методе синтеза МКО для конкретных $G \in \Gamma_2(L)$. Во-вторых, основываясь на группах эквивалентности, удовлетворяющих шкале $|K_{\tilde{L}}(Q)| \leq K_L$, мы можем использовать алгоритм отбора заведомо реализуемых структур для случая полностью определенной функции. При работе в «потоковом режиме», см. подраздел 4.1.1, модуль анализа применяется последовательно к каждой из поданных на вход структур. Для этого на основе рассматриваемой структуры составляется список разбиений групп листьев для проверки и модуль работает в первом режиме. В первом режиме производится анализ групп эквивалентности для всех возможных подгрупп листьев.

Порядок работы модуля.

Имеем: таблицу дискретных данных на l индикаторах (*table_data*), мл-запись структуры (опционально).

Получаем: таблицу со значениями групп эквивалентности для каждой из подгрупп входных параметров.

1. Производится проверка соответствия входных данных установленному формату представления. Принимается таблица с заголовками столбцов, без нумерации строк, в ячейках числовые значения. На основе введенных данных получаем количество параметров и шкалу представления данных. Создается таблица для представления результатов с пустыми ячейками.
2. Выбирается режим: анализ конкретного разбиения параметров или всех возможных разбиений для данного набора параметров входного набора данных.
3. Если выбран второй режим, формируется список разбиений групп листьев для рассмотрения.
4. Для каждого разбиения из списка разбиений или одного выбранного для рассмотрения разбиения:
 - а. Создается Карта Карно на основе введенной таблицы данных
 - б. Используя метод из раздела 2.4, вычисляется количество групп эквивалентности
5. Возвращение таблицы с результатами анализа.

Если был выбран первый режим, то полученную таблицу результатов анализа групп эквивалентности для всех возможных подгрупп листьев можно использовать для синтеза структур полных бинарных деревьев, см. алгоритм 2 из раздела 2.6, или синтеза таблицы ветвей, см. раздел 2.5. Синтезированный таким образом список можно подать на вход модуля синтеза матриц МКО из следующего раздела. На рисунке 10 представлен снимок экрана с разработанным веб-интерфейсом для блока анализа групп эквивалентности. В верхней части экрана представлены загруженные данные, а в нижней карта Карно с результатом анализа групп эквивалентности, построенная для заданного разбиения листьев (здесь, a соответствует листу l_1 , а b соответствует l_2).

voila

Cournot map builder

Data preloaded:

	11	12	13	14	15	IR
0	2	1	1	2	0	2
1	1	0	0	1	0	1
2	1	1	2	0	2	2
3	2	1	2	2	1	2
4	1	1	0	1	1	2

Upload your data and choose group for splitting, from the data column names

Upload (0) Upload

column split: ab

Get Cournot map

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	grp
0		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16		
1		000	100	010	110	120	101	201	011	111	211	021	121	221	202	022	222		
2	1	00	nan	0.0	1.0	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	2
3	2	10	1.0	nan	1.0	1.0	1.0	nan	nan	1.0	nan	nan	nan	nan	nan	nan	nan	nan	2
4	3	20	1.0	nan	nan	nan	nan	nan	nan	nan	2.0	nan	2.0	nan	nan	nan	nan	nan	0
5	4	01	nan	nan	2.0	nan	nan	nan	nan	nan	2.0	nan	nan	1.0	nan	nan	nan	nan	0
6	5	11	nan	nan	nan	nan	nan	1.0	1.0	2.0	2.0	2.0	nan	1.0	nan	2.0	2.0	nan	0
7	6	21	nan	1.0	nan	nan	2.0	nan	nan	nan	2.0	nan	nan	nan	2.0	nan	nan	nan	0
8	7	12	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	1.0	nan	nan	nan	nan	0
9	8	22	nan	nan	nan	2.0	nan	nan	nan	nan	nan	2.0	nan	2.0	2.0	nan	nan	2.0	1

'dim adr = '
3
'dim grp = '
3

file name: Cournout Enter

Рисунок 10 – веб-интерфейс модуля анализа групп эквивалентности.

4.1.3. Модуль синтеза матриц МКО

Заполненность набора входных данных примерами является одним из ключевых критериев выбора метода синтеза матриц МКО. В зависимости от того подан на вход полный или неполный набор, будет произведен выбор соответствующего метода. В основе синтеза матриц МКО на полном наборе данных лежит использование метода анализа групп эквивалентности, изложенного в разделе 2.4. Если результаты анализа групп эквивалентности показывают реализуемость подгрупп листьев необходимых для построения рассматриваемой структуры МКО, то на основе соответствующих карт Карно можно автоматически получить искомые матрицы МКО, алгоритм синтеза приведен в разделе 3.1.

Для случая неполного набора данных, в разделах 3.2 и 3.3 были предложены два метода синтеза матриц МКО. В результате выполнения алгоритма синтеза всех матриц МКО возвращается

полный список синтезированных матриц для рассматриваемой структуры *note*. Список структур, подаваемый на вход, поступает из модуля 4.1.2 анализа групп эквивалентности. В случае, когда количество индикаторов входного набора данных превышает семь, предполагается использование метода декомпозиции из раздела 3.3. Тогда задача синтеза всех матриц для рассматриваемой структуры разбивается на несколько менее крупных задач синтеза. В такой постановке рекомендуется подход с использованием таблицы ветвей, позволяющий, с одной стороны, убирать из рассмотрения целые множества структур если синтез корневой для них матрицы оказался невозможен, с другой, использовать реализующиеся корневые матрицы для множества структур синтезируемых на их основе. Порог в семь параметров был выбран исходя из наших вычислительных возможностей, см. подраздел 3.3.2. Задача синтеза МКО хорошо распараллеливается, например, можно запускать синтез МКО на разных потоках выполнения в рамках одного компьютера или на разных компьютерах, на каждом свою часть множества рассматриваемых структур МКО.

4.1.4. Модуль комплексного оценивания

Модуль используется для вычисления комплексной оценки на основе введенных матриц и записи структуры полного бинарного дерева. В рамках предложенного программного комплекса используется для проверки синтезированных МКО на корректность возвращаемых оценок. Как и во всей работе, в данном модуле используются матрицы одинаковой размерности.

Имеем: запись структуры полного бинарного дерева на именованных листьях (*note*), в мл-нотации, значения матриц, вектор значений индикаторов (*list_of_indicators*).

Получаем: значение комплексной оценки.

1. Формируем список листьев на основе числа полученных матриц.
2. Используя инструмент унитарного кодирования, как изложено в разделе 2.1 производим серию последовательных умножений матриц и индикаторов в соответствии с *note*.
3. **Пока** в *note* больше одного элемента:
 - a. Производим матричное умножение для последней матрицы из *note* и соответствующих ей индикаторов.
 - b. Удаляем использованные в предыдущем пункте элементы из *note*, *list_of_indicators*.
 - c. Записываем результат в качестве последнего индикатора в *list_of_indicators*.
 - d. Вывод оставшегося значения в *list_of_indicators*.

Данный модуль также может быть использован в формате проверки корректности возвращаемых оценок синтезированным МКО по всем примерам заданного набора данных. В этом случае

вместо значения конкретного примера, для которого требуется вычислить КО, на вход модуля подаются все примеры исходного дискретного набора данных, вместе с оценками. В таком случае в качестве результата возвращается список оценок по каждому из примеров заданного набора данных.

4.2. Общий алгоритм анализа и синтеза МКО

В данном разделе раскрывается применение описанных модулей для решения задачи синтеза и анализа МКО в рамках единого программного комплекса. Ключевым моментом в прикладном применении является полнота данных. Как было показано в разделе 3.1 третьей главы, задачи в которых имеется полный набор данных могут быть обработаны на основе процедуры анализа групп эквивалентности. Для ситуации с неполными наборами данных, следует использовать методы, предложенные в разделах 3.2, 3.3 третьей главы. Порядок работы на основе метода синтеза, изложенный в разделе 3.3, с точки зрения программного комплекса, отличается тем, что на основе результатов работы блока анализа групп эквивалентности составляется таблица ветвей, а не список структур. Общий алгоритм работы программного комплекса по синтезу МКО изложен ниже. В разделе 4.3 рассмотрим применение комплекса программ на конкретных примерах.

Общий алгоритм синтеза МКО.

Имеем: таблицу дискретных данных на l индикаторах (*table_data*), список структур полных бинарных деревьев на именованных листьях (*list_of_structures*), в мл-нотации, как описано в разделе 2.1, шкалу (*scale*).

Получаем: значения для $l-1$ матрицы рассматриваемой структуры.

1. Производится проверка набора входных данных на корректность ввода. Значения приводятся в единую шкалу. Убираются повторяющиеся и противоречивые примеры.
2. **Если** набор входных данных неполный:
 - а. **Если** выбрана потоковая генерация структур:
 - i. Чтобы избежать длительного ожидания формирования полного списка структур, используется потоковая генерация структур из раздела 4.1.1.
 - б. **Иначе:**
 - i. **Если** список структур *list_of_structures* не задан:
 1. Формируем список структур *list_of_structures* с помощью методов блока 4.1.2.
 - ii. **Иначе:**

1. Модуль анализа групп эквивалентности, из раздела 4.1.2, используется для отсева нереализующихся структур. В результате, в списке *list_of_structures* остаются только допустимые структуры. Сохраняем таблицу ветвей.
2. **Пока** в списке *list_of_structures* есть структуры:
 - а. Используем модуль синтеза матриц для заданной структуры из раздела 4.1.3. Для нереализуемых в рамках заданной шкалы структур, возвращаются матрицы, аппроксимирующие заданный входной набор данных. Если $l > 7$, используем метод декомпозиции из раздела 3.3 и составленную таблицу ветвей для представления задачи синтеза как композиции задач с числом листьев не превышающем шесть. Ограничение в шесть листьев вызвано ростом вычислительной сложности задачи синтеза от числа листьев, как показано в разделе 3.3. Для каждой из подзадач используем алгоритм синтеза всех матриц, описанный в разделе 3.2.
 - iii. Синтезированные матрицы МКО анализируются на предмет пустых ячеек.
3. **Если** набор входных данных полный:
 - а. На основе алгоритма из раздела 3.1, используется модуль синтеза матриц МКО для полного набора данных, описанный в подразделе 4.1.3. Возвращаем результат или сообщение о не реализуемости заданного набора в рамках заданной шкалы. Аналогично, ветви алгоритма для неполного набора данных, на полном наборе, возможен запуск как по списку структур, так и в потоковом режиме, опускаем, чтобы не слишком усложнять схему алгоритма.

Полученные результаты выгружаются в файлы формата txt или xlsx для дальнейшего рассмотрения и анализа. Примеры анализа синтезированных МКО приведены в разделах 4.3 и 4.4. Может оказаться так, что данные не представимы матрицами логической свертки в рамках шкалы исходного набора данных $|K_L|$. Тогда, можно рассмотреть механизмы с большими размерностями матриц свертки. Блок-схема описанного алгоритма представлена на рисунке 11.

