

**Московский государственный университет
имени М.В. Ломоносова
Механико-математический факультет**

На правах рукописи

МАХОРТОВ Сергей Дмитриевич

**Теория LP-структур для построения и исследования
моделей знаний продукционного типа**

Специальность 05.13.17 – теоретические основы информатики

АВТОРЕФЕРАТ

диссертации на соискание ученой степени
доктора физико-математических наук

Москва – 2009

Работа выполнена на кафедре математического обеспечения ЭВМ факультета прикладной математики, информатики и механики Воронежского государственного университета.

Научный консультант: доктор физико-математических наук,
профессор
Васенин Валерий Александрович

Официальные оппоненты: доктор физико-математических наук,
профессор
Чечкин Александр Витальевич

доктор физико-математических наук,
профессор
Кузнецов Сергей Олегович

доктор физико-математических наук,
профессор
Пальчунов Дмитрий Евгеньевич

Ведущая организация: Вычислительный центр
имени А.А. Дородницына РАН

Защита диссертации состоится 24 марта 2010 года в 16:45 на заседании диссертационного совета Д 501.002.16 при Московском государственном университете имени М.В. Ломоносова по адресу: 119991, Москва, ГСП-1, Ленинские горы, д.1, МГУ имени М.В. Ломоносова, механико-математический факультет, аудитория 14-08.

С диссертацией можно ознакомиться в библиотеке механико-математического факультета МГУ (главное здание, 14 этаж).

Автореферат разослан « ____ » _____ 2010 г.

Ученый секретарь диссертационного совета,
доктор физико-математических наук

Корнев А.А.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Тематика и актуальность исследования. Современное состояние информационных технологий характеризуется многоуровневой структурой – технологии, технологии для технологий и так далее. Соответственно и процесс программирования давно поддерживается собственными технологическими средствами, которые, в свою очередь, реализуются в виде специальных программных систем (см. монографии С.Р.Палмера–Дж.М.Фелсинга, А.Л.Фуксмана и др.). Объемы и сложность решаемых на данном направлении задач быстро растут, и естественной в этой связи выглядит потребность в автоматизации и переносе как можно большей части процесса производства программ на ЭВМ. Однако, чем сложнее процесс, тем выше ответственность разработчиков, риски ошибок, следующих за ними неблагоприятных ситуаций и различного рода потерь. Особое значение отмеченное обстоятельство имеет для информатизации объектов критически важных инфраструктур (см. монографию В.А.Васенина и соавторов), где допущенные «промахи» могут привести к чрезвычайным ситуациям, экологическим катастрофам, материальным и другим потерям государственного масштаба. В результате национально значимой становится проблема создания математической базы, с помощью которой можно было бы теоретически обосновывать корректность и надежность программного обеспечения, а также поддерживать процессы его автоматической оптимизации.

Эффективным средством формального построения и исследования программ, основанных на самых различных парадигмах, являются алгебраические системы (см. работы Р.И.Подловченко, А.В.Замулина, С.А.Нигияна–С.А.Аветисяна и других авторов). Это положение в полной мере относится и к логическому программированию. Алгебраическим методам представления знаний посвящены работы F.J.Oles, J.F.Sowa, а также монографии А.Тейза–П.Грибомона, Е.М.Бениаминова. Математическую основу для создания и исследования моделей знаний предоставляет алгебраическая логика. Ее начала были заложены в работах А.Линденбаума, А.Тарского, систематическое изложение дано в монографиях П.Халмоша и Е.Расёвой–Р.Сикорского. Теория Линденбаума–Тарского рассматривает логику нулевого порядка как универсальную алгебру, операции которой соответствуют логическим связкам пропозиционального языка. Примеры алгебраизации исчисления предикатов первого порядка представляют полиадические алгебры Халмоша и цилиндрические алгебры Хенкина–Тарского. Обзор методов алгебраизации кванторов содержится в монографии И.Немети.

Однако общая алгебраическая логика, расширяя возможности исследования самих логических теорий, существенно не облегчает их практического применения. В силу своей универсальности она не решает ряда важных частных проблем, связанных с широко распространенными на практике *логическими системами продукционного типа*. На этот факт указывается в книгах Н.Нильсона и Х.Уэно–Т.Коямы. К подобным проблемам могут быть отнесены вопросы эквивалентности, эквивалентных преобразований, верификации продукционных и подобных им систем, рассматривавшиеся частными методами в работах R.Agarwal, S.Lee–R.M.O’Keefe, N.K.Liu–T.Dillon и других исследователей. Обзоры подходов к верификации знаний содержатся в работах U.Gupta, а также Г.В.Рыбиной–В.В.Смирнова. Перечисленные вопросы играют существенную роль при создании и исследовании формальных

методов работы со знаниями, а также определяют принципы построения программных средств автоматизации управления знаниями.

Особое место в ряду названных проблем занимает минимизация, поскольку в любой парадигме программирования действует золотое правило – избегать неоправданного дублирования кода или данных. В общей математической логике минимальная система аксиом называется базисом. Проблемы существования базисов допустимых правил для различных логик рассматривались В.В.Рыбаковым и его последователями. Однако продукционные системы имеют особенности, дающие дополнительные возможности минимизации.

Эквивалентные преобразования и минимизация множества *унифицированных* правил продукционных экспертных систем в некоторых работах изучались на основе пропозициональных хорновых функций (например, E.Boros–O.Čeprek–A.Kogan, P.L.Hammer–A.Kogan). В статье A.Ginsberg для исключения избыточности знаний используется логика первого порядка. В работе F.Glower–H.J.Greenberg рассмотрено несколько частных случаев упрощения множества правил на основе теории графов. Еще один путь минимизации продукционных систем может дать использование представления продукции сетями Петри (R.Agarwal) и решение задачи редукции сетей Петри (П.А.Анишев, G.Berthelot, L.H.Landveber–E.L.Robertson).

В перечисленных работах нет строго обоснованного алгебраического подхода, универсального в рамках широкого спектра систем продукционного типа, который мог бы быть применен для решения задач эквивалентных преобразований, верификации и минимизации. Имеющиеся на настоящее время алгебраические исследования посвящены частным случаям продукционных систем либо другим их аспектам. В качестве примера можно привести связанную с теорией очевидности (G.Shafer) алгебраическую теорию «демпстероидов» (P.Hájek–J.V.Valdes). Она предназначена для вычислений результатов вывода в продукционных системах с функциями доверия (J.Gordon–E.H.Shortliffe).

В работе J.G.Shmolze–W.Snyder продукционная система сводится к системе переписывания термов (СПТ). В семантике последней предлагается процедура обнаружения избыточных правил. Как это принято в системах переписывания термов, требуется «завершаемость» СПТ и, соответственно, исходного множества правил, иначе нет гарантии завершаемости процедуры. Этот интересный метод не охватывает продукционные системы, множества правил которых содержат циклы.

Возможности алгебраического исследования продукционно-логических систем содержит теория ультраоператоров А.В.Чечкина. В приложении к математической логике она предлагает рассматривать импликации в виде отдельного соответствия. Однако в печати не представлены исследования ультраоператоров, связанные со свойством транзитивности логического вывода.

Интересная алгебраическая модель, позволяющая формализовать широкий круг логических задач, предложена Б.А.Куликом. Введенная им алгебра кортежей представляет собой еще одну алгебраическую интерпретацию математической логики. В монографии Б.А.Кулика описываются также возможности применения изложенной теории к моделированию экспертных систем. Не вдаваясь в подробности, отметим, что опубликованные для алгебры кортежей результаты не связаны с решением перечисленных выше задач для продукционных систем.

Выше наряду с продукционными системами недаром были упомянуты также «подобные им» системы. Многие модели в информатике имеют продукционный характер, а структуры представления информации часто являются иерархическими.

В первую очередь в контексте настоящей работы имеются в виду производственные экспертные системы. Они остаются актуальными, что подтверждается значительным числом публикаций последних лет, посвященных как их теоретическим исследованиям, так и решению прикладных задач. База знаний распространенного класса производственных систем представляет собой совокупность правил вида «если ..., то ...», где в предпосылке и заключении фигурируют множества так называемых фактов. Эти множества независимо от правил также образуют иерархию как элементы булеана – множества подмножеств. Правила расширенной производственной системы в предпосылке и заключении могут содержать пропозициональные формулы (см. R.Davis–J.King). Такие формулы кроме правил связаны еще и иерархией в рамках алгебры Линденбаума–Тарского. В подобном виде хранятся математические знания – теоремы записываются в виде «дано ..., требуется доказать ...», где предпосылка и заключение являются интерпретациями формул исчисления предикатов.

Широко применяемые в теории программирования условные системы переписывания термов (N.Dershowitz, J.W.Klop, С.Г.Воробьев) также основываются на правилах производственного вида, связывающих пары элементов из некоторой иерархии термов. Существуют и другие области информатики, на первый взгляд далекие от производств, но интерпретируемые ими. В частности, элементы иерархии типов объектно–ориентированной программной системы можно рассматривать как множество, на котором задано производственное отношение агрегирования. Даже в императивных программах просматриваются производственные связи между состояниями данных до и после выполнения каждой операции.

С учетом изложенного выше можно констатировать, что актуальной научной проблемой является создание алгебраической теории, которая бы

- адекватно отражала вторичные производственные связи в различных иерархических системах широкого спектра применения;
- обосновывала автоматизированные формальные исследования таких систем в плане их эквивалентных преобразований, верификации и оптимизации;
- допускала эффективную компьютерную реализацию.

Основная цель диссертации – разработка такой теории, а также описание и демонстрация возможностей ее применения на примерах из различных прикладных областей.

Использованные в работе **методы исследований** основаны на теориях множеств, решеток, бинарных отношений, универсальных алгебр, алгебраической логики, теории графов. При описании приложений применяются методы анализа алгоритмов, теории и технологий программирования. Используются методы обобщенных функций, функционального анализа, псевдодифференциальных операторов.

Научная новизна диссертации заключается в следующих положениях.

- Предложен новый подход для автоматизированной разработки и исследования систем производственного типа, выраженный в создании основанной на решетках алгебраической теории LP-структур (Lattice-Production Structure). Предполагается, что информация о некоторой предметной области может быть формально представлена в виде решетки. Описание методов использования решеток для представления знаний можно найти в работе F.J.Oles, монографиях J.F.Sowa, А.Тейза–П.Грибомона и др. Основная идея теории LP-структур состоит в моделировании производственных связей (совокупности правил) дополнительным бинарным отношением с заданными свойствами (рефлексивность, транзитивность и некоторые другие свойства, зависящие от конкретной модели). При этом определяющий решетку частичный порядок

отражает универсальные тавтологии и является фиксированным. Второе отношение порождается логическими связями конкретной предметной области и может подвергаться эквивалентным преобразованиям.