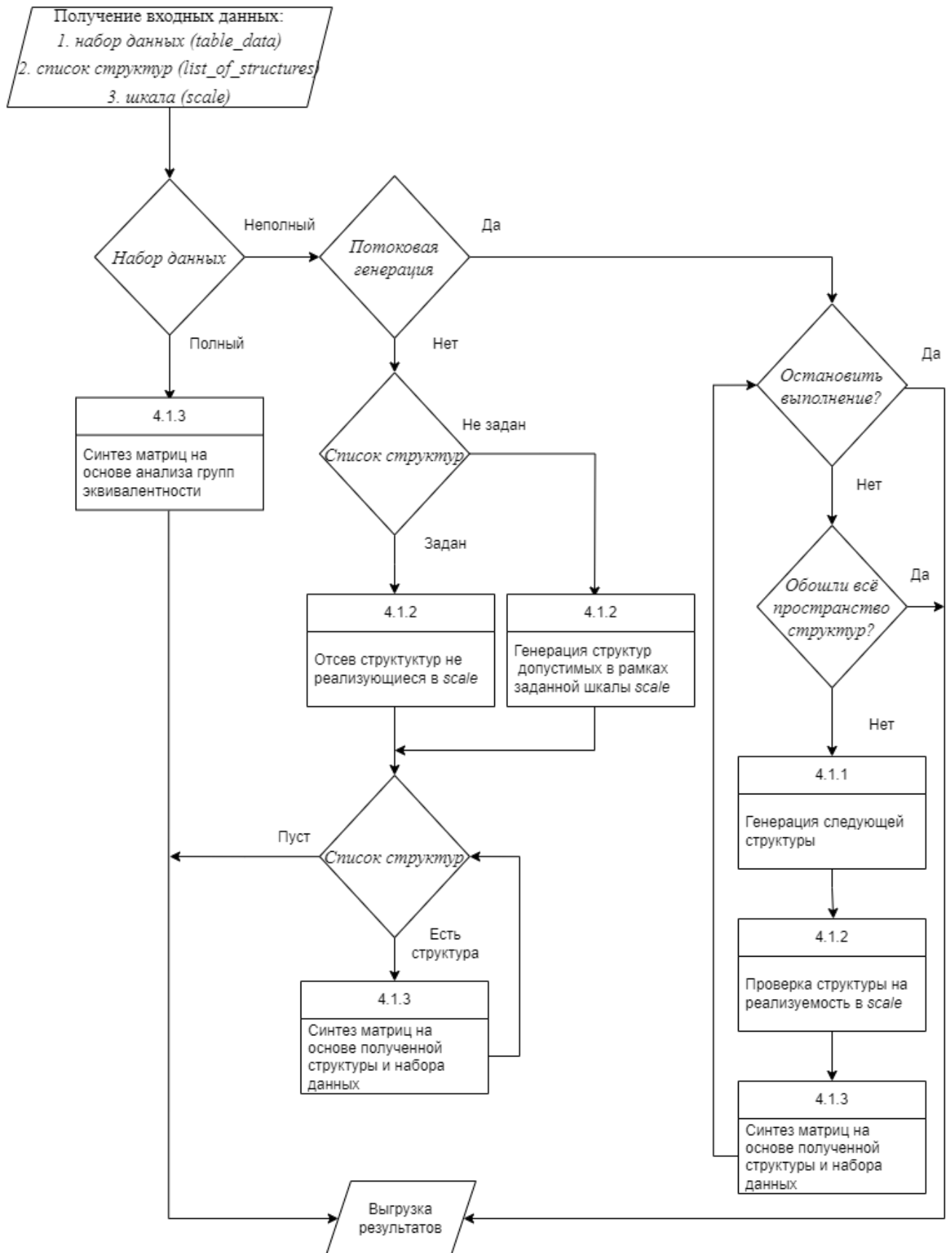


Рисунок 11 – Блок-схема общего алгоритма анализа и синтеза МКО.

4.3. Примеры анализа и синтеза для полного набора данных – булевы функции

Пример 9. Булева функция $f(x_1, x_2, x_3) = x_1 \wedge (x_2 \vee x_3)$.

Имеем $|L| = 3$ и $|K| = 2$, полный набор обучающих примеров представлен в таблице 19.

Решение задач (4), (5) для этого набора достигают два МКО со значением (4), равным 8 – обе реализуют этот набор (см., например, [79]).

Таблица 19 – Обучающий набор для примера 9.

q	x_1	x_2	x_3	f
1	0	0	0	0
2	0	0	1	0
3	0	1	0	0
4	1	0	0	0
5	1	1	0	1
6	1	0	1	0
7	0	1	1	1
8	1	1	1	1

Оба решения имеют похожие структуры – $\tilde{M}_1 \tilde{x}_1 \tilde{M}_2 \tilde{x}_2 \tilde{x}_3$. Оптимизационный функционал (4) для этой структуры:

$$(\tilde{m}2_{00}^0 + \tilde{m}2_{10}^0 + \tilde{m}2_{01}^0 + \tilde{m}2_{11}^0)\tilde{m}1_{00}^0 + (\tilde{m}2_{00}^1 + \tilde{m}2_{10}^1 + \tilde{m}2_{01}^1 + \tilde{m}2_{11}^1)\tilde{m}1_{10}^0 + \\ + \tilde{m}2_{00}^0 \tilde{m}1_{01}^0 + (\tilde{m}2_{00}^0 + \tilde{m}2_{10}^0 + \tilde{m}2_{11}^0)\tilde{m}1_{01}^1 + \tilde{m}2_{00}^1 \tilde{m}1_{11}^0 + (\tilde{m}2_{01}^1 + \tilde{m}2_{10}^1 + \tilde{m}2_{11}^1)\tilde{m}1_{11}^1.$$

Матрицы свертки для первого решения:

$$M_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, M_2 = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}.$$

Эти матрицы в точности соответствуют функции $f(x_1, x_2, x_3) = x_1 \wedge (x_2 \vee x_3)$. Матрицы свертки для второго МКО:

$$M_1 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, M_2 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}.$$

Эти матрицы соответствуют другому представлению той же функции:
 $f(x_1, x_2, x_3) = x_1 \wedge \overline{(x_2 \wedge x_3)}$. □

Пример 10. Изменяя значение функции f для обучающего примера q^7 в таблице 19 с 0 на 1, получаем обучающий набор для другой булевой функции:

$f(x_1, x_2, x_3) = (x_1 \wedge x_2) \vee (x_2 \wedge x_3) \vee (x_1 \wedge x_3)$. Оптимизационный функционал (4) $\forall IRM \in IRM_{M,2}$ для такого обучающего набора:

$$(\tilde{m}2_{00}^0 + \tilde{m}2_{10}^0 + \tilde{m}2_{01}^0)\tilde{m}1_{00}^0 + \tilde{m}2_{11}^0\tilde{m}1_{00}^1 + (\tilde{m}2_{00}^1 + \tilde{m}2_{10}^1 + \tilde{m}2_{01}^1)\tilde{m}1_{10}^0 + \tilde{m}2_{11}^1\tilde{m}1_{10}^1 + \\ + \tilde{m}2_{00}^0\tilde{m}1_{01}^0 + (\tilde{m}2_{01}^0 + \tilde{m}2_{10}^0 + \tilde{m}2_{11}^0)\tilde{m}1_{01}^1 + \tilde{m}2_{00}^1\tilde{m}1_{11}^0 + (\tilde{m}2_{01}^1 + \tilde{m}2_{10}^1 + \tilde{m}2_{11}^1)\tilde{m}1_{11}^1.$$

Решение задач (4), (5) дает максимальное значение 7 – это значит, что невозможно реализовать как минимум один из обучающих примеров. В частности, оба МКО, которые реализуют набор обучения для примера 8, достигают такого значения для целевой функции – реализуют все примеры обучения, кроме измененного примера q^7 . \square

Примеры 9 и 10 являются иллюстрацией для пары конкретных функций из класса булевых функций трех переменных. Всего таких функций 256, из которых только 20 являются неубывающими. Все они могут быть описаны с помощью таблицы истинности, например, как таблица 19. Мы провели вычислительный эксперимент, чтобы определить для каждой функции, можно ли реализовать ее с помощью МКО, определяя параметры соответствующего ей МКО.

В случае нереализуемой функции мы нашли МКО, который является наилучшим приближением для нее, и вычислили качество приближения.

Получилось, что среди всех 256 булевых функций трех переменных 152 реализуются через МКО и 56 из них – с любой из трех возможных древовидных структур. Оказалось, что нет такой булевой функции, которая была реализована в двух из трех возможных структур. Для всех 94 нереализуемых функций качество аппроксимации составляет $7/8$, т.е. только одна строка из ее таблицы истинности не реализуется путем аппроксимации МКО. Из 20 неубывающих функций только одна не реализуема – это функция голосования большинства. Остальные 19 неубывающих функций представимы в древовидной структуре, т.е. являются неповторными [30, 97].

Также проведена работа по аппроксимации булевых функций 4х переменных. Результаты представлены в таблицах 20 и 21. Всего исследовано 700 базовых функций, из которых все 65536 булевых функций четырех переменных могут быть получены элементарными преобразованиями.

Таблица 20 – Аппроксимации булевых функций 4х переменных первого класса (последовательные деревья).

Структура	Качество (макс 16)						Среднее	Среднее качество
	16	15	14	13	12	11		
$M_1l_1M_2l_2M_3l_3l_4$	59	173	248	181	39	0	14.04	0.8778
$M_1l_1M_2l_3M_3l_2l_4$	37	144	255	207	55	2	13.85	0.8656
$M_1l_1M_2l_4M_3l_2l_3$	57	195	267	158	23	0	14.15	0.8843
$M_1l_2M_2l_1M_3l_3l_4$	39	128	247	238	48	0	13.81	0.8635
$M_1l_2M_2l_3M_3l_1l_4$	29	94	203	262	104	8	13.51	0.8444
$M_1l_2M_2l_4M_3l_1l_3$	31	114	233	254	66	2	13.69	0.8557
$M_1l_3M_2l_1M_3l_2l_4$	31	112	225	260	70	2	13.66	0.8542
$M_1l_3M_2l_2M_3l_1l_4$	29	94	203	262	104	8	13.51	0.8444
$M_1l_3M_2l_4M_3l_1l_2$	41	136	247	230	46	0	13.85	0.8657
$M_1l_4M_2l_1M_3l_2l_3$	75	211	246	146	22	0	14.24	0.8902
$M_1l_4M_2l_2M_3l_1l_3$	49	167	245	187	50	2	13.96	0.8725
$M_1l_4M_2l_3M_3l_1l_2$	73	191	237	163	36	0	14.14	0.8841
Весь Класс 1	137	291	212	60	0	0	14.72	0.9200
Монотонные	86	49	14	0	0	0	15.48	0.9677

Таблица 21 – Аппроксимации булевых функций 4х переменных второго класса (непоследовательные деревья).

Структура	Качество (макс 16)						Среднее	Среднее качество
	16	15	14	13	12	11		
$M_1M_2l_1l_2M_3l_3l_4$	55	192	285	160	8	0	14.18	0.88625
$M_1M_2l_1l_3M_3l_2l_4$	33	134	255	238	40	0	13.83	0.86446
$M_1M_2l_1l_4M_3l_2l_3$	43	146	257	222	32	0	13.92	0.87017
Весь Класс 2	73	218	277	126	6	0	14.32	0.89517
монотонные	57	78	14	0	0	0	15.28	0.95553

4.4. Примеры анализа и синтеза для неполных наборов данных

На примере рассмотрения трех случаев синтеза МКО на основе трех неполных наборов данных демонстрируются возможности анализа данных (см., например, [126]). На примере синтеза первой модели, набора данных «О Зените» в двоичной шкале, мы покажем основной подход к работе с неполными наборами данных. На примерах второй и третьей задач покажем, какие еще подходы к анализу данных открываются при использовании предложенного метода. Данные для второго задания – оценка финансовой устойчивости ряда российских компаний в сфере строительства, см. [112], а для третьего – оценка дизайн-проектов студентов Уральского государственного архитектурно-художественного университета. (УРГАХУ) для условий крайнего севера из школы арктического дизайна, см. [129]. Данные по второму и третьему случаям представлены в троичной шкале. Начнем с рассмотрения примера по синтезу на данных задачи, представленной ранее в таблице 7.

Пример 11. Задача «О Зените». Как выше было сказано, в задаче рассматривается вопрос предсказания результата выступления футбольной команды в зависимости от четырех параметров, см. таблицу 7. В источнике [45] рассматривается широко известный подход с использованием деревьев решений. Мы же решим эту задачу с помощью синтеза механизма комплексного оценивания. Данный пример рассматривается как обучающий набор в единой шкале, согласованный и, как было сказано выше, неполный. Таким образом мы имеем всего 8 обучающих примеров вместо шестнадцати для полного набора, в шкале размером два, на четырех переменных.

В соответствии с (4), этот набор данных может быть реализован определенным МКО, если будут реализованы все примеры, описывающие этот набор. В результате расчета задачи синтеза оказалось, что эту задачу реализуют всего три структуры, из которых две последовательные и одна симметричная. В этой задаче у нас всего 8 примеров, следовательно реализованные примеры имеют значение восемь, см. таблицу 22. На оставшихся двенадцати структурах один из примеров не реализован, но мы можем аппроксимировать каждый из них.

Таблица 22 – Решение задачи синтеза для задачи «О Зените».

№	Структура	Матрицы			Количество реализованных примеров
		M_1	M_2	M_3	
1	$M_1l_1M_2l_2M_3l_3l_4$	1 0	0 0	0 1	7
		1 1	0 1	0 0	
2	$M_1l_1M_2l_3M_3l_2l_4$	1 1	1 0	1 0	7
		1 0	1 1	1 1	

Продолжение таблицы 22

№	Структура	M_1	M_2	M_3	Количество реализованных примеров
3	$M_1l_1M_2l_4M_3l_2l_3$	0 1 1 0	1 1 0 1	1 0 0 1	8
4	$M_1l_2M_2l_1M_3l_3l_4$	0 0 1 1	1 1 0 1	1 0 0 1	7
5	$M_1l_2M_2l_3M_3l_1l_4$	0 1 1 0	0 1 1 0	0 1 0 0	7
6	$M_1l_2M_2l_4M_3l_1l_3$	1 1 0 0	0 1 1 0	1 1 0 0	7
7	$M_1l_3M_2l_1M_3l_2l_4$	0 1 1 0	0 0 1 1	0 0 1 1	7
8	$M_1l_3M_2l_2M_3l_1l_4$	0 1 1 0	0 0 1 1	0 0 1 1	7
9	$M_1l_3M_2l_4M_3l_1l_2$	1 0 0 1	1 0 0 0	0 0 1 0	8
10	$M_1l_4M_2l_1M_3l_2l_3$	0 1 1 0	0 0 1 1	0 0 1 1	7
11	$M_1l_4M_2l_2M_3l_1l_3$	0 1 1 0	0 0 1 1	0 0 1 1	7

Проблема с неполным набором данных также может быть исследована с помощью групп эквивалентности. Рассмотрим две структуры из таблицы 22, тринадцатую и пятнадцатую. Видно, что тринадцатая реализуется, а пятнадцатая только аппроксимируется семью примерами. В таблицах 23 и 24 показано количество групп эквивалентности для каждой из них. Листьями дерева являются: l_1 - участник, l_2 - место, l_3 - лидер, l_4 - дождь; они соответствуют параметрам исходной задачи, значения в столбце КО соответствуют значениям столбца исходных данных «победа». Строки таблицы 22 расположены в таблицах 23 и 24 в качестве обучающих примеров. Столбцы l_1l_4 , l_1l_3 , l_2l_3 и l_2l_4 представляют разбиение на группы эквивалентности.

Таблица 23 – Рассмотрение групп нечувствительности для пятнадцатой структуры.

Номер примера	l_1	l_2	l_3	l_4	l_1l_4	l_2l_3	КО
1	0	0	0	0	0	0	0
2	0	1	0	0	0	1	1
3	1	1	0	0	1	1	0
4	0	0	1	0	0	1	1
5	1	1	1	0	1	0	1
6	0	1	0	1	0	1	1
7	1	0	1	1	2	1	0
8	1	1	1	1	2	0	0

Есть три группы эквивалентности для l_1l_4 и две для l_2l_3 .

Таблица 24 – Рассмотрение групп нечувствительности для тринадцатой структуры.

Номер примера	l_1	l_2	l_3	l_4	l_1l_3	l_2l_4	КО
1	0	0	0	0	0	0	0
2	0	1	0	0	1	0	1
3	1	1	0	0	0	0	0
4	0	0	1	0	0	1	1
5	1	1	1	0	0	1	1
6	0	1	0	1	1	0	1
7	1	0	1	1	0	0	0
8	1	1	1	1	0	0	0

Из таблицы 24 видно, что для тринадцатой структуры есть две группы эквивалентности для l_1l_3 и l_2l_4 каждая. Как было показано выше, для реализации механизма комплексного оценивания

количество групп эквивалентности должно быть меньше или равно размеру шкалы: $IRM_{L,G}(Q) \neq \emptyset$ тогда и только тогда когда $\forall i \in \{1, \dots, l-1\}$ верно $|K_L(Q)| \leq |K_L|$.

Что касается пятнадцатой структуры, то количество групп эквивалентности для пары листов l_1l_4 превышает размер шкалы, поэтому задача не может быть решена в рамках этой структуры, как это было продемонстрировано выше в таблице результатов синтеза. Для тринадцатой структуры, см. таблицу 24, количество групп эквивалентности соответствует размеру шкалы для пары l_1l_2 и для l_3l_4 , и эта структура реализует все примеры поставленной задачи. Анализ групп эквивалентности позволяет выделить заведомо нереализуемые структуры, в которых есть подгруппы переменных, не укладывающиеся в допустимый алфавит $|K_L|$.

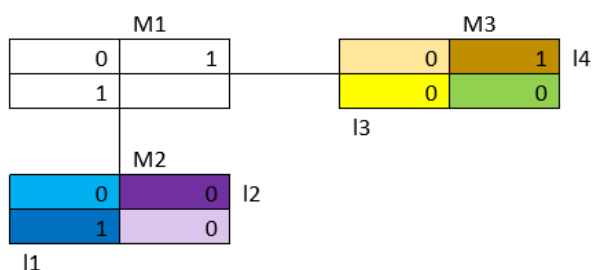


Рисунок 12 – Заполнение матриц дерева тринадцатой структуры.

Более того, как можно видеть из рисунка 12, в результате заполнения значений матриц, в матрице M_1 остается незаполненная ячейка, значение которой может быть определено расширением набора примеров. Например, выбор значения в нижней правой ячейке матрицы M_1 определит пойдет речь о монотонном или немонотонном наборе.

4.4.1. Задача «Оценка состоятельности компаний»

Следующая задача из области оценки состоятельности компаний. На основе работы [112] используются данные оценки компаний из области строительства в зависимости от пяти параметров: дебиторская задолженность, кредиторская задолженность, нераспределенная прибыль, запасы, основные средства. Исходно для выборки рассматривались двести российских предприятий строительной отрасли, из которых сто ликвидировано или находится в стадии ликвидации по причине банкротства и сто экономически благополучных, по которым дела о банкротстве не открывались. Для реализации процесса оценки финансового положения в работе использовались бухгалтерские балансы предприятий. В результате обработки данных осталось 48 примеров, повторяющиеся и противоречивые примеры были предварительно удалены при том, что для случая в троичной шкале на пяти листьях полный набор состоит из 243 примеров, см. таблицу 25.

Таблица 25 – Часть исходных данных для задачи «Оценка состоятельности компаний».

#	l_1	l_2	l_3	l_4	l_5	КО
1	0	0	0	0	0	0
2	0	0	0	2	0	0
3	0	0	1	2	0	1
4	0	1	1	0	0	0
5	0	1	1	0	1	0
6	0	1	1	1	0	0
...						

Здесь l_1 – дебиторская задолженность, l_2 – кредиторская задолженность, l_3 – нераспределенная прибыль, l_4 – запасы, l_5 – основные средства. Последний столбец – интегральный показатель финансового состояния компании – комплексная оценка. Оценки были закодированы экспертом в троичной системе: 0 – признак банкротства, 1 – признак компании с проблемой, 2 – признак финансово-устойчивой компании, по соответствующему критерию. Так как в этой задаче используются пять переменных, то мы рассматриваем сто пять возможных полных деревьев соответственно, что следует из формулы $|\Gamma_2(L)| = (2l - 3)!!$. Проверка всех возможных ста пяти структур полных деревьев с именованными листьями выявила, что лучший результат аппроксимации данной задачи реализует 45 из 48 примеров, и не существует структуры реализующей все 48 примеров. Лучшая реализация достигается на структуре: $M_1l_3M_2l_4M_3l_5M_4l_1l_2$ матрицы для нее приведены в таблице 26.

Таблица 26 – Матрица из решения для $M_1l_3M_2l_4M_3l_5M_4l_1l_2$.

Матрицы			
M_1	M_2	M_3	M_4
0 1 2	2 2 0	2 0 1	1 0 1
2 2 2	2 0 1	0 1 1	0 1 1
0 0 2	2 2 2	1 1 1	1 1 2

Из таблицы 26 видно, что множество немонотонно, так как только матрица M_1 может быть преобразована к монотонному виду. Для нее не реализуются три примера, с номерами 17, 19, 33

– это самые частотные из нереализующихся примеров. Гистограмма для нереализующихся примеров, для всех 105 структур данной задачи приведена на рисунке 13.

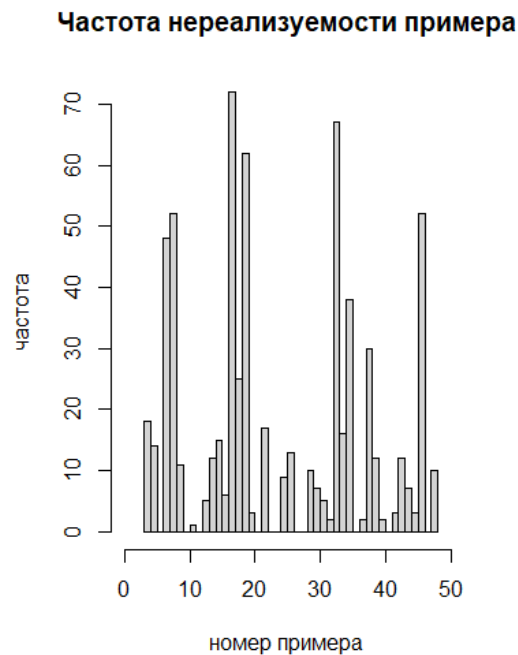


Рисунок 13 – Гистограмма нереализованных примеров для задачи «Оценка состоятельности компаний».

Можно обратить внимание, что структуры с некоторыми парами листьев дают лучшее качество аппроксимации. В данном случае нет пар листьев, которые реализуются на любой структуре, с точки зрения анализа групп эквивалентности это значит, что в некоторых структурах не хватает шкалы $|K_L(Q)| > |K_L|$, см. таблицу 27.

Таблица 27 – Качество аппроксимации для задачи «Оценка состоятельности компаний» в зависимости от использованных пар переменных.

Пара листьев	Макс.	Мин.	Среднее
l_1l_2	45	42	43,4
l_1l_3	43	39	41,6
l_1l_4	44	39	41,2

Продолжение таблицы 27

Пара листьев	Макс.	Мин.	Среднее
l_1l_5	44	40	42
l_2l_3	42	39	41
l_2l_4	43	39	41,7
l_2l_5	44	40	42,2
l_3l_4	44	40	42,1
l_3l_5	43	39	40,6
l_4l_5	43	39	41,6

Анализ пар переменных может использоваться для изучения декомпозиции критериев или для фильтрации нереализуемых структур.

В результате исследования набора данных данной задачи было получено, что МКО реализующий все заданные примеры может быть синтезирован в шкале 5, см. рисунок 14. А.О. Алексееву с коллегами [67] удалось синтезировать МКО, для данного набора, на основе предложенного ими метода в шкале 10.

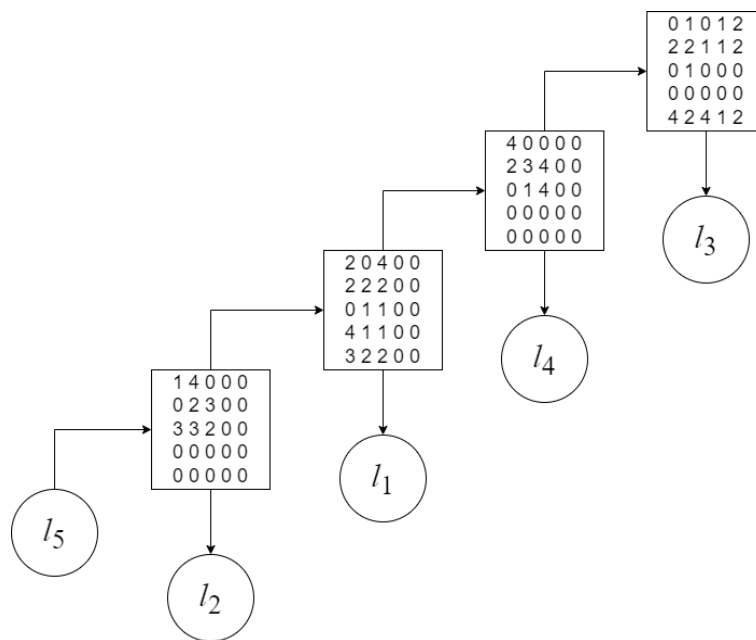


Рисунок 14 –МКО реализующий набор данных для задачи «Оценка состоятельности компаний».

4.4.2. Задача «Оценка дизайнерских проектов»

В третьей задаче рассматривается вопрос синтеза механизма комплексного оценивания на основе данных об оценках дипломных проектов для работ студентов дизайнеров из УрГАХУ посвященных освоению крайнего севера. В предоставленном наборе данных был представлен 31 пример из 243-х возможных для полного набора на пяти листьях в троичной шкале, в зависимости от пяти параметров: актуальность, экономика, этика/экология, эстетика, техническая часть. Дублирующиеся и противоречивые примеры были предварительно удалены, см. таблицу 28.

Таблица 28 – Часть исходных данных для задачи «Оценка дизайнерских проектов».

#	l_1	l_2	l_3	l_4	l_5	КО
1	2	1	1	2	0	2
2	1	0	0	1	0	1
3	1	1	2	0	2	2
4	2	1	2	2	1	2
5	2	1	2	2	1	2
6	1	1	0	1	1	2
...						

Здесь l_1 – актуальность, l_2 – экономика, l_3 – этика (экология), l_4 – эстетика (образ), l_5 – техническая часть. Кодирование оценок осуществлялось экспертом. Например, такие оценки как: «отлично с одобрением ГЭК», «лауреат международного конкурса», «рекомендовано ГЭК к внедрению», «одобрение ГЭК», «положительная оценка ГЭК», «реализовано в малом тираже» - получили оценку 2, «положительная оценка ГЭК» – 1, а «удовлетворительная оценка ГЭК» кодировалась нулем.

Были исследованы все 105 возможных структур и найден ряд структур с максимальным значением реализованных примеров. Рассмотрение групп листьев показывает, что сочетание критериев эстетики и технической части, а также эстетики и этики/экологии дает наилучшие результаты для ансамбля, см. таблицу 29.