- Впервые введено и обосновано понятие эквивалентности продукционно-логических систем на основе их логического замыкания. Доказаны возможности автоматических локально-эквивалентных преобразований LP-структур и соответственно моделируемых ими продукционных систем.

- Введено понятие продукционно-логического уравнения и обоснован способ его решения, что в применении соответствует *полному* обратному выводу. Концепция уравнений может быть также применена для верификации знаний. Интересные классы логических уравнений рассматривались в работах А.Д.Закревского, С.Н.Васильева, В.И.Левина, однако представленные в них уравнения имеют отличную от систем продукций природу. На нечетких бинарных отношениях основаны реляционные уравнения, рассматривавшиеся в работах В. De Vaets и других авторов. Основные трудности исследования здесь порождаются нечеткостью отношений, и процесс решения соответствует лишь единственному шагу вывода.

- Доказано существование и получен эффективный способ построения логической редукции LP-структур. В приложениях он означает минимизацию соответствующих баз знаний, то есть построение эквивалентной продукционной системы с минимальным набором правил.

- Новым является распространение единого алгебраического подхода на достаточно широкий спектр различных систем: стандартные и расширенные продукционные экспертные системы; формальные системы математических знаний; условные системы переписывания термов; иерархии типов в объектно-ориентированном программировании. В частности, новым является введенный и использованный в диссертации тезис о продукционной семантике иерархии типов с отношением агрегирования. В результате на базе LP-структур обоснованы автоматизированные решения некоторых важных задач верификации типов и рефакторинга. Показана возможность применения продукционно-логических структур к новым исследованиям свойств императивных алгоритмов.

- В качестве примера для иллюстрации приложения LP-структур первого порядка в диссертации изложены элементы теории неклассических псевдодифференциальных операторов. Они содержат новые результаты автора в соответствующей области.

- Сформулирована новая концепция трехмерной структуры расширяемой программной системы, которая в дополнение к актуальной ранее двумерной модели (А.Л.Фуксман, М.М.Горбунов-Посадов) содержит набор взаимосвязанных уровней программирования от системного до пользовательского, завершающийся на верхнем уровне продукционной экспертной системой.

- Разработаны и реализованы оригинальные эффективные методы компьютерного представления основанных на решетках алгебраических структур.

- Предложены и реализованы новые методы обратного вывода и верификации для систем продукционного типа, базирующиеся на решении логических уравнений. Концепция LP-вывода направлена на минимизацию количества запросов к внешнему источнику. Запросы по возможности отправляются только для тех фактов, которые необходимы при выводе. Отрицательный ответ на единственный запрос исключает все последующие запросы об элементах целого множества. Кроме того, при LP-выводе предпочтение отдается тестированию множества фактов минимальной мощности.

Данные идеи не пересекаются с известными методами быстрого вывода, а дополняют их. Во-первых, алгоритмы RETE (C.L.Forgy) и TREAT (D.Miranker), базирующиеся на специальных сетевых представлениях множеств правил, изначально разрабатывались для стратегии *прямого* вывода и использовались в продукционных системах с прямым выводом (например, OPS5, CLIPS). Алгоритм LEAPS (D.Miranker, D.Brant) осуществляет компиляцию в язык С множества правил той же продукционной системы OPS5. В последующих исследованиях наиболее популярный алгоритм RETE адаптировался для смешанного (двунаправленного) вывода (Y.H.Lee–S.I.Yoo, P.V.Homeier–T.C.Le). Изложенная в настоящей работе концепция уравнений предназначена для исключительно *обратного* вывода и не требует для своей реализации громоздких динамических структур, свойственных указанным выше алгоритмам, в случаях, когда нет никакой потребности в прямом (и соответственно смешанном) выводе. Во-вторых, ничто не мешает адаптировать имеющиеся быстрые алгоритмы обратного вывода для нахождения рассмотренных в диссертации решений продукционно-логических уравнений. Этот подход может оказаться интересным направлением развития результатов настоящей работы.

- Реализована интегрированная среда разработки и выполнения продукционно-логических систем, обладающая новыми возможностями исследования и оптимизации баз знаний.

Новая теория LP-структур занимает собственную нишу, однако при этом она соприкасается с некоторыми другими исследованиями.

Отметим в частности, что бирешетки (M.Ginsberg) также предполагают наличие двух отношений на общем множестве, однако в прочих аспектах теория LP-структур имеет с ними мало общего, как с точки зрения формального определения, так и возможных применений. В ряде работ (см. статью J.Czelakowski и библиографию в ней) рассматриваются общетеоретические вопросы о свойствах бинарных отношений на частично упорядоченных множествах (в том числе и решетках), такие как монотонность, неподвижные (рефлексивные) точки, рекурсии. Однако эти исследования не учитывают специфики моделей продукционных систем.

Задача логического вывода на LP-структурах перекликается с проблемой нахождения функциональных зависимостей в реляционных базах данных (см., например, монографию Г.Г.Молины–Д.Ульмана–Д.Уидом). При выводе функциональных зависимостей в базе данных используются дедуктивные правила Армстронга, применяемые к элементам булеана. Однако в этой теории из «решеточных» операций используется лишь операция объединения, а набор правил Армстронга существенно беднее множества правил вывода в LP-структурах. Вполне возможно, что исследование реляционных баз данных может стать одной из новых областей применения теории LP-структур. Имеются также определенные перспективы использования подобных LP-структурам алгебраических систем для моделирования некоторых классов полуструктурированных данных (см., например, работы В.А.Васенина, С.А.Афонины) с целью исследования и оптимизации запросов.

Близким к теории LP-структур может считаться FCA – формальный концептуальный анализ (R.Wille–B.Ganter, С.О.Кузнецов), имеющий широкую область применения в исследованиях двумерных данных с семантикой «объекты–признаки». Он также основан на решетках и рассматривает бинарные отношения между множествами. Однако LP-структуры и FCA имеют существенные отличия. Как видно из публикаций, общих приложений у этих двух теорий практически нет, несмотря на иногда похожие формулировки решаемых в их рамках задач. Например, минимальный

базис импликаций в FCA переключается с логической редукцией LP-структуры. С помощью этих подходов ставятся и решаются задачи, соответствующие различным моделям в информатике. В частности, данный факт подтверждают представленные далее возможности применения LP-структур в объектно-ориентированном программировании.

Теоретическая и практическая значимость. Работа носит в основном теоретический характер. Ее положения представляют собой научно обоснованные технологические решения для построения нового класса алгебраических структур, позволяющих формулировать и успешно решать задачи исследования и оптимизации логических систем продукционного типа. Формулируя специальные свойства бинарных отношений на LP-структурах, можно аналогичными методами получать адекватные алгебраические модели различных предметных областей применения информатики, в том числе и таких, которые являются национально значимыми и остались за рамками исследований настоящей работы.

Ценность диссертационной работы дополняется возможностью практического применения установленных в ней теоретических результатов. Каждая из представленных в диссертации возможностей применения LP-структур может быть доведена до практической реализации, а именно – эффективного решения задач автоматизации преобразований, верификации и минимизации продукционно-логических систем. Такой вывод относится к стандартным, бесконечным и расширенным продукционным экспертным системам, системам компьютерной алгебры, автоматического доказательства теорем, системам автоматизированного рефакторинга, системам автоматизации программирования и другим им подобным.

Практическая значимость работы подтверждается описанной в заключительной главе компьютерной реализацией стандартной теории LP-структур в виде объектно-ориентированного класса LPStructure. Этот класс был применен при разработке интегрированной среды логического программирования LPExpert. В результате получен пакет программ с новыми эффективными возможностями создания и исследования продукционных экспертных систем. Ее преимущества продемонстрированы экспериментально на конкретных примерах. Еще одним подтверждением применимости теории LP-структур является описанная в работе формализация математических знаний LP-структурой первого порядка.

Достоверность представленных результатов обеспечивается строгими математическими формулировками и доказательствами, а также приведенными результатами экспериментов.

Апробация. Результаты диссертации докладывались на следующих научных конференциях и семинарах: совместном заседании семинара им. И.Г. Петровского и Московского математического общества (Москва, МГУ имени М.В. Ломоносова, 1986); IX конференции молодых ученых и специалистов УДН им. П. Лумумбы (Москва, 1986); XIII Всесоюзной школе по теории операторов в функциональных пространствах (Куйбышев, 1988); XIV Всесоюзной школе по теории операторов в функциональных пространствах (Новгород, 1989); XV Всесоюзной школе по теории операторов в функциональных пространствах (Ульяновск, 1990); международном семинаре «Дифференциальные уравнения и их приложения» (Самара, 1995); международной конференции «Образование, наука, производство и управление в XXI веке» (С.Оскол, 2004); 7-й международной конференции «Информатика: проблемы, методология, технологии» (Воронеж, ВГУ, февраль 2007); международной школе-семинаре «Современные проблемы механики и прикладной математики» (Воронеж,

ВГУ, сентябрь 2007); семинаре «Теоретические проблемы программирования» кафедры математической кибернетики МГУ имени М.В. Ломоносова, рук. Р.И. Подловченко, В.А.Захаров (ноябрь 2007); объединенном семинаре по компьютерной алгебре ф-та ВМК и НИИ ядерной физики МГУ имени М.В. Ломоносова, рук. С.А. Абрамов (январь 2008, март 2009); научно-исследовательском семинаре по автоматизации программирования кафедры системного программирования МГУ имени М.В. Ломоносова, рук. М.Р. Шура-Бура (февраль 2008); XLIV Всероссийской конференции по проблемам математики, информатики, физики в РУДН (апрель 2008); на XII Международной научно-практической конференции-выставке «Актуальные проблемы информатики и информационных технологий» (Тамбов, сентябрь 2008); IX Всероссийском симпозиуме по прикладной и промышленной математике (Кисловодск, май 2008; Волжский, октябрь 2008); X Всероссийском симпозиуме по прикладной и промышленной математике (Санкт-Петербург, май 2009); Общемосковском научном семинаре «Проблемы искусственного интеллекта» (июнь 2009); международной конференции «Актуальные проблемы прикладной математики, информатики и механики» (Воронеж, ВГУ, июнь 2009); международной конференции Mathematical Modeling and Computational Physics (Dubna, July 7–11, 2009); международной конференции «Мальцевские чтения» (Новосибирск, ИМ СО РАН, август 2009); научном семинаре отдела систем математического обеспечения ВЦ имени А.А. Дородницына РАН, рук. В.А. Серебряков (Москва, октябрь 2009); научном семинаре «Проблемы современных вычислительных систем» механико-математического факультета МГУ имени М.В. Ломоносова, рук. В.А. Васенин (октябрь 2009); третьей Всероссийской научной конференции «Методы и средства обработки информации» (Москва, МГУ имени М.В. Ломоносова, октябрь 2009); научном семинаре «Теоретическое программирование» Института систем информатики им. А.П. Ершова СО РАН, рук. Н.В. Шилов (октябрь 2009); второй Всероссийской конференции с международным участием «Знания – Онтологии – Теории» (ЗОНТ-09) (Новосибирск, ИМ СО РАН, октябрь 2009); а также научных сессиях Воронежского государственного университета (1987–2009).