Таблица 29 – Качество аппроксимации для задачи «Оценка состоятельности компаний» в зависимости от использованных пар переменных.

Пара листьев	Макс.	Мин.	Среднее
l_1l_2	31	29	30,4
l_1l_3	31	29	30,4
l_1l_4	31	30	30,7
l_1l_5	31	29	30,3
l_2l_3	31	29	30,6
l_2l_4	31	30	30,7
l_2l_5	31	29	30,5
l_3l_4	31	30	30,8
l_3l_5	31	29	30,4
l_4l_5	31	30	30,9

Из анализа результатов следует, что присоединение первого листа l_1 «актуальность» к верхней матрице M_1 позволяет реализовать набор в любой структуре. То есть с точки зрения декомпозиции параметр «актуальность» не зависит от других, мы можем его отделить. Если же в дереве к верхней матрице присоединяется l_4 «эстетика (образ)», то набор не реализуется ни в какой из возможных структур, что говорит о принципиальной недекомпозируемости, т.е. взаимозависимости критериев с параметром «эстетика (образ)». Если продолжить рассматривать декомпозицию структуры, то можно отметить, что в наборе нет структур, которые были бы реализуемы для выбранной пары листьев, в то время если к паре листьев присоединить еще один лист, то структуру с такой группой листьев уже можно найти среди реализованных структур.

4.4.3 Задача «Авиаторы»

В качестве еще одного примера, иллюстрирующего возможность применения МКО как моделей выбора с возможностью комплексного учета разнородных целевых показателей, рассмотрим МКО для оценки степени достижения приемлемого уровня эффективности обеспечения

безопасности полетов при определении уровня научно-технологического задела работ в авиастроении [37]. В качестве основы задачи была использована МКО, предложенная в работе [37], она представлена на рисунке 15. На основе предоставленного варианта МКО была восстановлена дискретная функция для всех возможных 625 примеров. Из сформированного таким образом набора данных, были отобраны 70 примеров, см. приложение А, на основе которых производилась дальнейшая работа по синтезу МКО.

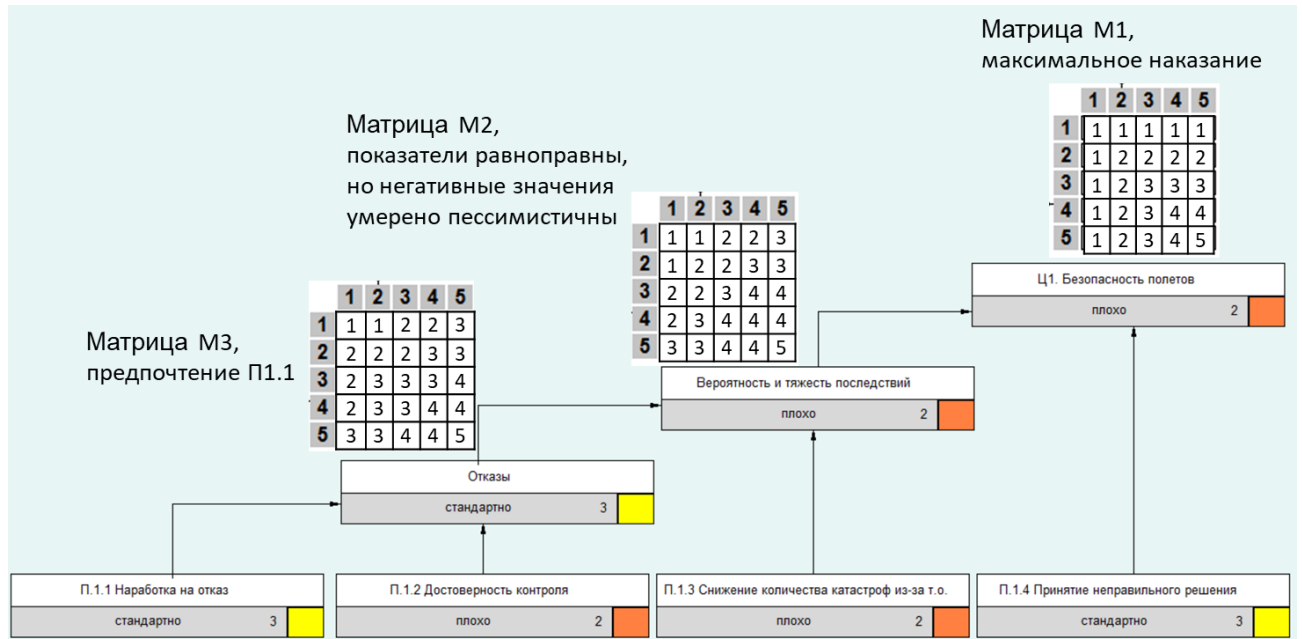


Рисунок 15 – МКО оценки степени достижения приемлемого уровня эффективности обеспечения безопасности полетов.

Общая схема подхода к задаче синтеза всех матриц рассматриваемой структуры МКО изложена в разделе 4.1. Для неполного обучающего набора данных, приведенного в приложении А, был проведен анализ групп эквивалентности. В таблице 30 приведены его результаты.

Таблица 30 – Группы эквивалентности на основе анализа неполного обучающего набора – 70 примеров из 625 возможных.

Группы переменных	Количество групп эквивалентности
П1.3,П1.4	4
П1.2,П1.4	4
П1.1,П1.4	3
П1.2,П1.3	4
П1.1,П1.3	3
П1.1,П1.2	3
П1.2,П1.3, П1.4	4
П1.1,П1.3, П1.4	5
П1.1,П1.2, П1.4	6
П1.1,П1.2, П1.3	3

Из таблицы следует, что для исследуемого набора данных не допустимы структуры МКО, в которых вместе группируются переменные П1.1, П1.2, П1.4 – для этих групп требуется шкала описания их влияния на итоговый результат мощностью не менее 6 позиций, в то время как в МКО допускается не более 5. Таким образом, из рассмотрения можно убрать структуры содержащие ветви вида: $MiP1.4MjP1.3P1.2$, $MiP1.3MjP1.4P1.2$, $MiP1.2MjP1.4P1.3$. Поэтому из списка структур полных двоичных деревьев на именованных листьях вычеркиваем структуры: $M1P1.3M2P1.4M3P1.2P1.1$, $M1P1.3M2P1.2M3P1.4P1.1$, $M1P1.3M2P1.1M3P1.4P1.2$. Остальные 12 передаются в блок синтеза матриц МКО.

В разделе 4.1 приведен порядок работы программного модуля анализа групп эквивалентности для отбора структур МКО. В данном случае, имеются четыре параметра, по которым производится оценка, соответственно необходимо рассмотреть 15 возможных структур МКО. На основе результатов анализа групп эквивалентности программный модуль отбирает 12 из 15 возможных структур – бинарных деревьев с четырьмя листьями, перечень которых приведен в таблице 31 – девять последовательных и три параллельные. Отобранные структуры передаются в модуль синтеза матриц МКО.

Таблица 31 – Перечень отобранных структур.

№ порядковый	№ структуры	Структура
1	1	$M1P1.4M2P1.3M3P1.2P1.1$
2	2	$M1P1.4M2P1.2M3P1.3P1.1$
3	3	$M1P1.4M2P1.1M3P1.3P1.2$
4	7	$M1P1.2M2P1.4M3P1.3P1.1$
5	8	$M1P1.2M2P1.3M3P1.4P1.1$
6	9	$M1P1.2M2P1.1M3P1.4P1.3$
7	10	$M1P1.1M2P1.4M3P1.3P1.2$
8	11	$M1P1.1M2P1.3M3P1.4P1.2$
9	12	$M1P1.1M2P1.2M3P1.4P1.3$
10	13	$M1M2P1.4P1.3M3P1.2P1.1$
11	14	$M1M2P1.4P1.2M3P1.3P1.1$
12	15	$M1M2P1.4P1.1M3P1.3P1.2$

Для неполного обучающего набора данных и 12 отобранных ранее структур программный модуль синтеза матриц МКО находит точное решение для пяти, приведенных в таблице 30. Еще для семи структур, не удастся найти точного решения с использованием всех обучающих примеров, поэтому приведены решения в виде матриц аппроксимирующих заданный набор данных.

Например, для первой структуры свертки, см строку 1 таблицы 32, в матрице 2 все значения для 2 (5) не определены, т.к. для определения влияния группы переменных П 1.1 и П 1.2 на неполном обучающем наборе было достаточно использовать шкалу из 4-х значений. Так как для других 4-х структур во всех матрицах необходимо было использовать шкалу из 5 значений, можно сделать вывод, что первая структура наиболее «экономно» описывала данный неполный обучающий набор. Кроме того, для первой структуры, соответствующей структуре исходной МКО (см. рисунок 15), на основе которой был сгенерирован полный обучающий набор, матрица M_1 , на основе которой определяется итоговое значение комплексной оценки, во всех ячейках, для которых определено значение шкалы комплексного оценивания, в точности совпадает с матрицей M_1 исходной МКО на рисунке 15. Это иллюстрирует возможность предлагаемого подхода к точному восстановлению МКО, синтезированных экспертным путем, на основе ограниченного числа обучающих примеров.

Таблица 32 – Результаты, полученные из блока синтеза матриц МКО.

№	Структура	Матрицы			Количество реализованных примеров
		M_1	M_2	M_3	
1	$M_1П1.4M_2П1.3M_3П1.2П1.1$	$x\ 1\ 1\ x\ 1$ $1\ 2\ 2\ 2\ 2$ $x\ 2\ 3\ 3\ 3$ $1\ 2\ 3\ 4\ 4$ $1\ 2\ x\ 4\ 5$	$1\ 5\ 2\ 2\ 4$ $5\ 1\ 3\ 3\ 4$ $2\ 2\ 3\ 4\ 4$ $2\ 2\ 4\ 4\ 5$ $x\ x\ x\ x\ x$	$2\ 1\ 2\ 4\ 3$ $1\ x\ 4\ 3\ 2$ $2\ 3\ 4\ 4\ 4$ $1\ 4\ 3\ 4\ 1$ $4\ 2\ 4\ 2\ 4$	70
2	$M_1П1.4M_2П1.2M_3П1.3П1.1$	$1\ 2\ 3\ 2\ 4$ $x\ 2\ 3\ 3\ 5$ $1\ 1\ 1\ 4\ 1$ $1\ 2\ x\ 4\ 3$ $1\ 2\ 2\ 2\ 2$	$2\ 4\ 1\ 1\ 1$ $3\ 3\ 4\ 1\ 2$ $1\ 5\ 1\ 3\ 4$ $5\ 5\ 4\ 4\ 4$ $1\ 5\ 5\ 5\ 2$	$2\ 2\ 5\ 4\ 1$ $x\ 1\ 4\ 1\ x$ $5\ 5\ 1\ 1\ 3$ $5\ 3\ 3\ 1\ 2$ $5\ 5\ 3\ 3\ 2$	70
3	$M_1П1.4M_2П1.1M_3П1.3П1.2$	$1\ 2\ 3\ 4\ 4$ $1\ 2\ 2\ 2\ 2$ $x\ 2\ x\ x\ 3$ $1\ 1\ 1\ 3\ 1$ $1\ 2\ 3\ 4\ 5$	$5\ 3\ 1\ 1\ 1$ $3\ 1\ 4\ 1\ 1$ $1\ 2\ 2\ 2\ 5$ $4\ 1\ 4\ 4\ 3$ $2\ 3\ 5\ 2\ 2$	$2\ 4\ 3\ 2\ 1$ $4\ 3\ x\ 5\ 5$ $5\ 3\ 2\ 2\ 1$ $5\ 3\ 5\ 2\ 1$ $3\ 4\ 1\ 1\ 3$	70

Продолжение таблицы 32

№	Структура	M_1	M_2	M_3	Количество реализованных примеров
4	$M_1\Pi_1.2M_2\Pi_1.4M_3\Pi_1.3\Pi_1.1$	3 2 3 4 4 2 1 4 3 3 3 2 2 2 3 2 1 2 2 2 1 1 1 x 1	x x x 1 2 5 4 5 4 2 x x 3 2 4 5 4 3 3 1 5 3 1 1 1	2 2 4 4 4 x 2 2 4 x 4 3 4 4 5 3 4 2 5 3 3 4 1 5 4	68
5	$M_1\Pi_1.2M_2\Pi_1.3M_3\Pi_1.4\Pi_1.1$	3 2 3 4 5 2 1 1 2 3 3 4 4 4 4 1 2 2 2 1 1 1 2 2 2	2 1 2 1 3 2 4 2 1 2 5 5 3 3 1 5 2 2 5 5 x 4 4 2 1	2 4 2 x 1 4 4 x 4 2 4 4 5 3 3 5 2 1 1 3 4 4 1 3 3	67
6	$M_1\Pi_1.2M_2\Pi_1.1M_3\Pi_1.4\Pi_1.3$	2 1 1 4 1 1 1 4 2 2 2 2 2 2 4 2 2 3 3 3 3 4 3 4 5	3 x 4 x 5 1 4 3 1 2 2 3 2 3 3 4 1 1 1 1 4 x 5 2 3	3 3 2 x 3 4 3 3 3 2 4 2 5 3 2 2 3 1 5 4 2 2 5 5 1	67
7	$M_1\Pi_1.1M_2\Pi_1.4M_3\Pi_1.3\Pi_1.2$	1 4 3 4 3 2 2 2 2 4 3 2 2 2 2 4 3 4 2 5 1 1 1 1 1	5 3 3 1 4 x 2 3 2 3 5 3 1 4 2 5 2 1 2 3 5 1 1 1 1	5 3 2 5 4 5 2 x 3 1 2 4 3 4 1 1 2 1 5 3 2 4 1 4 1	69
8	$M_1\Pi_1.1M_2\Pi_1.3M_3\Pi_1.4\Pi_1.2$	2 2 2 2 2 3 4 3 4 2 3 3 4 2 4 1 1 1 1 1 4 2 2 2 3	1 4 1 5 2 4 1 1 3 3 4 4 4 4 4 1 1 3 2 3 4 5 3 1 5	3 2 1 4 1 3 5 2 2 5 3 5 1 1 4 x 2 2 2 4 3 1 4 2 5	68

Продолжение таблицы 32

№	Структура	M_1	M_2	M_3	Количество реализованных примеров
9	$M_1\Pi 1.1M_2\Pi 1.2M_3\Pi 1.4\Pi 1.3$	2 2 2 1 4 1 1 3 4 1 2 2 1 2 2 3 3 4 1 3 1 3 2 2 2	5 5 3 1 1 3 3 1 5 4 4 1 4 2 2 3 2 5 3 5 2 4 2 4 1	2 1 1 x 4 5 4 1 1 2 1 2 2 1 2 5 1 3 5 3 4 4 2 5 1	66
10	$M_1M_2\Pi 1.4\Pi 1.3M_3\Pi 1.2\Pi 1.1$	x 2 2 1 1 4 3 4 2 1 5 4 3 2 1 x 4 2 2 1 4 3 4 2 1	5 4 4 x 4 5 4 4 4 4 5 3 2 2 3 5 4 2 2 2 5 4 2 2 1	1 1 2 4 5 4 x 4 3 3 2 3 5 4 2 1 3 4 4 1 2 4 5 2 3	70
11	$M_1M_2\Pi 1.4\Pi 1.2M_3\Pi 1.3\Pi 1.1$	2 2 1 2 2 x 4 1 2 3 2 5 1 4 3 4 3 2 3 x 2 x 1 2 1	3 4 5 5 3 3 1 5 4 3 3 1 5 1 4 x 4 2 4 4 3 1 5 4 2	5 5 2 4 2 x 1 4 3 x 5 1 2 3 3 2 1 1 3 4 1 1 3 3 3	69
12	$M_1M_2\Pi 1.4\Pi 1.1M_3\Pi 1.3\Pi 1.2$	1 5 4 3 2 1 x x 4 4 1 3 3 3 2 1 2 1 1 2 1 4 2 2 2	1 3 4 x 3 1 5 x 3 4 1 5 4 2 3 1 4 4 3 5 1 5 4 3 2	5 4 5 3 1 4 5 x 5 2 4 5 5 3 1 1 5 1 2 5 5 1 1 1 1	70

Символ x обозначает неиспользуемую ячейку. Таким образом, в результате синтеза мы получили пять механизмов реализующих входной набор данных. Еще для семи структур удалось найти матрицы, аппроксимирующие искомый механизм на меньшем количестве входных примеров обучающей выборки. Из списка реализованных структур видно, что для последовательных структур присоединение индикатора $\Pi 1.4$ к верхней матрице позволяет реализовать МКО. Таким образом, из 12 потенциально реализующихся, на основе анализа групп эквивалентности структур, оказались синтезируемы только 5. Также были исследованы несколько наборов данных со 100 случайно отобранными примерами. Во всех случаях, в список реализованных

попала структура МКО, соответствующая структуре, выбранной при экспертном синтезе МКО для данной задачи.

4.5. Прогнозная система основанная на моделях МКО

Для практических задач довольно типична ситуация, когда в результате процедуры синтеза МКО на множестве рассматриваемых структур находятся несколько МКО реализующих заданный набор данных. В данном разделе предлагается подход к построению прогнозной системы на основе использования результатов, получаемых от нескольких МКО, реализующих заданный набор дискретных данных. Подобно методу случайного леса, оценки от нескольких МКО агрегируются в единую оценку, однако вместо решающих деревьев используются синтезированные механизмы комплексного оценивания. В качестве иллюстрации используются результаты рассмотренной выше задачи «Авиаторы», см. рисунок 16. Построение прогнозной системы на основе одной МКО – вырожденный случай модели прогнозной системы. Таким образом, данный подход к прогнозированию является естественным развитием задачи синтеза МКО.

Для каждого из примеров данного набора данных k на основе значений комплексных оценок формируется следующая информация: доля МКО, определивших комплексную оценку в рамках одной из возможных категорий допустимых значений комплексной оценки; среднее значение оценки по всем категориям; доля МКО, допускающих произвольное значение комплексной оценки для данной комбинации значений входных показателей k ; относится ли рассматриваемый пример к множеству обучающих примеров. При этом не происходит переход к количественной оценке, т.к. агрегация оценок от разных моделей в единый показатель хотя и производится с оценками в дискретных величинах, но данные оценки кодируют качественные значения. Для конкретного введенного примера полученная информация выводится в интерфейсе прогнозной системы в соответствующих полях.

В качестве иллюстрации в данном разделе используются результаты синтеза МКО для задачи «Авиаторы», рассмотренной в разделе 4.4. Интерфейс системы прогнозирования представлен на рисунке 16. На демонстрируемом экране выводится информация о принадлежности комбинации значений входных переменных обучающей выборке и прогнозу, ожидаемое усредненное значение в случае увеличения или уменьшения значения одного из входных показателей. Для представленного на рисунке примера прогноза значения комплексной оценки для значения входных переменных П1.1=1, П1.2=2, П1.3=2, П1.4=3, отсутствующего в обучающем наборе две из пяти МКО предсказали значение комплексной оценки как 1, по одной МКО – 2 и 3, и одна МКО не

содержала информации о возможном значении комплексной оценки. При этом из системы прогнозирования следует, что увеличение значения показателя 1.4 позволит повысить значение ожидаемой комплексной оценки по совокупности всех пяти МКО, в то время как с точки зрения снижения ожидаемого значения комплексной оценки наиболее значимым является снижение одного из показателей 1.3 или 1.2.

показатель	Исходные показатели					Прогноз категории по моделям						пример из обучающей выборки	
	П1.4	П1.3	П1.2	П1.1		среднее значение	категория 1	категория 2	категория 3	категория 4	категория 5		категория не определена
прогноз среднего значения категории при улучшении показателя	1.50	0.25	0.25	0.25									
текущее значение (1-5)	3	2	2	1	3221	1.75	40%	20%	20%	0%	0%	20%	нет
прогноз среднего значения категории при ухудшении показателя	-0.15	-0.75	-0.75	0.00									

Рисунок 16 – Интерфейс системы прогнозирования комплексной оценки на основе набора моделей МКО, синтезированных на неполном наборе данных.

Из 555 примеров полного набора, не вошедших в обучающий набор, система прогнозирования на основе пяти синтезированных МКО верно предсказывает значения 210 примеров, при этом для 93 из них наблюдается единогласие – все пять МКО определяют значение комплексной оценки такое же, как и в исходном полном наборе. Для одного примера все пять МКО единогласно ошибаются. Для тех примеров из полного набора, для которых система прогнозирования дает неточный прогноз, средняя ошибка прогнозирования составила менее одной категории (0,62).

4.6. Выводы по главе 4

В данной главе приведено описание работы модулей программного комплекса для синтеза МКО, на основе которых изложен общий алгоритм синтеза. С помощью программно реализованного алгоритма синтеза МКО, рассмотрен ряд задач по синтезу МКО на основе полных и неполных наборов данных. Наряду с тестовыми примерами по синтезу дискретных функций, на основе полных таблиц истинности рассмотрены примеры синтеза МКО на неполных наборах: оценки дизайнерских проектов, оценки жизнеспособности компаний из сферы строительства, восстановления МКО для оценки степени достижения приемлемого уровня

эффективности обеспечения безопасности полетов при определении уровня научно-технологического задела работ в авиастроении на основе примерно 10% примеров от исходной выборки. В последнем разделе главы демонстрируется подход по синтезу прогнозной системы на основе группы синтезированных моделей МКО. В рассмотренных задачах применялись алгоритмы и методы, описанные во второй и третьей главах данной работы, а для вычисления максимума оптимизационного функционала использован решатель Gurobi.

На приведенных примерах показано, что предлагаемый подход к анализу позволяет:

- Исследовать возможность декомпозирующего (древовидного) представления модели, описывающей предоставленные данные.
- Проводить структурный анализ вклада отдельных переменных в итоговое значение.
- Исследовать анализ влияния отдельных переменных и групп переменных на итоговое значение агрегирующего показателя.
- Исследовать монотонность обучающего набора дискретных данных.
- Выделять целевые примеры для дальнейшего уточнения параметров МКО.
- Прогнозировать результаты для недостающих примеров как по отдельным МКО, так и по совокупности нескольких МКО.

Эта функциональность вполне применима ко многим областям организационной деятельности, как например: экологическая оценка, оценка уровня инновационного развития отрасли, оценка проектов НИОКР, оценка программ регионального развития, в системе управления развитием организации, оценка проектов в сфере малого и среднего предпринимательства, оценка рисков и многих других.

Заключение

Главная цель, проработанная в данной диссертационной работе – исследование вопроса синтеза механизмов комплексного оценивания на основе набора обучающих примеров как принципиально новый подход к синтезу МКО. В процессе решения обозначенного вопроса были поставлены задачи анализа отдельных элементов МКО, что привело к получению инструментов анализа дискретных данных на основе МКО. Основные результаты диссертационной работы.

1. Разработаны методы записи, синтеза и сокращения множества рассматриваемых структур полных бинарных деревьев: метод поиска числа групп эквивалентности, метод таблицы ветвей, методы генерации структур. Предложенные методы позволяют выбирать перспективные структуры для задачи синтеза матриц МКО.
2. Разработаны методы синтеза матриц МКО заданной размерности на основе наборов дискретных данных: методы синтеза матриц рассматриваемой структуры МКО для случаев полного и неполного наборов данных, декомпозиционный метод синтеза для неполного набора данных.
3. Предложен общий алгоритм анализа и синтеза МКО. Алгоритм разработан с использованием предложенных алгоритмов анализа и синтеза, и впервые реализован в виде комплекса программ. Разработанный комплекс программ позволяет осуществлять анализ и синтез механизмов комплексного оценивания на основе наборов дискретных данных.

Список литературы

1. Акинфеев В.К., Цвиркун А.Д. Программный комплекс «ТЭО-ИНВЕСТ» // Проблемы управления инвестициями. – 2013. – № 4. – С. 32–40.
2. Алексеев А.О., Харитонов В.А., Ясницкий В.Л. Разработка концепции комплексного нейросетевого моделирования процессов массовой оценки и сценарного прогнозирования рыночной стоимости жилой недвижимости // Известия вузов. Инвестиции. Строительство. Недвижимость. – 2018. – № 1 (24). – С. 11–22.
3. Алексеев А.О. и др. Алгоритмические основы нечеткой процедуры комплексного оценивания объектов различной природы // Фундаментальные исследования. – 2014. – № 3-3. – С. 469–474.
4. Алексеев А.О. Применение механизмов комплексного оценивания и матричных неанонимных обобщенных медианных механизмов для согласования интересов агентов / А.О. Алексеев, Т.А. Катаева // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». – 2021. – Т. 21, № 3. – С. 75–89.
5. Андронникова Н.Г., Баркалов С.А., Бурков В.Н., Котенко А.М. Модели и методы оптимизации региональных программ развития. М.: ИПУ РАН, 2001. – 60 с.
6. Андронникова Н.Г., Бурков В.Н., Леонтьев С.В. Комплексное оценивание в задачах регионального управления – М.: ИПУ РАН, 2002. – 58 с.
7. Арнольд В.И. О функциях трех переменных // ДАН СССР. – 1957. – т. 114. – № 4. – С. 679–681.
8. Беляков А. Ю., Елохова И. В., Мерсон М. Э., Харитонов В. А. Транзитивные замыкания на деревьях комплексного оценивания // Управление большими системами: сборник трудов. – 2004. – вып. 9. – С. 53–56.
9. Блачев Р. Н., Особенности процедуры бинарной агрегации многокритериальных экспертных оценок // Автомат. и телемех. – 1997. – вып. 5. – С. 126–132.
10. Бурков В.Н. Проблемы синтеза механизма комплексного оценивания на основе обучающего набора данных / В.Н. Бурков, Н.А. Коргин, О.Л. Марин // XIII Всероссийское совещание по проблемам управления (ВСПУ–2019): сб. тр. (Москва, 17–20 июня 2019 г.) / Ин-т проблем управления им. В.А. Трапезникова РАН. – М.: ИПУ РАН, 2019. – С. 2280–2284.
11. Бурков В.Н., Буркова И.В., Коргин Н.А., Щепкин А.В. Модели согласованного комплексного оценивания в задачах принятия решений // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». – 2020. – Т. 20, № 2. – С. 5–13.