Публикации. По теме диссертации опубликовано 50 научных работ, в том числе 1 монография и 19 работ, соответствующих Перечню ВАК РФ. Опубликованные работы отражают содержание диссертации.

Личный вклад автора. В диссертации изложены результаты, полученные лично автором, включая исследование проблематики, постановки задач, методы их решения, алгоритмы и программные реализации. Из совместно опубликованных работ в диссертацию вошли только результаты автора.

Структура и объем работы. Диссертация состоит из введения, шести глав, заключения, двух приложений и списка литературы. Общий объем диссертации 552с., список литературы содержит 219 наименований.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Поскольку в автореферате приводится лишь часть доказанных в работе теорем, непрерывность нумерации формулируемых здесь математических утверждений может не соблюдаться.

Во **введении** содержатся общий обзор тематики исследований и ее ретроспектива, показана актуальность работы, представлены цели и задачи исследований, описаны

методы решения задач, сформулированы полученные в работе новые научные результаты, показано научное и практическое значение работы.

Глава 1 носит вводный характер. В ней перечисляются задачи из различных областей информатики, описание которых может быть сведено к системам продукционного типа, моделируемым LP-структурами. Одна из целей данной главы – сделать более ясными как происхождение теории LP-структур, так и общие возможности ее применения.

Начальный раздел главы посвящен обсуждению новой концепции трехмерной структуры расширяемой программной системы. Эта концепция развивает двумерную модель, описанную в работах А.Л.Фуксмана и М.М.Горбунова-Посадова. Согласно двумерной модели, каждое расширение функциональности программы можно рассматривать как добавление определенного компонента (вертикального слоя), относящегося к некоторым ранее сформированным образованиям (горизонтальным слоям). Изъятие вертикального слоя не приводит к «катастрофическим» последствиям для программы, а лишь обедняет ее функциональность. Каждый же горизонтальный слой программы является ее неотъемлемой частью.

Анализ особенностей современных программных систем приводит к идее о целесообразности ввода в модель описания структуры программ ее третьего измерения, содержащего параллельные уровни разработки. Согласно данной концепции, программная система разрабатывается сразу на нескольких взаимосвязанных уровнях, от нижнего системного до верхнего пользовательского. При этом самый верхний уровень разработки содержит интеллектуальную часть для поддержки пользовательского программирования («настройка» программы), которая может быть представлена в виде специальной продукционной системы. Изложенная концепция изначально стала одной из отправных посылок для введения LP-структур.

Далее в главе 1 вводится основная терминология, связанная с бинарными отношениями, решетками, LP-структурами и продукционными системами.

Отношение R на произвольном множестве F называется рефлексивным, если для любого $a \in F$ справедливо $(a, a) \in R$; транзитивным, если для любых $a, b, c \in F$ из $(a, b), (b, c) \in R$ следует $(a, c) \in R$. Существует замыкание R^* произвольного отношения R относительно свойств рефлексивности и транзитивности (рефлексивно-транзитивное замыкание). Пара элементов $a, b \in F$ называется транзитивной в R , если $(a, b) \in R_1^*$, где R_1^* – транзитивное замыкание отношения $R_1 = R \setminus \{(a, b)\}$.

Обратная задача – нахождение транзитивной редукции: ищется минимальное отношение R' такое, что его транзитивное замыкание совпадает с транзитивным замыканием исходного R . Здесь минимальность подразумевается в смысле частично упорядоченных множеств, то есть различаются понятия минимального элемента (для него нет меньшего элемента) и наименьшего элемента (он меньше всех).

Решеткой \mathbb{F} называется множество с частичным порядком \leq («не больше», «содержится»), на котором для любой пары элементов $a, b \in \mathbb{F}$ определены операции \wedge («пересечение») и \vee («объединение»). Элемент $a \wedge b$ – точная нижняя грань элементов a, b в смысле отношения \leq , $a \vee b$ – точная верхняя грань a, b . Пример решетки – множество $\lambda(F)$ всех конечных подмножеств некоторого универсума F .

Решетка \mathbb{F} называется

- ограниченной, если она содержит общие нижнюю и верхнюю грани, а именно – такие два элемента O, I , что $O \leq a \leq I$ для любого $a \in \mathbb{F}$;
- полной, если каждое ее подмножество имеет в \mathbb{F} обе точные грани.

Пример полной ограниченной решетки – булеан 2^F . В полной решетке для любого подмножества элементов (конечного или бесконечного) можно определить (многоместные) пересечение и объединение. Для таких операций используются обозначения \bigwedge и \bigvee (с соответствующими индексами, если они необходимы).

Решетка называется дистрибутивной, если в ней при любых a, b, c выполняются равенства $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$; $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$. Дополнением элемента a в ограниченной решетке \mathbb{F} называют элемент $a' \in \mathbb{F}$ такой, что $a \wedge a' = O$ и $a \vee a' = I$. Решетка, в которой любой элемент имеет дополнение, называется решеткой с дополнениями. Дистрибутивная решетка с дополнениями называется булевой.

Под LP-структурой будем подразумевать алгебраическую систему, представляющую решетку, на которой задано бинарное отношение, обладающее продукционно-логическими свойствами. К ним относятся рефлексивность, транзитивность и другие свойства, определяемые конкретной моделью.

Одно из таких свойств – дистрибутивность. Неформально оно означает возможность логического вывода по частям и объединения его результатов на основе «решеточных» операций \wedge и \vee . Заданное на абстрактной решетке отношение R называется: \wedge -дистрибутивным, если из $(a, b_1), (a, b_2) \in R$ следует $(a, b_1 \wedge b_2) \in R$, \vee -дистрибутивным, если из $(a_1, b), (a_2, b) \in R$ следует $(a_1 \vee a_2, b) \in R$. Отношение называется дистрибутивным при наличии обоих указанных свойств. На полной решетке (полная) дистрибутивность отношения определяется аналогично, однако с использованием бесконечноместных операций пересечения и объединения.

В некоторых разделах диссертации в качестве основы LP-структур рассматриваются решетки с семантикой подмножеств. Соответственно вместо символов \leq, \geq, \wedge и \vee используются знаки теоретико-множественных операций $\subseteq, \supseteq, \cap$ и \cup , а элементы решетки обозначаются большими буквами. В этих случаях будем иметь в виду лишь второе из двух указанных выше свойств. В такой нотации \cup -дистрибутивность трактуется в следующем смысле: из $(A, B_1), (A, B_2) \in R$ следует $(A, B_1 \cup B_2) \in R$. Это свойство будем называть дистрибутивностью.

Подобно транзитивному замыканию и редукции отношения на обычном множестве, в теории LP-структур решаются более сложные задачи нахождения логического замыкания и логической редукции отношений на иерархических множествах – решетках. Логическим замыканием заданного на решетке отношения R называется наименьшее продукционно-логическое отношение, содержащее R . Два отношения R_1, R_2 эквивалентны, если их логические замыкания совпадают (обозначается $R_1 \sim R_2$). Эквивалентным преобразованием отношения называется замена его подмножества, приводящая к эквивалентному отношению. Логической редукцией отношения называется минимальное эквивалентное ему отношение.

Последующие разделы главы посвящены возможным применениям LP-структур к системам продукционного типа. В связи с этим вводится также основная терминология, связанная с экспертными продукционными системами. Указанные системы манипулируют множествами фактов и правил (продукций). Факт представляет собой единицу декларативной информации – некоторое суждение о внешнем мире или состоянии системы. Стандартным представлением факта является триплет вида «объект.атрибут = значение» (например, «термометр.температура = высокая»). В некоторых работах (например, книге В.Sowyer–D.Foster) триплет редуцируется к паре «параметр = значение». Это означает, что объект и атрибут

интегрируются в единый параметр. В данной связи «термометр.температура» и «термометр.изготовитель» считаются разными параметрами. В некоторых разделах данной работы пойдём дальше, интегрируя в факт параметр и его значение, считая факт независимым элементом общего множества фактов. Упрощение делается там, где необходимо больше сосредоточиться на основных идеях применения LP-структур. В дальнейшем можно реализовать и более сложную конструкцию фактов, скорректировав соответствующим образом описание LP-структуры.

Производственная система содержит так называемую *рабочую память*. Это некоторое подмножество фактов, которые на текущий момент времени считаются выполненными.

Правило (продукция) состоит из предпосылки и заключения. *Предпосылка* обычно представляет собой выражение над фактами (например, их конъюнкцию или дизъюнкцию). Предпосылка может быть выполненной (истинной) или невыполненной (ложной) при текущем состоянии рабочей памяти. Если предпосылка верна, то правило может быть применено. *Заключение* – это действие, которое можно осуществить, если верна предпосылка (например, добавить к рабочей памяти некоторый новый факт). *Применение правила* состоит в выполнении действия заключения. В контексте настоящей работы рассматриваются приложения, в которых заключение, как и предпосылка, является выражением над фактами, и соответствующее заключению действие интерпретируется как «считать истинным». Таким образом, в данном случае применение правила означает некоторую модификацию рабочей памяти. Обычно – это запись в нее тех фактов, справедливость которых вытекает из истинности выражения в заключении правила. Совокупность правил называется *базой знаний*.

Прямой выводом называется процесс циклического применения правил к содержимому рабочей памяти (его исходное состояние задано в начале работы), и соответственно получение в результате новых фактов, которые считаются справедливыми. Прямой вывод может производиться до тех пор, когда получение новых фактов станет невозможным. *Обратный вывод* – противоположный процесс. В нем по некоторому набору результирующих фактов – *гипотезе*, путем анализа правил в направлении от заключения к предпосылке, подтверждается или опровергается справедливость гипотезы при заданном исходном содержимом рабочей памяти. В процессе обратного вывода содержимое рабочей памяти также меняется.

Далее в главе 1 в качестве возможных приложений LP-структур рассматриваются несколько видов систем производственного типа, различающихся используемой формой правила, точнее – выражений для его предпосылки и заключения. В каждом случае производственную систему можно формально представить LP-структурой на основе решетки, элементами которой являются предпосылки и заключения правил.

В *стандартной* производственной системе правила в качестве предпосылки и заключения могут содержать наборы элементарных фактов ($\{a\}$, $\{a,b\}$, $\{a,b,c\}$, ...). Общий вид продукции в данном случае таков: $\{a_1, \dots, a_n\} \rightarrow \{b_1, \dots, b_m\}$, где a_i и b_j – факты. Смысл подобного правила состоит в следующем: если верны все $a_i, i=1, \dots, n$, то верны и все $b_j, j=1, \dots, m$. В этой модели основой LP-структуры является решетка конечных подмножеств $\mathbb{F} = \lambda(F)$, где F – исходное множество атомарных фактов.