12. Бурков В.Н., Буркова И.В., Попок М.В. Метод дихотомического программирования // Управление большими системами. – 2004. – № 9. – С. 57–75.
13. Бурков В.Н., Исаков М.Б., Коргин Н.А. Применение обобщенных медианных схем для построения неманипулируемых механизмов активной экспертизы // Проблемы управления. – 2008. – № 4. – С. 38–47.
14. Бурков В.Н. Метод синтеза системы комплексного оценивания / В.Н. Бурков, И.В. Буркова, А.В. Щепкин // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». – 2020. – Т. 20, № 4. – С. 63–73.
15. Бурков В.Н., Новиков Д.А., Щепкин А.В. Механизмы управления эколого-экономическими системами / Под ред. академика С.Н. Васильева. – М.: Издательство физико-математической литературы, 2008. – 244 с.
16. Бурков В.Н., Сергеев В.А., Коргин Н.А. Идентификация механизмов комплексного оценивания на основе унитарного кода // Управление большими системами: сборник трудов. – 2020. – № 87. – С. 67–85.
17. Бурков В.Н., Щепкин А.В. Экологическая безопасность. – М.: ИПУ РАН, 2003. – 92 с.
18. Бурков В.Н., Щепкин А.В., Амелина К.Е., Даулбаева З.М., Рязанцев С.А. Комплексный механизм управления развитием организации // Вестник ЮУрГУ. Серия: Компьютерные технологии, управление, радиоэлектроника. – 2019. – №3.
19. Бурков В.Н., Гореликов Н.И., Черкашин А.М. Методические основы комплексной оценки результатов деятельности предприятий с учетом их прогрессивности в ВПО «Союзэлектроприбор» // Приборы и системы управления. – 1982. – №11. – С. 21.
20. Бурков В.Н., Тесарж И., Трубка Я., Цыганов В.В., Чамски Б., Черкашин А.М. Проектирование систем комплексной оценки инноваций и принятия решений / В кн.: Моделирование и идентификация производственных систем. – М.: ИПУ, 1988. – С. 49–55.
21. Буркова И. В., Дранко О. И., Крюков С. В., Струков А. Ю. Дихотомическое представление при комплексной оценке предприятий // Вестник ВГТУ. – 2010. – № 11.
22. Выхованец В.С. Алгебраическая декомпозиция дискретных функций // Автоматика и телемеханика. – 2006. – №3. – С. 20–56.
23. Выхованец В.С. Спектральная идентификация дискретных систем // Труды VIII-й Международной конференции «Идентификация систем и задачи управления»: РАСО 2009. – М.: ИПУ РАН, 2009. – С. 1500–1517.
24. Гвишиани Д.М. Многокритериальные задачи принятия решений / Под ред. Д.М. Гвишиани, С.В. Емельянова. – М.: Машиностроение, 1978. – 192 с.

25. Глотов В.А., Гречко В.М., Павельев В.В. Метод принятия решений при лингвистических переменных / Многокритериальные задачи принятия решений. под ред. Гвишиани Д.М., Емельянов С.В. – М.: Машиностроение, – 1978. – 192 с.
26. Глотов В.А., Павельев В.В. Векторная стратификация. – М.: Наука, 1984.
27. Глотов В.А., Павельев В.В. Комплексное оценивание многомерных объектов. – М: Институт проблем управления, 1984.
28. Гореликов Н.И. Проблемы совершенствования отраслевого механизма управления разработкой и производством новой продукции // Автоматика и телемеханика. – 1984. – №5. – С. 63–70.
29. Губко М.В. Математические модели формирования рациональных организационных иерархий // Автоматика и телемеханика. – 2008. – №9. – С. 114–139.
30. Губко М. В. Алгоритмы построения субоптимальных организационных иерархий // Автоматика и телемеханика. – 2009. – № 1. – С. 162–179.
31. Гурвич В.А. О неповторных булевых функциях // Успехи математических наук. – 1977. – Т. 32, №1 – С. 183–184.
32. Казакова Е. А. Формирование матричных механизмов комплексного оценивания в региональном управлении: диссертация ... кандидата технических наук: 05.13.10 / Казакова Е. А. – Воронеж, 2010. – 125 с.
33. Казакова Е.А., Курочка П.Н., Половинкина А.И. Автоматизированное построение матричных процедур комплексного оценивания на основе оптимизационного подхода // Вестник Воронежского государственного технического университета. – 2010. – Т. 6. – №10.
34. Кафтан Д.В. Древесное представление неповторных функций в расширенных элементарных базисах / Д. В. Кафтан // Известия высших учебных заведений. Поволжский регион. Физико-математические науки. – 2017. – № 3 (43). –С. 37–49.
35. Колмогоров А.Н. О представлении непрерывных функций нескольких переменных суперпозициями непрерывных функций меньшего числа переменных // ДАН СССР. – 1956. – т. 108. – № 2. – С. 179–182.
36. Коргин Н.А., Кравчук С.Г., Сергеев В.А. Проблемы согласования интересов в проектах развития инфраструктуры, обеспечивающей функционирование легкого внедорожного электротранспорта / Материалы 14-й Мультиконференции по проблемам управления (МКПУ-2021, Дивноморское, Геленджик). – 2021. – Т. 2. – С. 155–157.
37. Коргин Н.А. Рождественская С.М. Методика оценки и ранжирования проектов при формировании программ создания научно-технического задела // Вторая научно-практическая конференция «Управление созданием научно-технического задела в

- жизненном цикле высокотехно-логичной продукции». Сборник докладов. ИПУ РАН им. В.А. Трапезникова. – М., 2017. С. 58–64.
38. Коргин Н.А., Сергеев В.А. Выбор структур при решении задач идентификации механизмов комплексного оценивания для полных наборов данных / Материалы 14-й Мультиконференции по проблемам управления (МКПУ-2021, Дивноморское, Геленджик). – 2021. – Т. 2. – С. 158–160.
 39. Краскова, Н. И. Использование процедур комплексного оценивания при реализации государственной политики в сфере малого и среднего предпринимательства // Вестник СамГУ. – 2013. – № 1 (102).
 40. Кузнецов А.В. О неповторных контактных схемах и неповторных суперпозициях функций алгебры логики // Тр. МИАН СССР, 1958. – том 51. – С. 186–225.
 41. Ларичев О.И. Теория и методы принятия решений, а также Хроника событий в Волшебных странах: Учебник. Изд. второе, перераб. и доп. - М.: Логос, 2002. – С. 392.
 42. Ларичев С.И. Вербальный анализ решений / О.И. Ларичев; [отв. ред. А.Б. Петровский]; Ин-т системного анализа РАН. - М.: Наука, 2006. – 181 с.
 43. Левинталь А.Б., Ефременко В.Ф., Гусев В.Б., Павельев В.В., Пащенко Ф.Ф. Комплексное оценивание и планирование развития региона – М., 2006.
 44. Макаров В.М. Теория менеджмента: Учеб. пособие. Спб.: Изд-во Политехн. ун-та, 2012. – 125 с.
 45. Николенко С., Тулупьев А. Самообучающиеся ситемы. – М.: МЦНМО, 2009. – 288 с.
 46. Новиков Д.А. Теория управления организационными системами. – М.: МПСИ, 2005. – 584 с.
 47. Павельев В.В. Структурная идентификация целевой функции в задачах выбора многопараметрических объектов // Труды IX Международной конференции «Идентификация систем и задач управления» SICPRO '12. – 2012.
 48. Петровский А. Б. Теория принятия решений: учебник для студ. высш. учеб. заведений / А. Б. Петровский. — М.: Издательский центр «Академия», – 2009. – 400 с.
 49. Подиновская О.В., Подиновский В.В. Анализ иерархических многокритериальных задач принятия решений методами теории важности критериев // Проблемы управления. – 2014. –№ 6. – С. 2–8.
 50. Подиновский В.В., Подиновская О.В. О некорректности метода анализа иерархий // Проблемы управления – № 1. – 2011.
 51. Пуликовский К.Б., Щепкин А.В. Комплексная оценка соответствия опасных производственных объектов требованиям безопасности // Безопасность труда в промышленности. – 2007. – №4. – С. 2–7.

52. Сергеев В.А. Идентификация механизмов комплексного оценивания с применением разделительной декомпозиции // Труды 18-ой Всероссийской школы-конференции молодых ученых «Управление большими системами» (УБС'2022, Челябинск). – 2022. – С. 546–552.
53. Сергеев В.А. Синтез механизмов комплексного оценивания на основе разделительной декомпозиции // Проблемы управления. – 2022. – № 6. – С. 3–13.
54. Сергеев В.А., Коргин Н.А. Сборщик полных бинарных деревьев на именованных листьях на основе данных анализа групп эквивалентности: Свидетельство о государственной регистрации программы для ЭВМ № 2023618561 РФ; Зарег. 26.04.2023.
55. Сергеев В.А. Коргин Н.А. Исследование чувствительности дискретной функции для решения задачи синтеза МКО на основе дискретных наборов данных / Материалы 16-й Мультиконференции по проблемам управления (МКПУ-2023, Волгоград). – 2023. – Т. 2. – С. 345–348.
56. Спирина В. С., Алексеев А. О., Андропова А. А. Анализ и прогнозирование управленческих решений в управлении коммерческой недвижимостью // Изв. Саратов. ун-та. Нов. сер. Сер. Экономика. Управление. Право. – 2019. – Т. 19, вып. 3. – С. 274–285.
57. Трапезников В.А., Гореликов Н.И., Бурков В.Н., Зимоха В.А., Толстых А.В., Черкашин А.М., Цыганов В.В. Комплексный подход к управлению научно-техническим прогрессом в отрасли // Вестник АН СССР. – 1983. – № 3. – С. 33–43.
58. Трахтенброт Б. А. К теории неповторных контактных схем, Тр. МИАН СССР, 1958. – том 51. – С. 226–269
59. Тутыгин А. Г., Коробов В. Б. Преимущества и недостатки метода анализа иерархий // Известия Российского государственного педагогического университета им. А. И. Герцена. – 2010.
60. Уандыкова М.К. Формирование интегральной оценки уровня инновационного развития отрасли / М.К. Уандыкова, А.Д. Елеукулова // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». – 2017. – Т. 17, № 3. – С. 75–87.
61. Фияшко К.С., Кривогино Д.Н. Метод комплексного оценивания экологической составляющей среды на примере Кировского района г. Пермь. Известия вузов. Инвестиции. Строительство. Недвижимость. – 2021. – Т. 11. – № 1. – С. 84–93.
62. Харитонов В. А., Алексеев А. О. Количественный анализ уровней риска на основе универсальной бинарной модели предпочтения лицом, принимающим решение (ЛПР) // Вестник ПГУ. Серия: Экономика. – 2009. – № 2.
63. Харитонов В.А., Дмитриюков М.С., Ларионова Р.А. Алгоритмы интеллектуальной поддержки принятия согласованных инвестиционных решений в задачах управления

- объектами культурного наследия // Вестник Пермского университета. Сер. «Экономика» = Perm University Herald. Economy. – 2016. – № 3(30). – С. 61–76.
64. Харитонов В.А., Белых А.А. Технологии современного менеджмента – Пермь: Изд-во Перм. гос. техн. ун-та, 2007. – 190 с.
 65. Adadi A., Berrada M. Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). // IEEE Access. – Vol 6. – 2018. – P. 52138-52160.
 66. Alekseev A.O. Identification of Integrated Rating Mechanisms Based on Training Set // 2020 2nd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA). – 2020. – P. 398-403.
 67. Alekseev A., Alekseeva I., Noskova A., Chugainova A. Detection of the Financial Patterns Based on Integrated Solvency Scoring Systems // 2021 3rd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA). – 2021. – P. 607-612.
 68. Alekseev A., Galiaskarov E., Koskova K. Application of the Matrix Rating Mechanisms and System Cognitive Analysis Methods at the Task of Residential Real Estate Conceptual Designing. // IEEE 21st Conference on Business Informatics (CBI). – 2019. – Vol. 2. – P. 111–116.
 69. Angluin D., Hellerstein L., Karpinski M. Learning read-once formulas with queries. Technical report, UC Berkeley, Report No. 89/528. – 1989.
 70. Angluin D., Kharitonov M., When won't membership queries help? // J. Comput. Syst. Sci. 50. – 1991. – P. 336-355.
 71. Ashenhurst R. The Decomposition of Switching Functions // Technical report, Bell Laboratories BL-1. – 11. – 1952. – P. 541–602.
 72. Beekman J. Multiple Criteria Decision Analysis Software // Software Survey. – 2020.
 73. Belle V., Papantonis I. Principles and Practice of Explainable Machine Learning // Front. Big Data. – Vol. 4 68896. – 2021.
 74. Belton V., Stewart T. Multiple Criteria Decision Analysis: An Integrated Approach. Springer Science & Business Media, 2002.
 75. Blum A., Hellerstein L., Littlestone N. Learning in the presence of finitely or infinitely many irrelevant attributes / Journal of computer and system sciences. – vol. 50. – 1995. – P. 32–40.
 76. Bohanec M., Zupan B. A function-decomposition method for development of hierarchical multi-attribute decision models // Decision Support Systems. – vol. 36. – 2004. – P. 215–233.
 77. Borchering K., Thomas Eppel, Detlof von Winterfeldt. Comparison of Weighting Judgements in Multiattribute Utility Measurement // Management Science. – vol. 37, no. 12. – 1991. – P. 1603–19.

78. Budescu D., Decisions Based on Numerically and Verbally Expressed Uncertainties // *Journal of Experimental Psychology: Human Perception and Performance*. – 1988. – Vol. 14, No. 2. – P. 281–294.
79. Burkov V.N., Korgin N.A., Sergeev V.A. Identification of Integrated Rating Mechanisms as Optimization Problem // *Proceedings of 13th International Conference "Management of large-scale system development" (MLSD)*. – 2020. – P. 1–5.
80. Craswell E., Bonnell M., Bossio D., Demuth S., van de Giesen, N. *Integrated Assessment of Water Resources and Global Change: A North-South Analysis*. – Vol. 1. – 2007.
10.1007/978-1-4020-5591-1.
81. Curtis H.A. *A New Approach to the Design of Switching circuits*. – 1962.
82. Edwards, W., Miles, R. F., Von, W. D. *Advances in decision analysis: From foundations to applications*. Cambridge: Cambridge University Press, 2007.
83. Figueira J., Salvatore G., Ehrgott M. Multiple criteria decision analysis: state of the art surveys // *Operations Research and Management Science*. – 2005. – P. 1085.
84. Files C., Drechsler R., Perkowski M. Functional decomposition of MVL functions using multi-valued decision diagrams // *Proceedings 1997 27th International Symposium on Multiple-Valued Logic*. – 1997. P. 27-32. 10.1109/ISMVL.1997.601370.
85. Flanders N.E., Brown R.V., Andreeva Y, Larichev O. I. Justifying Public Decisions in Arctic Oil and Gas Development: American and Russian Approaches // *Arctic*. – Vol. 51, No. 3. – 1998. – P. 201–300.
86. Garnelo M., Shanahan M. Reconciling deep learning with symbolic artificial intelligence: representing objects and relations // *Current Opinion in Behavioral Sciences* 29. – 2019. – P. 17–23.
87. Girardin P., Bockstaller, C., van der Werf, Hayo. Assessment of potential impacts of agricultural practices on the environment: The AGRO*ECO method // *Environmental Impact Assessment Review*. – Volume 20, Issue 2. – 2000. – P. 227–239.
88. Harris D., Harris S. *Digital design and computer architecture*. – Morgan Kaufmann, 2010.
89. Herrera F., Herrera-Viedma E. Linguistic decision analysis: steps for solving decision problems under linguistic information // *Fuzzy Sets and Systems*. – Vol. 115. – 2000. – P. 67–82.
90. Hsu Wen-Kai Kevin, Shu Lian, Show-Hui Sheree Huang. An assessment model based on a hybrid MCDM approach for the port choice of liner carriers // *Research in transportation business and management*. – Vol. 34. – 2020. – P. 100426.
91. Javeed K., Suchetha Vijaykumar, Vijeta P. Bellubbi, P. Vishal and Meghana Shankara. Comparison of different MCDM techniques used to evaluate optimal generation // 2015

- International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT). – 2015. – P. 172–177.
92. Karp R. functional decomposition and switching circuit design // J. Soc, INDJST. APPI, MATH. – Vol. 11, No. 2, – 1963.
 93. Kearns M., Li M., Pitt L., Valiant L. On the learnability of Boolean formulae // In Proceedings of the nineteenth annual ACM symposium on Theory of computing (STOC '87). Association for Computing Machinery, New York, NY, USA, 1987. – P. 285–295.
 94. Koksalan Murat & Wallenius J. & Zionts S. Multiple criteria decision making: From early history to the 21st century. – 2011. 10.1142/8042.
 95. Korgin N., Sergeev V. Identification of Integrated Rating Mechanisms on Complete Data Sets / Proceedings of IFIP WG 5.7 International Conference "Advances in Production Management Systems" (APMS 2021) (Artificial Intelligence for Sustainable and Resilient Production Systems). – 2021. – Vol. 630. – P. 610–616.
 96. Korgin N. A., Rozhdestvenskaya S. Concordant Approach for R&D Projects' Evaluation and Ranking for Formation of Programs // 2017 IEEE 11th International Conference on Application of Information and Communication Technologies (AICT). – 2017. – P. 1–5.
 97. Kozachinskiy A. Recognizing Read-Once Functions from Depth-Three Formulas // Theory Comput Syst. – Vol. 64. – 2020. – P. 3–16.
 98. Larichev O. I., Moshkovich H. M. Verbal decision analysis for unstructured problems. Springer-Science+Business Media, B.Y, 1997. – P. 267.
 99. Larichev O., Olson D., Moshkovich H., Mechitov A. Numerical vs cardinal measurements in multiattribute decision making: how exact is enough? / Organizational behavior and human decision processes. – vol. 64, No. 1. – 1995. – P. 9–21.
 100. Larichev Oleg I., Rex V. Brown. Numerical and verbal decision analysis: comparison on practical cases // Journal of Multi-criteria Decision Analysis. – Vol. 9. – 2000. – P. 263–273.
 101. Larichev Oleg I., Artyom Asanov, Yevgeny Naryzhny. Effectiveness evaluation of expert classification methods // Eur. J. Oper. Res. – Vol. 138. – 2002. – P. 260–273.
 102. Linkov I., Moberg E., Trump B.D., Yatsalo B., Keisler, J.M. Multi-Criteria Decision Analysis: Case Studies in Engineering and the Environment (2nd ed.). CRC Press, 2020.
 103. Liu Ziquan, Yanchun Zhang Comprehensive Sustainable Assessment and Prioritization of Different Railway Projects Based on a Hybrid MCDM Model // Sustainability. – Vol. 14, no. 19: 12065. – 2022.
 104. Luba T. Decomposition of multiple-valued functions // Proceedings 25th International Symposium on Multiple-Valued Logic. – 1995. – P. 256–261.

105. Luba T., Selvaraj H. A General Approach to Boolean Function Decomposition and its Application in FPGA-Based Synthesis // *VLSI DESIGN*. – 1995. – Vol. 3, Nos. 3-4. – P. 289–300.
106. Mardani A., Jusoh A., Nor K., Khalifah Z., Zakwan N., Valipour A. Multiple criteria decision-making techniques and their applications – a review of the literature from 2000 to 2014 // *Economic Research-Ekonomska Istraživanja*. – Vol. 28. – 2015. – P. 516–571.
107. Mechitov A., Moshkovich H., Underwood, S., Taylor, R. Comparative Analysis of Academic Web Sites // *Education*. – Vol. 121. – 2001. – P. 652–662.
108. Mizumoto M., Zimmermann Hans-Jürgen Comparison of Fuzzy Reasoning Methods // *Fuzzy Sets and Systems*. – Vol. 8. – 1982. – P. 253–283. 10.1016/S0165-0114(82)80004-3.
109. Moshkovich H.M., Mechitov A.I. Verbal Decision Analysis: Foundations and Trends // *Advances in Decision Sciences*. – Vol. – 2013. – P. 697072:1-697072:9.
110. Nguyen T. Designing a MCDM Model for Selection of an Optimal ERP Software in Organization // *Systems*. – 2022.
111. Nitta T. Solving the XOR problem and the detection of symmetry using a single complex-valued neuron // *Neural Networks*. – 2003. – Vol. 16(8). – P. 1101–1105.
112. Noskova A., Alekseev A. Research of the bankruptcy prediction reliability in case to introduce a new category of the companies financial position // *Applied Mathematics and Control Sciences*. – 2020. – no. 3. – P. 105–122.
113. Okada S., Ohzeki M., Taguchi S. Efficient partition of integer optimization problems with one-hot encoding // *arXiv preprint arXiv:1906.07385*. – 2019.
114. Olson D. *Decision Aids for Selection Problems* // Springer Science & Business Media. – 1996. – P. 194.
115. Olson D. L. *Decision Aids for Selection Problems*. // *Journal of the Operational Research Society*. – 1995. – Vol. 48. – 1995. – P. 541–542.
116. Perkowski M., Luba T., Grygiel S., Burkey P., Burns H., et. all. Unified approach to functional decompositions of switching functions. – 1995.
117. Ramin Gharizadeh Beiragh & Reza Alizadeh & Saeid Shafiei Kaleibari & Fausto Cavallaro & Sarfaraz Hashemkhani Zolfani & Romualdas Bausys & Abbas Mardani, An integrated Multi-Criteria Decision Making Model for Sustainability Performance Assessment for Insurance Companies // *Sustainability, MDPI*. – 2020. – vol. 12(3). – P. 1–24.
118. Ren J, (ed.). *Waste-to-Energy: Multi-Criteria Decision Analysis and Sustainability Assessment and Ranking* // Elsevier. – 2020. – P. 406.
119. Roscher R., Bohn B., Duarte M., Garcke J. Explainable Machine Learning for Scientific Insights and Discoveries // *IEEE Access*. – 2020. – vol. 8. – P. 42200-42216.

120. Rudin C. Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead // *Nat Mach Intell.* – 2019. – Vol. 1(5). – P. 206–215.
121. Rudin C., Radin J. Why Are We Using Black Box Models in AI When We Don't Need To? A Lesson From an Explainable AI Competition // *Harvard Data Science Review.* – 2019. – Vol. 1(2).
122. Ryan M. *Deep Learning with Structured Data.* Simon and Schuster, 2020.
123. Samuel A. Some studies in machine learning using the game of checkers II: recent progress // *IBM Journal of Research and Development.* – 1967. – Vol. 11. – P. 601–617.
124. Scholten L., Maurer M., Lienert J. Comparing multi-criteria decision analysis and integrated assessment to support long-term water supply planning // *PLoS ONE.* – 2017. – Vol. 12(5).
125. Seo F., Sakawa M. *Multiple Criteria Decision Analysis in Regional Planning: Concepts, Methods and Applications.* Springer, 1988. – P. 554.
126. Sergeev V.A., Korgin N.A. Identification of Integrated Rating Mechanisms As An Approach To Discrete Data Analysis // *IFAC-PapersOnLine.* – 2021. – Vol. 54, Issue 13. – P. 134–139.
127. Tjoa E., Guan C. A Survey on Explainable Artificial Intelligence (XAI): Toward Medical XAI // *IEEE transactions on neural networks and learning systems.* – 2021. – vol. 32,11. – P. 4793-4813.
128. Tonekaboni S., Joshi S., McCradden M., Goldenberg A., What Clinicians Want: Contextualizing Explainable Machine Learning for Clinical End Use // *ArXiv abs/1905.05134.* – 2019.
129. Usenyuk-Kravchuk S., Garin N., Trofimenko A., Kukanov D. Arctic design: revisiting traditional fur clothing within the daily routine of reindeer nomads // *Heliyon.* – Vol. 6, issue 2, e03355.
130. Valiant L. A theory of the learnable // *Communications of the ACM.* – 1984. – Vol. 27, issue 11. – P. 1134–1142.
131. Valiant L. *Probably Approximately Correct: Nature's Algorithms for Learning and Prospering in a Complex World.* Basic Books. New York, 2013.
132. Weistroffer H., Li Y. Multiple Criteria Decision Analysis Software // Greco S., Ehrgott M., Figueira J. (eds) *Multiple Criteria Decision Analysis. International Series in Operations Research & Management Science.* – Springer, New York, 2016. – vol. 233.
133. Yi Pingtao, Weiwei Li, Danning Zhang. Assessment of City Sustainability Using MCDM with Interdependent Criteria Weight // *Sustainability.* – 2019. – Vol. 11, no. 6. – P. 1632.
134. Yu L. et al.: Missing Data Preprocessing in Credit Classification: One-Hot Encoding or Imputation? *Emerging Markets Finance and Trade.* – 2020. – P. 1–11.
135. Yves C., Hammer P. *Boolean Functions - Theory, Algorithms, and Applications* // *Encyclopedia of mathematics and its applications.* – 2011.