Теоретически возможны производственные системы с бесконечными множествами правил. Они могут быть описаны правилами с использованием в них переменных, имеющих формально бесконечные области определения. Эквивалентные

преобразования таких систем приводят к правилам с бесконечными предпосылками и заключениями. Стандартная продукционная система с бесконечными правилами моделируется на основе булеана $\mathbb{F} = 2^F$ – полной ограниченной решетки.

В правилах *расширенной* продукционной системы могут присутствовать пропозициональные формулы (без импликаций, которые можно исключить с использованием дизъюнкции и отрицания). Данную систему можно назвать продукционной системой нулевого порядка. Соответствующую ей LP-структуру можно описать на основе булевой решетки Линденбаума–Тарского, элементами которой являются формулы пропозиционального исчисления. Логическим операциям конъюнкции, дизъюнкции и отрицания будут соответствовать определенные на булевой решетке алгебраические операции \wedge , \vee и $'$. Для любых двух формул $a, b \in \mathbb{F}$ справедливо $a \leq b$, если из истинности формулы a следует истинность формулы b .

Продукционная система *первого порядка* в качестве элементов правил допускает формулы исчисления предикатов. Такая система может формально представлять математические знания. Для построения соответствующей ей LP-структуры требуется алгебраическое описание логических кванторов, которое дополнило бы булеву решетку. Можно воспользоваться алгебраизацией кванторов из монографии Е.Расевой и Р.Сикорского. Согласно этой работе, кванторы общности и существования соответственно моделируются в общем случае бесконечноместными операциями пересечения и объединения на полной булевой решетке.

Еще одним направлением применения LP-структур являются некоторые проблемы эквациональных теорий и систем переписывания термов (СПТ). Эти системы служат математической основой многих исследований в различных областях теории программирования. Важными задачами, связанными с СПТ, являются эквивалентные преобразования и оптимизация их множеств правил. В то время как для обычных СПТ подобные проблемы уже решались в ряде работ (например, У.Тоуама), для условных СПТ они еще остаются открытыми. При определении СПТ отправной точкой является эквациональная теория, множество правил которой состоит из равенств. Поскольку обычно именно эквациональная теория является критерием эквивалентности систем переписывания, исследование в этом плане условных СПТ можно начать с рассмотрения эквивалентности условных эквациональных теорий.

Эквациональная теория может содержать набор позитивно-условных правил вида $s_1 = t_1, \dots, s_n = t_n : s = t$. Смысл такого правила в следующем: если имеют место все равенства термов $s_i = t_i, i = 1, \dots, n$, то выполнено и $s = t$. Если вместо термов рассматривать независимые элементы, то упрощение множества правил может быть сведено к логической редукции LP-структуры, которая соответствует стандартной продукционной системе. Этот же метод применим и в случае равенств между термами, однако оптимизация при этом может оказаться лишь частичной.

Таким образом, можно ввести новую LP-структуру – алгебраическую модель условной эквациональной теории. Множество правил можно задать как бинарное отношение на решетке – конечных наборах равенств вида $\{s_i = t_i\}$. На этой решетке необходимо ввести дополнительные операции, отражающие использования функциональных символов и подстановок в термах. Такая модель даст теоретическую основу для автоматического преобразования и минимизации множества правил условной эквациональной теории.

Перечисленные выше возможности применения LP-структур ограничены системами с *монотонным* выводом. Однако, существуют практические задачи, модели

которых подобным свойством не обладают. Одна из таких основана на использовании решетки типов. Структура типов объектно-ориентированной системы образует математическую решетку. При этом частичный порядок \leq задается следующим образом: если тип b является наследником a , то $b \leq a$. Для двух типов объединением считается их ближайший общий тип-предок, пересечением – ближайший общий тип-потомок. Для ограниченности полученной решетки можно добавить к ней два специальных элемента: I – универсальный тип (общий предок, имеется во многих современных системах) и O – фиктивный потомок всех типов.

На такой решетке рассматривается отношение R , соответствующее агрегации типов: если объект типа a в качестве атрибута содержится в типе b , то $(b, a) \in R$. Оба отношения (\leq и R) имеют общую семантику: в каждом случае, $b \leq a$ или $(b, a) \in R$, тип b получает возможности типа a в виде доступа к его атрибутам. Семантически ясно, что данное общее отношение «обладания набором возможностей» обязано быть рефлексивным и транзитивным. В диссертации показано, что это продукционно-логическое отношение еще обладает ограниченным свойством дистрибутивности. Оно содержит семантику автоматического решения задачи «поднятия» общих атрибутов по иерархии типов – одной из задач рефакторинга в объектно-ориентированном программировании (ООП).

Построенная таким образом LP-структура позволит производить автоматизированные исследования иерархий типов, включая эквивалентные преобразования, верификацию и минимизацию. Она может служить основой для практической реализации или модернизации типов.

К немоноктонным продукционным системам в диссертации сводится еще одна предметная область – анализ свойств и преобразования императивных алгоритмов. Во время их работы информация не только накапливается, но и замещается. Используемые в программе значения переменных можно считать фактами базы знаний. Инициализация переменной может рассматриваться как добавление нового факта к базе знаний. Присвоение же переменной нового значения можно интерпретировать как модификацию имеющегося факта.

Классическое понятие решетки оказывается недостаточно выразительным для моделирования немоноктонного продукционно-логического вывода. В связи с этим обстоятельством будут введены некоммутативные решетки как обобщение классических решеток. В новых решетках операция объединения выполняется с частичным замещением одного из операндов. Для построения соответствующих LP-структур рассматриваются логические бинарные отношения на этих решетках, а также описываются возможности применения этой модели к формальному исследованию логических свойств императивных алгоритмов.

Глава 2 содержит описание общей теории LP-структур. Теория названа общей, поскольку результаты последующих глав работы основываются на тех же положениях, развивая их в том или ином специальном направлении.

Предварительно в **разделе 2.1** формулируется частная конечная теоретико-множественная модель продукционной системы, без использования аппарата решеток. Вводится ряд математических понятий, базовым среди которых является минимальное порождающее множество. Затем показывается, как эта теория может быть применена для исследования свойств продукционных систем и разработки связанных с ними алгоритмов. Использование предлагаемого подхода позволяет:

- математически обосновывать возможности эквивалентных преобразований баз знаний;

- оптимизировать и исследовать алгоритмы обратного вывода;
- строго формулировать критерии корректности баз знаний, а также разрабатывать методы их верификации.

Обобщение этой модели и связанных с ней результатов приводит в дальнейшем к теории LP-структур, которая излагается в разделах 2.2–2.6. В главе 2 в качестве основы LP-структур рассматриваются решетки с семантикой множества $\lambda(F)$. В разделе 2.2 вводятся еще некоторые термины, которые понадобятся в дальнейшем изложении, а также доказываются некоторые вспомогательные утверждения из общей теории решеток и отношений.

Точкой ограниченной (снизу) решетки \mathbb{F} называется минимальный элемент ее подмножества $\mathbb{F} \setminus \{O\}$. Точки обычно обозначаются маленькими буквами. Решетка называется точечной, если каждый ее элемент является объединением точек. Например, в булеане точками являются все подмножества, состоящие ровно из одного элемента универсума. Для точки a элемента A иногда используется обозначение $a \in A$ (наряду с $a \subseteq A$). В точечной решетке, если она по определению не является полной, каждый элемент полагается состоящим из конечного числа точек, поскольку в такой решетке бесконечноместные операции не определены.

В разделе 2.3 вводится понятие LP-структуры, которое обобщает теоретико-множественную модель продукционной системы раздела 2.1 и выводит ее на более абстрактный уровень. Это достигается использованием решетки конечных множеств в качестве основы алгебраической системы. На решетке вводятся бинарные отношения, содержащие семантику продукционно-логического вывода. Доказывается существование логического замыкания произвольного отношения, что позволяет определить понятие эквивалентного отношения. Согласно рассматриваемой в данной главе модели, отношение на решетке называется продукционно-логическим, если оно содержит \supseteq , дистрибутивно и транзитивно.

Теорема 2.3.1. Для произвольного отношения R на решетке \mathbb{F} существует логическое замыкание \xrightarrow{R} .

Теорема 2.3.1 позволяет ввести понятие эквивалентных отношений, что служит основой формальных преобразований и оптимизации продукционных баз знаний.

Раздел 2.4 посвящен обоснованию эквивалентных преобразований LP-структур.

Теорема 2.4.1. Пусть R_1, R_2, R_3, R_4 – отношения на \mathbb{F} . Если при этом $R_1 \sim R_2$ и $R_3 \sim R_4$, то $R_1 \cup R_3 \sim R_2 \cup R_4$.

Она обосновывает принцип локальности эквивалентных преобразований: заменяя некоторое подмножество данного отношения на эквивалентное множество, мы получаем эквивалентное общее отношение.

Теорема 2.4.2. Если в отношении R на решетке \mathbb{F} каждую пару вида (A, B) , где $B = \bigcup B_i$ – конечное объединение элементов $B_i \in \mathbb{F}$ ($1 \leq i \leq n$), заменить парами $\{(A, B_1), \dots, (A, B_n)\}$, то полученное отношение R' будет эквивалентно исходному R .

Теоремы 2.4.1–2.4.2 могут быть применены для приведения отношений к каноническому виду. Некоторыми исследователями (например, R. Agarwal) рассматриваются множества хорновых правил. В данной диссертации такой системе соответствует каноническое отношение. Отношение R на точечной решетке \mathbb{F} называется каноническим, если оно задано множеством пар вида (A, a) , где $A \in \mathbb{F}$, a – точка \mathbb{F} . Из теоремы 2.4.2 следует, что для любого отношения на точечной решетке существует эквивалентное ему каноническое отношение.

Раздел 2.5 посвящен минимизации LP-структур. Вначале рассматривается вопрос о том, является ли логическое замыкание отношения R транзитивным замыканием некоторого другого отношения $\tilde{R} \supseteq R$. Положительный ответ на него полезен для разработки алгоритмов построения логического замыкания и редукции. Он позволяет свести исследование вопросов о нахождении логического замыкания и редукции отношений к рассмотрению соответствующих свойств транзитивных отношений.

Для отношения R на решетке \mathbb{F} вводится отношение \tilde{R} , построенное последовательным выполнением следующих шагов:

- добавить к R все пары вида (A, A) , где $A \in \mathbb{F}$ (рефлексивные пары), и обозначить новое отношение R_1 ;
- добавить к R_1 всевозможные пары (A, B) с элементами вида $A = \bigcup A_i$, $B = \bigcup B_i$, где все (A_i, B_i) ($i = 1, \dots, n$) принадлежат R_1 ;
- объединить полученное отношение с отношением включения \supseteq .

Теорема 2.5.1. Для отношения R на решетке \mathbb{F} логическое замыкание \xrightarrow{R} представляет собой транзитивное замыкание \tilde{R}^* соответствующего отношения \tilde{R} .

Далее рассматривается вопрос о существовании логической редукции отношений. Для отношения R вводится отношение \underline{R} , построенное последовательным выполнением шагов, обратных построению \tilde{R} , а именно:

- исключить из R содержащиеся в нем пары вида $A \supset B$ и обозначить новое отношение R_{-1} ;
- исключить из R_{-1} все пары (A, B) с элементами вида $A = \bigcup A_i$, $B = \bigcup B_i$, где (A_i, B_i) ($i = 1, \dots, n$) принадлежат R_{-1} и не совпадают с (A, B) ;
- исключить из полученного отношения все рефлексивные пары.

Аналогичный подход к построению *транзитивной редукции* («удалить все транзитивные пары») был бы ошибочным. Причина в том, что в некоторых ситуациях (наличие циклов) транзитивная редукция не единственна, и одновременное удаление всех транзитивных пар может привести к потере связей. Однако, поскольку решетка ациклична, удаление отмеченных выше «объединенных» пар (A, B) приводит к одному и тому же результату независимо от порядка удаления. В связи с данным обстоятельством можно говорить об одновременном удалении всех таких пар.

Следующая теорема указывает достаточное условие существования и способ построения логической редукции данного отношения.

Теорема 2.5.2. Пусть для отношения R построено соответствующее отношение \tilde{R} . Тогда, если для \tilde{R} существует транзитивная редукция R^0 , то соответствующее ей \underline{R}^0 представляет собой логическую редукцию исходного отношения R .

Заметим, что требование существования транзитивной редукции отношения \tilde{R} может оказаться избыточным для существования логической редукции исходного отношения R . При конечном R из него всегда можно удалить «лишние» пары, получив в результате логическую редукцию. Однако если при этом решетка бесконечна и имеет соответствующую структуру, то, объединяя R с отношением \supseteq , можно построить не имеющее транзитивной редукции отношение \tilde{R} . Таким образом, возникает вопрос об усилении теоремы 2.5.2. Одно из возможных направлений такого усиления – использование вместо \supseteq отношения вида \supseteq_R , содержащего лишь необходимое для получения логической редукции R подмножество \supseteq . В этом случае утверждения и доказательства настоящей главы будут более громоздкими, однако

принципиальных новшеств не принесут. По этой причине данные идеи оставлены лишь в качестве рекомендации для конкретных приложений.

Следует также отметить, что на практике при построении \tilde{R} не обязательно физически добавлять указанные пары. Достаточно реализовать эффективный механизм проверки того, относится ли заданная пара к теоретически добавляемым.

В разделе 2.6 вводится и изучается связанный с LP-структурами специальный класс уравнений. Рассматриваются вопросы о разрешимости и количестве решений этих уравнений, а также методы их решения.

Пусть дано некоторое отношение R на решетке \mathbb{F} и имеет место $A \xrightarrow{R} B$. Тогда B называется образом A , а A – прообразом B при отношении \xrightarrow{R} . Каждый элемент из \mathbb{F} может иметь много образов и прообразов. Более того, в данном случае в силу определения логического отношения любой $B_1 \subset B$ является образом A и каждый $A_1 \supset A$ является прообразом B . По этой причине при изучении образов и прообразов логических отношений необходимо уточнение рассматриваемых понятий. Для элемента $B \in \mathbb{F}$ минимальным прообразом при отношении \xrightarrow{R} называется такой элемент $A \in \mathbb{F}$, что $A \xrightarrow{R} B$ и A является минимальным, то есть не содержит никакого другого $A_1 \in \mathbb{F}$, для которого $A_1 \xrightarrow{R} B$. В оставшейся части данного раздела рассматриваются лишь точечные решетки.

Определение 2.6.1. Точка x решетки \mathbb{F} называется начальной при отношении R , если в R нет ни одной такой пары (A, B) , что x содержится в B и не содержится в A . Элемент X точечной решетки \mathbb{F} называется начальным, если все его точки являются начальными (при отношении R). Подмножество $\mathbb{F}_0(R)$ (обозначается \mathbb{F}_0 , если это не вызывает неоднозначностей) точечной решетки \mathbb{F} , состоящее из всех начальных элементов \mathbb{F} , называется начальным множеством решетки \mathbb{F} (при отношении R).

Рассматривается уравнение

$$R(X) = B, \tag{2.6.1}$$

где $B \in \mathbb{F}$ – заданный элемент, $X \in \mathbb{F}$ – неизвестный.

Определение 2.6.2. Частным решением X уравнения (2.6.1) называется любой минимальный прообраз элемента B , содержащийся в \mathbb{F}_0 . Приближенным (частным) решением X уравнения (2.6.1) называется любой прообраз элемента B , содержащийся в \mathbb{F}_0 . Общим решением уравнения (2.6.1) называется совокупность всех его частных решений $\{X_s\}, s \in S$.

Исследуется вопрос о том, как меняются решения уравнений вида (2.6.1) при объединении их правых частей. С этой целью вводится также уравнение

$$R(X) = B_1 \cup B_2 \tag{2.6.2}$$

Теорема 2.6.1. Пусть $\{X_s\}, s \in S_1$ – общее решение уравнения вида (2.6.1) с правой частью B_1 , а $\{Y_p\}, p \in S_2$ – общее решение уравнения того же вида с правой частью B_2 . Тогда общее решение уравнения (2.6.2) представляет собой множество всех элементов вида $X_s \cup Y_p$, из которого исключены элементы, содержащие другие элементы этого же множества.

Рассматриваются методы решения уравнений. В оставшейся части раздела 2.6 предполагается, что R является каноническим отношением на точечной решетке \mathbb{F} , не содержащим пар отношения \supseteq , а правая часть B уравнения (2.6.1) представляет собой конечное объединение точек. Вводится понятие расслоения $\{R_t \mid t \in T\}$ отношения R . Оно упрощает изучение отношения \xrightarrow{R} и построение алгоритмов решения

уравнений. С этой целью вначале R разбивается на непересекающиеся подмножества $R^p, p \in P$, образованные парами вида (A, x_p) с общим элементом x_p .

Определение 2.6.3. Слоем R_i в отношении R называется подмножество R , образованное упорядоченными парами, взятыми по одной из каждого непустого $R^p, p \in P$. Решение X уравнения (2.6.1) порождается слоем R_i , если $X \xrightarrow{R_i} B$.

Теорема 2.6.2. Для нахождения общего решения уравнения (2.6.1) достаточно найти частное решение X_i в каждом слое R_i , если оно существует, после чего из полученного множества решений исключить элементы, содержащие другие элементы этого же множества.

Следствие 2.6.3. Количество частных решений уравнения (2.6.1) оценивается сверху выражением $N = \prod N_p$, где N_p – кардинальное число подмножества пар отношения R с правой частью x_p .

Теорема 2.6.2 позволяет свести решение уравнения (2.6.1) к нахождению частного решения уравнения

$$R_i(X) = B, \quad (2.6.3)$$

где B – неначальный элемент решетки \mathbb{F} , R_i – произвольный слой в R .

Исследования раздела 2.6 показывают, что для решения уравнения (2.6.3) достаточно решить уравнение с каждой точкой элемента B в качестве правой части. В связи с данным обстоятельством рассматривается задача нахождения частного решения следующего уравнения:

$$R_i(X) = b, \quad (2.6.4)$$

где b – неначальная точка решетки \mathbb{F} , R_i – произвольный слой в R .

Доказывается, что она эквивалентна задаче на ориентированном графе перечисления входных вершин, из которых достижима данная вершина. Каждой точке решетки \mathbb{F} , участвующей в отношении R , сопоставляется вершина графа. Далее для каждой пары (A, a) рассматриваемого слоя R_i проводятся дуги, ведущие из каждой вершины, соответствующей точке A , в вершину, соответствующую данной a . Для краткости точки \mathbb{F} и соответствующие им вершины в графе отождествляются.

В полученном графе G_{R_i} выбирается вершина b . Рассматривается подграф $G_{R_i, b} \subseteq G_{R_i}$, содержащий все вершины, из которых достижима вершина b (включая саму b), и все дуги, соединяющие такие вершины.

Теорема 2.6.3. Уравнение (2.6.4) имеет не более одного решения. Если граф $G_{R_i, b}$ не содержит циклов, то единственное решение уравнения состоит из всех точек, соответствующих входным вершинам графа. Если $G_{R_i, b}$ содержит хотя бы один цикл, то уравнение решений не имеет.

Данная теорема завершает обоснование пошагового процесса решения исходного логического уравнения (2.6.1).

В главе 3 определяются LP-структуры на решетках, обладающих свойством полноты. Вводятся отношения, содержащие семантику продукционно-логического вывода со свойством нетеровости, то есть транзитивной завершаемости. Заметим, что условие нетеровости LP-структуры слабее аналогичного свойства систем переписывания термов, так как в отличие от них допускает циклы. Приводится обоснование необходимости введения в модель ограничений, соответствующих этому свойству. Данные исследования обобщают материал главы 2. Они предназначены для

моделирования продукционных систем, правила которых могут содержать бесконечные множества фактов. Некоторые определения по форме выглядят подобными соответствующим определениям главы 2, однако теперь их содержание связано с *полными* решетками, и это обстоятельство усложняет как смысл данных понятий, так и доказательство их свойств. Получены результаты, соответствующие изложенным в разделах 2.3–2.6, но относящиеся к нетеровым продукционно-логическим отношениям на полных решетках.

Важной общей особенностью глав 2–3 является нерекурсивный подход при определении понятий продукционно-логических связей и доказательстве их свойств. Этот подход усложняет математические выкладки, но предоставляет больше возможностей для использования параллельных вычислений. Параллельным методам в продукционных системах посвящены, например, работы A.Gupta, D.E.Neiman, J.G.Schmolze. Описанные в них методы могут быть применены на более абстрактном уровне при компьютерной реализации LP-структур.

Глава 4 диссертации посвящена развитию теории LP-структур применительно к некоторым расширенным моделям продукционных систем. Ввиду усложнения механизмов вывода при определении соответствующих понятий и доказательстве их свойств в данной главе используется рекурсивный подход, аналогичный принятому подходу в пропозициональном исчислении (например, С.Клини, Ч.Чень–Р.Ли).

В **разделе 4.1** исследуется LP-структура, логика которой расширена до набора связей пропозиционального языка. Отсюда происходит название – «LP-структура нулевого порядка». В качестве ее основы используется булева решетка. Изучены следующие основные вопросы: существование LP-замыкания, его архитектура, эквивалентные преобразования. Доказана теорема о существовании логической редукции произвольного отношения и описан процесс ее построения, один из этапов которого связан с нахождением транзитивной редукции отношения.