136. Zadeh, L. A. Outline of a New Approach to the Analysis of Complex Systems and Decision Processes // *IEEE Trans. Syst. Man Cybern.* – 1973. – Vol. 3. – P. 28–44.
137. Zavadskas E. K., Turskis Z., Kildienė S. State of art surveys of overviews on MCDM/MADM methods // *Technological and Economic Development of Economy.* – 2014. – Vol. 20(1). – P. 165-179.
138. Zhang X., Xu Z. *Hesitant Fuzzy Methods for Multiple Criteria Decision Analysis.* Springer International Publishing Switzerland, 2017.
139. Zopounidis, C., Doumpos M. Multicriteria classification and sorting methods: A literature review // *Eur. J. Oper. Res.* – 2002. – Vol. 138. – P. 229–246.
140. Zupan B., Bohanec M., Bratko I., Demsar, J. *Machine Learning by Function Decomposition.* International Conference on Machine Learning. – 1997.
141. Zupan B., Bohanec M., Demsar J., Bratko I. Feature transformation by function decomposition // *IEEE Intelligent Systems & Their Applications.* – 1997. – Vol. 13(2). – P. 38–43.
142. Zupan B., Bratko I., Bohanec M., Demsar J. *Function Decomposition in Machine Learning* // ACAI 99, LNAI 2049. – Springer-Verlag Berlin Heidelberg, 2001. – P. 71–101.
143. Zupan B., Dzeroski S. Acquiring background knowledge for machine learning using function decomposition: a case study in rheumatology // *Artificial Intelligence in Medicine.* – 1998, – Vol. 14. – P. 101–117.

Приложение А

Таблица с данными неполного обучающего набора для иллюстрации работы алгоритмов анализа и синтеза МКО

№	П1.4	П1.3	П1.2	П1.1	F
8	3	2	1	1	1
12	2	3	1	1	2
27	2	1	2	1	1
28	3	1	2	1	1
30	5	1	2	1	1
45	5	4	2	1	2
46	1	5	2	1	1
70	5	4	3	1	3
71	1	5	3	1	1
82	2	2	4	1	2
102	2	1	5	1	2
113	3	3	5	1	3
121	1	5	5	1	1
123	3	5	5	1	3
125	5	5	5	1	4
132	2	2	1	2	2
140	5	3	1	2	2
181	1	2	3	2	1
192	2	4	3	2	2
207	2	2	4	2	2
209	4	2	4	2	2
215	5	3	4	2	3
220	5	4	4	2	4
240	5	3	5	2	3
242	2	4	5	2	2
251	1	1	1	3	1
269	4	4	1	3	3
271	1	5	1	3	1

Продолжение таблицы с данными неполного обучающего набора

№	П1.4	П1.3	П1.2	П1.1	F
273	3	5	1	3	3
276	1	1	2	3	1
283	3	2	2	3	2
294	4	4	2	3	4
310	5	2	3	3	2
311	1	3	3	3	1
322	2	5	3	3	2
333	3	2	4	3	2
335	5	2	4	3	2
337	2	3	4	3	2
349	4	5	4	3	4
352	2	1	5	3	2
365	5	3	5	3	4
367	2	4	5	3	2
368	3	4	5	3	3
387	2	3	1	4	2
389	4	3	1	4	2
408	3	2	2	4	2
409	4	2	2	4	2
416	1	4	2	4	1
417	2	4	2	4	2
430	5	1	3	4	2
432	2	2	3	4	2
449	4	5	3	4	4
455	5	1	4	4	2
470	5	4	4	4	4
481	1	2	5	4	1
486	1	3	5	4	1
507	2	2	1	5	2
516	1	4	1	5	1
518	3	4	1	5	3
542	2	4	2	5	2

Продолжение таблицы с данными неполного обучающего набора

№	П1.4	П1.3	П1.2	П1.1	F
546	1	5	2	5	1
555	5	1	3	5	2
556	1	2	3	5	1
565	5	3	3	5	4
568	3	4	3	5	3
600	5	5	4	5	4
614	4	3	5	5	4
619	4	4	5	5	4
622	2	5	5	5	2
625	5	5	5	5	5

Приложение Б

МИНОБРНАУКИ РОССИИ

**Федеральное государственное
автономное образовательное
учреждение высшего образования
«Национальный исследовательский
Томский государственный университет»
(ТГУ, НИ ТГУ)**

Ленина пр., 36, г. Томск, 634050
Тел. (3822) 52-98-52, факс (3822) 52-95-85
E-mail: rector@tsu.ru
http://www.tsu.ru
ОКПО 02069318, ОГРН 1027000853978
ИНН 7018012970, КПП 701701001

№ _____
на № _____ от _____

**Акт
о внедрении результатов диссертационной работы
Сергеева Владимира Александровича**

Комиссия в составе:

Председатель комиссии: руководитель проекта «Арктический дизайн: методы технической эстетики в освоении и развитии территорий Российского Севера», старший научный сотрудник НОЦ «Сибирский центр промышленного дизайна и прототипирования» Кравчук Светлана Геннадьевна

Члены комиссии:

Основной исполнитель проекта «Арктический дизайн: методы технической эстетики в освоении и развитии территорий Российского Севера» Гарин Николай Петрович;

основной исполнитель проекта «Арктический дизайн: методы технической эстетики в освоении и развитии территорий Российского Севера» Прокопова Софья Михайловна

Составила настоящий акт о том, что результаты диссертационного исследования Сергеева В.А. были апробированы в рамках реализации проекта «Арктический дизайн: методы технической эстетики в освоении и развитии территорий Российского Севера» (грант РФФИ № 17-78-20047-П, 2020-2022 гг.), и методы синтеза механизмов комплексного оценивания (МКО) внедрены в процедуру оценки дизайн-решений для направления «Арктический дизайн». С помощью разработанного в рамках диссертационной работы метода синтеза МКО на примере выборки студенческих дипломных и курсовых проектов по тематикам снаряжения, жилищ и транспортных средств экстремальных природно-климатических условий Российского Севера, выполненных на кафедре промышленного дизайна УрГАХУ за 40+ лет существования арктического направления, стало возможным выявить проектные характеристики, приоритизация которых приведет к успешному дизайн-проекту. В результате применения метода синтеза МКО был обнаружен так называемый «парадокс арктического дизайна»: проект, получивший в заданном наборе критериев сочетание максимально высоких и низких оценок, в итоге будет оценен ниже, чем проект, получивший сочетание средних оценок. В дальнейшем планируется проверить данный парадокс на реальных объектах и экспериментальных концептах и сопоставить результаты математического моделирования с эмпирически обнаруженными проектными стратегиями.

В связи с изложенным, считаем целесообразным использовать предложенный Сергеевым В.А. подход для синтеза МКО, для оценки сложных объектов, таких как дизайн-проекты объектов жизнеобеспечения (транспортных средств, жилищ, снаряжения) для условий Крайнего Севера и Арктики.

Настоящий акт составлен в 3 (трех) экземплярах.

Председатель комиссии:

Руководитель проекта «Арктический дизайн: методы технической эстетики в освоении и развитии территорий Российского Севера» кандидат искусствоведения, старший научный сотрудник НОЦ «Сибирский центр промышленного дизайна и прототипирования»
Кравчук С. Г. *Кравчук* «29» апреля 2023г.

Члены комиссии:

Основной исполнитель проекта «Арктический дизайн: методы технической эстетики в освоении и развитии территорий Российского Севера», кандидат искусствоведения, научный сотрудник Уральского федерального университета

Гарин Н.Н. *Гарин* «29» апреля 2023г.

Основной исполнитель проекта «Арктический дизайн: методы технической эстетики в освоении и развитии территорий Российского Севера», аспирант архитектуры Уральского государственного архитектурно-художественного университета, младший научный сотрудник Уральского федерального университета

Прокопова С.М. *Прокопова* «29» апреля 2023г.

Проректор по НИД Томского государственного университета

А.Б. Ворожцов




Приложение В

Экз № ____

УТВЕРЖДАЮ

Заместитель генерального директора ФГБУ
«НИИЦ «Институт имени Н.Е. Жуковского»
по АТВиСН, доктор технических наук, профессор


_____ Д.А. Базлев



Акт
о внедрении результатов диссертационной работы
Сергеева Владимира Александровича

Комиссия в составе:

Председатель комиссии директор проектного комплекса «Роботизированные авиационные системы», доктор технических наук, профессор Кутахов Владимир Павлович

Члены комиссии:

начальник отделения проектного комплекса «Роботизированные авиационные системы», кандидат технических наук Настас Геннадий Николаевич;

ведущий специалист проектного комплекса «Роботизированные авиационные системы» Смолин Андрей Леонидович.


Составила настоящий акт о том, что результаты диссертационного исследования Сергеева В.А. были апробированы в виде методов синтеза механизмов комплексного оценивания (МКО) внедрены в процедуру синтеза МКО для контура системы управления робототехническими комплексами (экспериментального стенда) в рамках реализации научно-исследовательской работы «ВИАС-Модель-2022».

Разработанный в рамках диссертационной работы метод синтеза МКО на основе табличных данных использовался для синтеза системы оценки степени достижения приемлемого уровня эффективности обеспечения безопасности полетов при определении уровня научно-технологического задела работ в авиастроении агрегирующей значения четырёх разнородных показателей в пятибалльной шкале в комплексный показатель, значение которого также определено в пятибалльной шкале. Для данной задачи, время синтеза МКО было уменьшено на 50% по сравнению с экспертным методом.


В связи с изложенным, считаем целесообразным использовать предложенный Сергеевым В.А. подход для синтеза МКО в системах управления, для оценки сложных объектов, таких как степень достижения приемлемого уровня эффективности обеспечения безопасности полетов в авиастроении.

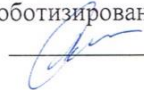
Настоящий акт составлен в 3 (трех) экземплярах.

Председатель комиссии:

Директор проектного комплекса «Роботизированные авиационные системы», доктор технических наук, профессор Кутахов В.П.  «14» апреля 2023г.

Члены комиссии:

Начальник отделения проектного комплекса «Роботизированные авиационные системы», кандидат технических наук Настас Г.Н.  «14» апреля 2023г.

Ведущий специалист проектного комплекса «Роботизированные авиационные системы» Смолин А.Л.  «14» апреля 2023г.

Приложение Г

Общество с ограниченной ответственностью
«Пермский центр поддержки принятия решений»
ОГРН 1225900019509 ИНН 5906174573
614017 Пермский край, г. Пермь, ул. Лебедева, д. 256, ЭТАЖ 3
Научные исследования и разработки в области естественных и технических наук

г. Пермь

«21» апреля 2023 г.

**Акт о внедрении результатов диссертационной работы
Сергеева Владимира Александровича**

Результаты диссертационного исследования Сергеева В.А. были апробированы в ООО «Пермский центр поддержки принятия решений» и методы синтеза механизмов комплексного оценивания внедрены для синтеза системы комплексного оценивания платежеспособности российских строительных компаний.

При апробации рассматривались двести российских предприятий строительной отрасли, из которых сто ликвидировано по причине банкротства и сто организаций, по которым дела о банкротстве не открывались, которые были условно признаны экономически благополучными. Для реализации процесса оценки финансового положения в работе использовались бухгалтерские балансы предприятий. На основе финансовых признаков и специальных критериев удельные веса 5 статей бухгалтерского баланса исследуемых компаний были закодированы в троичной системе: 0 – признак банкрота, 1 – признак компании с проблемой, 2 – признак финансово-устойчивой компании. В результате применения метода синтеза МКО было обнаружено, что невозможно синтезировать МКО, на основе матриц размерности три на три, соответствующих заданной размерности данных. Однако, с помощью предложенного метода, удалось синтезировать систему оценивания в шкале 5.

Последнее свидетельствует об эффективности разработанных в диссертации Сергеева В.А. методов синтеза механизмов комплексного оценивания, поскольку системы комплексного оценивания платежеспособности российских строительных компаний, синтезированные в ООО «Пермский центр поддержки принятия решений», имеют размерность шкал 10, что вдвое больше, чем в найденной Сергеевым В.А. системе комплексного оценивания.

Директор

Алексеев Александр Олегович

кандидат экономических наук

(08.00.13 — Математические и инструментальные методы экономики),

доцент (05.13.10 — Управление в социальных и экономических системах)



Приложение Д

3415

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2023618561

**«Сборщик полных бинарных деревьев на именованных
листьях на основе данных анализа групп
эквивалентности»**

Правообладатель: *Федеральное государственное бюджетное
учреждение науки Институт проблем управления им.
В.А. Трапезникова Российской академии наук (RU)*

Авторы: *Сергеев Владимир Александрович (RU), Коргин
Николай Андреевич (RU)*

Заявка № 2023617154

Дата поступления 14 апреля 2023 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 26 апреля 2023 г.



Руководитель Федеральной службы
по интеллектуальной собственности

Ю.С. Зубов