В **разделе 4.2** определяются LP-структуры на полных булевых решетках. В данных алгебраических системах определены бесконечно-местные операции пересечения и объединения, которые для модели продукционной логики реализуют кванторы общности и существования. Как и в главе 3, переход к полным решеткам привел к необходимости введения дополнительного ограничения модели в виде свойства нетеровости.

Рассматриваются вопросы, связанные с существованием логического замыкания, его архитектурой, эквивалентными преобразованиями. Доказаны новые теоремы об архитектуре логического замыкания и о существовании логической редукции рассматриваемой LP-структуры. Полученные результаты полностью соответствуют разделу 4.1, однако относятся к нетеровым продукционно-логическим отношениям на полных булевых решетках.

Раздел 4.3 посвящен модели эквациональных LP-структур. Она возникает в качестве алгебраической интерпретации условной эквациональной теории и соответствующей условной системы переписывания термов. Определяется основанная на решетках модель условной эквациональной теории, учитывающая связи между термами, которые обусловлены применениями функций и подстановок. Исходная теория состоит из условных соотношений $s_1 = t_1, \dots, s_n = t_n : u_1 = v_1, \dots, u_m = v_m$ («если имеют место равенства термов $s_i = t_i, i = 1, \dots, n$, то выполнены и все $u_j = v_j, j = 1, \dots, m$ »). Такие соотношения называются (условными) эквациональными правилами. В предложенной модели условные эквациональные правила реализуются бинарным отношением R на решетке, порожденной множествами равенств $\{s_i = t_i\}$.

В начале раздела приводятся связанные с терминами базовые определения. Пусть Σ – алфавит, представляющий объединение непересекающихся множеств: V – множество переменных; $\Sigma_n, n=0,1,\dots$ – множества n -арных функций. Стандартным образом определяется множество термов $T(\Sigma)$ (J.W. Klop). Отображение $\sigma:V \rightarrow T(\Sigma)$ называется подстановкой. Это понятие распространяется на все $t \in T(\Sigma)$.

Эквациональной теорией называется пара (Σ, E) , где Σ – алфавит, состоящий из счетного множества переменных и непустого множества функциональных символов, а $E \subseteq T(\Sigma) \times T(\Sigma)$ – множество равенств вида $s=t$ ($s, t \in T(\Sigma)$). Для данного множества равенств E рассматривается множество конечных подмножеств $\lambda(E)$. В нем заданы отношения включения \subseteq, \supseteq , а также «решеточные» операции \cap и \cup – теоретико-множественные пересечение и объединение. Кроме них задаются еще две группы операций, связанных соответственно с функциями и подстановками термов:

- 1) если $a = \{s_i = t_i \mid i=1, \dots, n\}$ и $f \in \Sigma_n$, то $f(a) = \{f(s_1, \dots, s_n) = f(t_1, \dots, t_n)\}$;
- 2) если $a = \{s_j = t_j \mid j=1, \dots, m\}$, то $\sigma(a) = \{\sigma(s_j) = \sigma(t_j) \mid j=1, \dots, m\}$ для любой подстановки σ .

Определение 4.3.1. Пусть задана эквациональная теория (Σ, E) . Эквациональной решеткой называется решетка, полученная пополнением $\lambda(E)$ относительно определенных выше операций 1)–2).

Как говорилось выше, в данной модели рассматривается условная эквациональная теория с правилами вида $s_1 = t_1, \dots, s_n = t_n : u_1 = v_1, \dots, u_m = v_m$. Таким образом, предпосылка и заключение правила являются элементами эквациональной решетки.

По аналогии с известными работами (J.W.Klop), вводятся аксиомы и правила условной эквациональной дедукции. Аксиомы порождаются правилами вывода равенств: $a : f(a)$ для любых $a = \{s_1 = t_1, \dots, s_n = t_n\}$ и $f \in \Sigma_n$; $a : \sigma(a)$ для любых $a \in \mathbb{F}$ и подстановки σ . Такие условные правила можно назвать тавтологиями. Еще одной очевидной тавтологией является правило $a : b, a, b \in \mathbb{F}$ при $a \supseteq b$.

Правила вывода в условной эквациональной логике таковы:

- $a : b \vdash \sigma(a) : \sigma(b)$ ($a, b \in \mathbb{F}$) для любой подстановки σ (следует J.W.Klop);
- $a : b, a : c \vdash a : b \cup c$ ($a, b, c \in \mathbb{F}$) (возможность вывода по частям);
- $a : b, b : c \vdash a : c$ ($a, b, c \in \mathbb{F}$) (транзитивность).

Далее вводится понятие продукционно-логического отношения, которое соответствует условной эквациональной логике. Во-первых, логическое отношение R должно содержать все тавтологии. Для тавтологий вводится общее обозначение: $a \succ b$, если $a \supseteq b$, $b = \sigma(a)$ или $b = f(a)$. Таким образом, для логического отношения R справедливо $\succ \subseteq R$. Другие свойства логического отношения вытекают из правил дедукции. В частности, отношение R называется применимым, если для любой подстановки σ из $(a, b) \in R$ следует $(\sigma(a), \sigma(b)) \in R$.

Бинарное отношение на эквациональной решетке называется продукционно-логическим, если оно содержит тавтологии, а также является применимым, \cup -дистрибутивным и транзитивным.

Кроме построенной модели, материалы раздела содержат исследование стандартного для настоящей диссертации круга вопросов: существование логического замыкания, возможности локально-эквивалентных преобразований, исследование архитектуры логического замыкания, существование и построение логической редукции бинарного отношения.

Применение рассмотренных выше моделей LP-структур ограничено системами с *монотонным* выводом, то есть таких систем продукционного типа, логический вывод в которых означает монотонное накопление знаний. Однако существуют системы, предполагающие не только накопление, но и модификацию знаний.

В **главе 5** рассматриваются две задачи, исследование которых может быть осуществлено на основе LP-структур с немонотонным выводом.

Модель **раздела 5.1** содержит семантику продукционно-логического вывода на иерархии типов в объектно-ориентированной системе с дополнительным отношением. Анализируется стандартный набор свойств таких структур, а именно – замкнутость, эквивалентные преобразования, существование логической редукции. Показано, как изложенная теория может быть применена для верификации и автоматической оптимизации иерархий типов, в частности, при рефакторинге – модернизации устаревшего кода. Одним из важных направлений в этом плане является устранение дублирования кода «подъемом» общих атрибутов по иерархии типов. Такая задача решается автоматически при построении логической редукции LP-структуры.

Решение родственных задач методами FCA предлагается в статье R.Godin–P.Valtchev. В этой работе элементам множества классов требуется в определенном смысле оптимальным образом назначить наборы атрибутов – элементов другого независимого множества. В соответствии с выбранными назначениями формируется иерархия классов. В постановке раздела 5.1 атрибуты сами относятся к исследуемой иерархии классов (типов). Это обстоятельство усложняет решение задачи и вряд ли оставляет возможность применения FCA.

Рассматривается некоторая иерархия типов \mathbb{F} в смысле ООП. Между парами типов могут существовать два вида связей – наследование (тип наследует атрибуты типа-предка) и агрегация (тип содержит в качестве атрибута представителя другого типа). На множестве \mathbb{F} вводится отношение частичного порядка, а именно – если тип b является наследником a (соответственно a – предком b), то $b \leq a$. Для любых $a, b \in \mathbb{F}$ определены две операции: пересечение $a \wedge b$ – наибольший общий потомок и объединение $a \vee b$ – наименьший общий предок a, b . Для ограниченности \mathbb{F} добавляются два специальных элемента: I – универсальный тип (общий предок, имеется во многих современных системах) и O – фиктивный потомок всех типов.

На решетке \mathbb{F} рассматривается второе (соответствующее агрегации) отношение R : если объект типа a в качестве атрибута содержится в типе b , то $(b, a) \in R$. Оба отношения (\leq и R) имеют общую семантику. Она состоит в том, что в каждом из случаев, $b \leq a$ или $(b, a) \in R$, тип b получает возможности типа a в виде доступа к его атрибутам. Это общее отношение \xleftarrow{R} обязано быть рефлексивным и транзитивным.

Рассматривается еще одно свойство введенных отношений. Пусть для $a, b_1, b_2 \in \mathbb{F}$ справедливо $b_1 \leq a$, $b_2 \leq a$. Тогда в соответствии с определением решетки имеем $b_1 \vee b_2 \leq a$. Это естественное для отношения \leq свойство называется \vee -дистрибутивностью.

Пусть теперь $b_1 \xleftarrow{R} a$ и $b_2 \xleftarrow{R} a$, то есть каждый тип b_1 и b_2 обладает возможностями типа a . Тогда, если предположить \vee -дистрибутивность отношения \xleftarrow{R} , справедливо $b_1 \vee b_2 \xleftarrow{R} a$. Этот факт означает, что тип $b_1 \vee b_2$ также обладает возможностями a . С точки зрения проектирования типов это не обязательно. Однако в случае, если более одного наследника (b_1 и b_2) содержат одинаковые атрибуты, по принципам рефакторинга нужно «поднять» общие атрибуты, то есть поместить один

атрибут в общий тип-предок $b_1 \vee b_2$. В результате каждый b_1 и b_2 получит возможности a в порядке наследования. В данном случае \vee -дистрибутивность отношения \leftarrow^R определяет решение одной из задач рефакторинга – устранение дублирования кода.

Отношение \leftarrow^R , обладая свойством транзитивности вне зависимости от контекста, не может аналогично во всех ситуациях удовлетворять \vee -дистрибутивности, иначе это приведет к некорректным результатам. Для иллюстрации можно привести пример $b \leftarrow^R a$ и $a \leftarrow^R a$. При дистрибутивности \leftarrow^R выполнялось бы $b \vee a \leftarrow^R a$. По идеологии ООП тип $b \vee a$ не имеет права знать о своих наследниках. По этой причине $b \vee a$, являясь общим предком типов a и b , может обладать возможностями типа a лишь в случае, если он совпадает с a ($b \leq a$). В остальных вариантах соотношение $b \vee a \leftarrow^R a$ будет некорректным.

Существуют ситуации, когда выполнение \vee -дистрибутивности отношения \leftarrow^R теоретически возможно, но нецелесообразно с точки зрения качества кода. Пусть при $b_1 \leftarrow^R a, b_2 \leftarrow^R a$ элементы $b_1 \vee b_2$ и a имеют непустое пересечение: $(b_1 \vee b_2) \wedge a = d \neq O$, причем $d < b_1 \vee b_2$ и $d < a$. Если в данном случае допустить $(b_1 \vee b_2, a) \in R$, то окажется, что тип d обладает возможностями типа a одновременно по двум линиям, а именно – как его наследник и как наследник типа $b_1 \vee b_2$, также имеющего возможности a . Другая подобная ситуация – наличие конфликта. Пусть имеет место $(b_1, a), (b_2, a), (b_3, a) \in R$, причем $b_1 \vee b_2 \neq b_2 \vee b_3$. Тогда пары $(b_1, a), (b_2, a)$ «конфликтуют» с парами $(b_2, a), (b_3, a)$. Это означает, что если в обоих случаях «поднять» атрибуты, то тип b_2 унаследует атрибут типа a от двух различных предков – $b_1 \vee b_2$ и $b_2 \vee b_3$, что также ухудшит код. Принятая в работе стратегия предполагает отказ от «поднятия» общих атрибутов при наличии подобных ситуаций (невыполнение \vee -дистрибутивности). В дальнейшем возможны другие подходы, тоньше учитывающие особенности конкретной системы.

Из приведенных соображений дается определение логического отношения на ограниченной решетке. Логическое замыкание произвольного отношения R на решетке \mathbb{F} предоставляет все такие пары (b, a) , что в типе b доступны возможности типа a . Логическая редукция строит иерархию с минимальным эквивалентным набором связей и в определенном смысле минимальным дублированием кода.

Формулируются определения и условия, связанные с ограничением свойства \vee -дистрибутивности продукционно-логических отношений. Пара (b, a) элементов ограниченной решетки \mathbb{F} называется простой, если $a \wedge b = O$ – нижняя грань \mathbb{F} .

Определение 5.1.1.1. Пусть R – отношение на ограниченной решетке \mathbb{F} . Две пары вида $(b_1, a), (b_2, a) \in R$ называются \vee -совместимыми в R , если существуют такие $c_1, c_2 \in \mathbb{F}$, что $b_1 \vee b_2 \leq c_1 \vee c_2$, причем пара $(c_1 \vee c_2, a)$ простая, а пары $(c_1, a), (c_2, a) \in R$ нетранзитивны в $R \cup \leq$. При этом набор элементов $C = (b_1, b_2, c_1, c_2, a)$ называется \vee -дистрибутивным кортежем, а $T = (c_1, c_2, a)$ – \vee -дистрибутивной тройкой (в R).

Пусть имеются \vee -дистрибутивная тройка $T = (c_1, c_2, a)$, а также тройка $T' = (c'_1, c'_2, a')$, проверяемая на аналогичное свойство. Тройку T будем называть нейтрализующей для T' (обозначается $T' \prec T$), если выполнено одно из следующих условий:

- 1) $a = a', c_1 \vee c_2 \neq c'_1 \vee c'_2$ и справедливо хотя бы одно из неравенств $c'_i < c_1 \vee c_2$;
- 2) $a < a', c_1 \vee c_2 \neq c'_1 \vee c'_2$ и выполнено хотя бы одно из неравенств $c'_i \leq c_1 \vee c_2$;

$$3) a < a', c_1 \vee c_2 = c'_1 \vee c'_2.$$

С позиции неформальной эти условия описывают ситуации, когда присутствие нейтрализующей тройки T «угрожает» свойству \vee -дистрибутивности тройки T' . Тройка $T' = (c'_1, c'_2, a')$ называется неконфликтной, если для нее не существует ни одной нейтрализующей \vee -дистрибутивной тройки.

Понятия, введенные выше для троек вида $T = (c_1, c_2, a)$ и $T' = (c'_1, c'_2, a')$, автоматически распространяются и на связанные с ними кортежи $C = (b_1, b_2, c_1, c_2, a)$ и $C' = (b'_1, b'_2, c'_1, c'_2, a')$. Две \vee -совместимые пары $(b_1, a), (b_2, a)$ называются неконфликтно \vee -совместимыми, если для них существует неконфликтный \vee -дистрибутивный кортеж (b_1, b_2, c_1, c_2, a) . Отношение R на решетке называется ограничено \vee -дистрибутивным, если для любых неконфликтно \vee -совместимых в R пар $(b_1, a), (b_2, a)$ справедливо $(b_1 \vee b_2, a) \in R$.

Отношение называется продукционно-логическим с ограничением объединений, если оно содержит \leq , транзитивно и ограничено \vee -дистрибутивно. Логическим замыканием отношения R называется наименьшее логическое отношение, содержащее R и его множество неконфликтно совместимых пар.

В новой модели рассматриваются стандартные для LP-структур вопросы: существование логического замыкания, возможности локально-эквивалентных преобразований, архитектура логического замыкания, построение логической редукции бинарного отношения. Установлены аналогичные результаты, за одним исключением. Его суть в том, что в данной модели эквивалентное преобразование части исходного множества пар не всегда приводит к эквивалентному общему отношению. Однако имеет место эквивалентность ряда преобразований.

Теорема 5.1.3.2. Пусть R – отношение на ограниченной решетке \mathbb{F} . Тогда каждая из следующих операций над R приводит к эквивалентному отношению:

- добавление или исключение пары (b, a) , если $b \leq a$;
- добавление пары $(c_1 \vee c_2, a)$, если тройка $T = (c_1, c_2, a)$ \vee -дистрибутивна и неконфликтна в R ;
- добавление или исключение пары (c, a) при наличии пар $(c, b), (b, a) \in (R \cup \leq)$, $c \neq b, b \neq a$.

Заключительный подраздел посвящен оценкам сложности построения логического замыкания и логической редукции LP-структур на решетках типов. Решетка типов обычно не является дистрибутивной, и число ее элементов не так велико по сравнению с другими моделями (например, булеаном или решеткой Линденбаума–Тарского). По этой причине в данном случае в качестве основы анализа сложности можно выбрать общее число элементов решетки, а также хранить саму решетку и бинарные отношения на ней в виде матриц смежности.

Теорема 5.1.5.1. Задачи нахождения логического замыкания и логической редукции LP-структуры на решетке типов решаются за полиномиальное время, не превышающее $O(N^6)$, где N – общее число элементов решетки.

Существуют области искусственного интеллекта, предполагающие не только накопление, но и модификацию получаемых знаний. В работе к ним добавлена еще одна – анализ свойств и преобразования императивных алгоритмов. Соответствующая LP-структура исследуется в п. 5.2. Как обобщение классических решеток определяются некоммутативные решетки. В некоммутативной решетке \mathbb{F}_X операция

объединения \cup_X выполняется с частичным замещением одного из операндов. Операции на некоммутативной решетке формализуют как накопление, так и замещение знаний. Вводятся действующие на этих решетках продукционно-логические отношения. Доказана теорема о существовании логического замыкания отношений на некоммутативных решетках.

В заключение рассматривается пример приложения теории некоммутативных LP-структур к исследованию логических свойств императивных алгоритмов – сопоставляются возможности императивного и логического программирования.

Логическая цепочка $\vec{r}_{AB} = (B_1, \dots, B_m)$ называется *полной*, если при добавлении к ней в любую позицию любого элемента $B_{m+1} \in \mathbb{F}_X$, не совпадающего с соседними элементами, она перестает быть цепочкой, логически связывающей пару (A, B) . Логическая цепочка \vec{r}_{AB} называется *терминальной*, если она не может быть продолжена: не существует такого элемента $C \in \mathbb{F}_X$, что пара (A, C) логически связана R , $C \neq B$ и цепочка \vec{r}_{AB} содержится в начале цепочки \vec{r}_{AC} .

Определение 5.2.4.1. *Логической программой* на решетке \mathbb{F}_X называется тройка $P_R = (F_0, F_{Res}, R)$, где F_0, F_{Res} – два фиксированных подмножества решетки \mathbb{F}_X , называемые соответственно множеством *начальных элементов* и множеством *результатов*, R – некоторое бинарное отношение на \mathbb{F}_X . *Выполнением* логической программы P_R с данным начальным элементом $A_0 \in F_0$ называется нахождение совокупности всех таких максимальных элементов $A_t \in F_{Res}$, что пара (A_0, A_t) логически связана полной терминальной цепочкой в R . Полученная совокупность элементов $\{A_t\}$ называется *результатами* программы P_R , соответствующими данному начальному элементу A_0 ($t \in T$). Если T состоит из единственного элемента, то результат логической программы называется *однозначным*.

Далее устанавливается связь логических программ с императивными алгоритмами. Для этого рассматривается подкласс – детерминированные логические программы на решетках. Логическая программа $P_R = (F_0, F_{Res}, R)$ на \mathbb{F}_X называется *детерминированной* при данном $A_M \in \mathbb{F}_X$, если для любого $A_0 \in F_0$ при начальном элементе $A_0 \cup_X A_M$ ее результат $A_{Res} \in F_{Res}$ однозначен и достигается единственной (с точностью до порядка элементов в подцепочках) логической цепочкой в R .

Для таких программ можно выделить еще одно подмножество решетки \mathbb{F}_X – множество *управляющих элементов* F_M , не пересекающееся с F_0 и F_{Res} . Их роль состоит в обеспечении детерминированности действий. Таковым является указанный выше управляющий начальный элемент $A_M \in F_M$. В связи с этим обстоятельством для детерминированных программ также используется обозначение $P_{R,M} = (F_M, F_0, F_{Res}, R)$.

Проводится формализация императивного алгоритма в терминах отношений на решетках. Она опирается на структурную теорему Дейкстры, согласно которой любой алгоритм может быть преобразован в эквивалентный, содержащий лишь действия, управляемые структурами трех видов: *цепочкой*, *альтернативой* и *циклом while*. В качестве (императивных) *элементарных действий* на некоммутативной решетке \mathbb{F}_X рассматриваются применения операции некоммутативного объединения. В частности, результатом действия над некоторым элементом $A \in \mathbb{F}_X$ можно считать $C = A \cup_X B$, где $B \in \mathbb{F}_X$.

Для выполнения императивной программы необходимы исходные данные – совокупность начальных элементов F_0 решетки \mathbb{F}_X . При выполнении действий (операций \cup_X) начальный элемент $A_0 \in F_0$ последовательно трансформируется в некоторый новый. На заключительном этапе получается результат программы – результирующий элемент $A_{Res} \in F_{Res} \subset \mathbb{F}_X$. Следуя изложенным соображениям, обозначим A_{Curr} переменную со значениями текущего элемента, который имеется на некотором шаге после выполнения всех предыдущих операций программы. Таким образом, до выполнения программы $A_{Curr} = A_0$, после – $A_{Curr} = A_{Res}$.

Далее необходимо формализовать понятия управляющих структур.

Императивной цепочкой длины m будем называть произвольный упорядоченный набор $I = (B_C^1, \dots, B_C^m)$ элементов решетки \mathbb{F}_X . Индекс «C» означает возможную зависимость каждого элемента цепочки от текущего значения A_{Curr} . Результатом последовательного применения императивной цепочки $I = (B_C^1, \dots, B_C^m)$ к исходному значению A_{Curr} назовем элемент $A_{Curr} \cup_X B_C^1 \cup_X \dots \cup_X B_C^m$ – новое значение A_{Curr} .

Для определения альтернативы и цикла необходимо понятие *условия*. Предполагается, что вычисление логических выражений не вызывает побочных эффектов. В рассматриваемой модели условие эквивалентно принадлежности одного из заданных элементов решетки $\{A_{Cond}^t \mid t \in T\}$ текущему элементу A_{Curr} . Символами $\{\bar{A}_{Cond}^t \mid t \in \bar{T}\}$ обозначается набор элементов, соответствующих отрицанию условия с элементами $\{A_{Cond}^t \mid t \in T\}$.

Альтернативой называется выражение вида $\text{if } \{A_{Cond}^t\} \text{ then } B_C^1 \text{ else } B_C^2$. Результат ее альтернативы – новое значение A_{Curr} , вычисляемое по формуле $A_{Curr} \cup_X B_C^1$, если при некотором $t \in T$ справедливо $A_{Cond}^t \subseteq A_{Curr}$, и $A_{Curr} \cup_X B_C^2$ – в противном случае. *Циклом* называется конструкция $\text{while } \{A_{Cond}^t\} \text{ do } B_C$. Ее выполнение заключается в рекуррентном вычислении нового значения $A_{Curr} := A_{Curr} \cup_X B_C$, пока существует такое $t \in T$, что $A_{Cond}^t \subseteq A_{Curr}$.

Введенные определения расширяются предположением о том, что управляющие конструкции могут быть вложенными. Этот факт означает, что в качестве их содержимого $(B_C^1, B_C^2, \dots, B_C^m, B_C)$ могут фигурировать не только элементы решетки, но и, в свою очередь, другие цепочка, альтернатива или цикл. Для перечисленных конструкций используется общее название – *императивная структура*.

Определение 5.2.4.2. *Императивной программой* на некоммутативной решетке \mathbb{F}_X называется тройка $P_I = (F_0, F_{Res}, I)$, где F_0, F_{Res} – два фиксированных подмножества элементов решетки \mathbb{F}_X (множества *начальных элементов* и *результатов*), а I – любая императивная структура на \mathbb{F}_X . *Выполнением* императивной программы P_I с данным *начальным элементом* $A_0 \in F_0$ называется применение конструкции I к элементу A_0 . Получаемый после этого элемент $A_{Res} \in F_{Res}$ называется *результатом* программы P_I , соответствующим данному начальному A_0 .

Сравнивая представленные определения логических и императивных программ, нетрудно заметить, что каждый из этих видов программ по исходному элементу решетки вычисляет некоторый результирующий. Таким образом, правомерна постановка вопроса о сравнении их выразительных возможностей.

Теорема 5.2.4.1. Для любой императивной программы $P_I = (F_0, F_{Res}, I)$ на решетке \mathbb{F}_X можно построить детерминированную логическую программу $P_{R,M} = (F_M, F_0, F_{Res}, R)$, которая при любом начальном элементе $A_0 \in F_0$ и результате $A_{Res} \in F_{Res}$ программы P_I выдает для своего начального элемента вида $A_0 \cup_X A_M$ тот же результат A_{Res} , сохраняя при этом порядок и содержание действий P_I .

Основная цель **главы 6** – продемонстрировать возможность применения теории LP-структур на практике. Описывается созданная автором интегрированная среда разработки продукционных экспертных систем, а также анализируются реализация и применение LP-структуры для верификации и оптимизации созданных с ее помощью баз знаний. В заключительной части главы демонстрируется применение теории LP-структур для алгебраической формализации математических знаний.

Предварительно приводятся некоторые общие принципы реализации положений п.п.2.3–2.6, не столь значимые для теоретического исследования, однако существенные на практике. Далее рассматриваются вопросы, которые призваны продемонстрировать возможности использования теоретических положений LP-структур для решения актуальных задач.

Исходя из специфики решеток, при их кодировании необходимо в первую очередь заботиться об экономии памяти. В результате выбран способ кодирования LP-структур, опирающийся на известный метод представления решеток битовыми векторами. При реализации в работе битовый вектор фактически хранится в виде набора смежных кластеров, размер которых может настраиваться исходя из потребностей пользователя.

В целях экономии памяти автором также принято решение о хранении отношений в виде динамических множеств пар, в противовес матрицам смежности. Как известно, эффективность операций доступа к множеству на практике обеспечивается возможностью его хранения в виде сбалансированного бинарного дерева с уникальными ключами в узлах. Ключ по возможности должен помещаться в слове (или двойном слове) компьютера, тогда при доступе к множеству операция сравнения ключей будет выполняться за константное время. Таким образом, реализована идея хранения пары элементов решетки как пары ссылок на соответствующие им битовые векторы. Эти ссылки содержатся в смежных частях двойного целого числа, и данное число может использоваться в качестве упомянутого выше ключа.

Для хранения *канонических* отношений на решетках при реализации ряда алгоритмов в работе применяется еще одно новое представление – в виде вектора множеств. Напомним, что каноническое отношение на точечной решетке состоит из пар «хорновского» типа. Правая часть такой пары представляет собой точку решетки, а левая – объединение точек. Для данного отношения строится специальный массив. Индексом массива служит номер точки решетки, содержимым компоненты массива – множество элементов решетки, составляющих с данной точкой пары в качестве их левых частей. Указанное представление весьма эффективно как с точки зрения использования памяти – хранятся лишь левые части пар, так и быстродействия – доступ к ним осуществляется по индексу в массиве. Этот способ применяется для реализации в LP-структуре алгоритмов обратного вывода.

Далее в главе 6 описывается интерфейс класса LPStructure, разработанного автором и реализованного на языке C++ с использованием библиотеки STL. Класс LPStructure инкапсулирует наиболее важные свойства и методы описанных в главе 2 стандартных

LP-структур, включая нахождение логической редукции и решение продукционно-логических уравнений.

В последующем подразделе обсуждаются особенности архитектуры и основные возможности пакета программ LPExpert (реализованного на Delphi), предназначенного для разработки и эксплуатации продукционных экспертных систем. Данный пакет создавался на основе идей, изложенных в книге В.Sowyer–D.Foster, с существенными их усовершенствованиями в синтаксисе правил и программной реализации. Впоследствии к нему была добавлена динамическая библиотека – интерфейс класса LPStructure, что значительно расширило возможности пакета по созданию баз знаний. К основным преимуществам данной системы логического программирования можно отнести следующие возможности:

- исчерпывающее выявление избыточных правил в базах знаний;
- проверка их непротиворечивости;
- новые методы обратного логического вывода (релевантный и кластерно-релевантный LP-выводы);
- исследование образов и прообразов подмножеств фактов.

LP-вывод осуществляется путем построения минимальных начальных прообразов значений объекта экспертизы и последующего их исследования. В построенном множестве достаточно найти тот прообраз, который содержит лишь выполненные факты, тогда сразу можно сделать заключение о соответствующем значении объекта экспертизы. В представленном подходе избран эффективный метод, который заключается в приоритетном просмотре прообразов, содержащих значения «релевантных» объектов. Таковыми в первую очередь считаются объекты, чьи значения присутствуют в максимальном количестве прообразов. Тогда единственный отрицательный ответ пользователя на заданный вопрос исключает из рассмотрения сразу большое количество прообразов. Второй фактор релевантности – присутствие объекта в прообразах с минимальной мощностью. Таким образом, предпочтение отдается решениям, которые требуют меньшего количества внешних запросов. В результате существенно снижается количество обращений за фактами к внешнему источнику. Еще одним важным свойством *релевантного* LP-вывода является получение более полных результатов по сравнению с обычным выводом при наличии противоречий в базе знаний.

Эксперименты показали, что при больших объемах баз знаний процесс одновременного построения *всех* минимальных прообразов может потребовать чрезмерного объема ресурсов компьютера. В связи с данным обстоятельством метод *релевантного* LP-вывода был модифицирован. *Кластерно-релевантный* LP-вывод предполагает последовательное построение кластеров начальных прообразов (подмножеств ограниченной мощности) с их динамическим релевантным исследованием. Модифицированный LP-вывод дает достаточно эффективные результаты для промышленных БЗ больших размеров.

В п.6.4.5 описываются функциональные возможности и интерфейс пользователя пакета LPExpert. В п.6.4.6. его возможности экспериментально демонстрируются на ряде тестовых примеров. Первый из них – база знаний «Здоровье» из книги В.Sowyer–D.Foster. Выбор данного примера сделан в силу его популярности в некоторых других исследованиях (например, работа С.G.Giraud-Carrier и Т.R Martinez). Для тестовой базы знаний получены следующие новые результаты:

- выявлено три избыточных правила (в работе С.G.Giraud-Carrier и Т.R Martinez отмечено лишь одно таковое);

- показано, что LP-вывод по сравнению с обычным обратным выводом дает аналогичные результаты, но при этом пользователю задается меньше вопросов (в тестовой экспертизе – на 20%);
- доказана противоречивость данной БЗ, что свидетельствует о нецелесообразности ее практического применения.

Далее в работе рассматриваются еще две тестовые базы знаний. Результаты проведенных экспериментов также свидетельствуют о преимуществах применения теории LP-структур для исследования, оптимизации и верификации баз знаний.

В заключительном разделе главы 6 демонстрируется еще одно подтверждение применимости теории LP-структур – алгебраическая формализация математических знаний LP-структурой первого порядка. В качестве примера выбрана разработанная автором теория весовых псевдодифференциальных операторов (ПДО). Построена LP-структура первого порядка – элементы соответствующей полной решетки Линденбаума–Тарского и пары элементов логического отношения, которые в виде записанных продукционных правил формализуют указанные математические знания.

Приложение А диссертации содержит тексты трех демонстрационных баз знаний.

В **Приложении В** приведены элементы разработанной автором теории весовых ПДО. Кроме новых результатов, относящихся к области неклассических уравнений с частными производными, оно иллюстрирует возможности применения LP-структур к моделированию математических знаний.

Рассматриваются специальные классы ПДО, не относящиеся к главному типу. Эти операторы имеют множество вырождения ненулевой меры, обусловленное наличием в их символах весового множителя α . Предлагаемый метод исследования вырождающихся ПДО основан на их сравнении с другими ПДО, построенными по введенному В.П.Глушко весовому преобразованию Фурье F_α . Для построения этой теории требуются также весовые пространства типа Соболева–Слободецкого.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ ДИССЕРТАЦИИ И ПЕРСПЕКТИВЫ

Диссертация посвящена созданию математической базы, обосновывающей корректность и надежность систем продукционного типа, а также их автоматическую оптимизацию. В работе представлена созданная автором на основе выполненных им исследований теория LP-структур, описывающая новые эффективные методы для построения широкого спектра моделей знаний продукционного типа. Теория LP-структур дает общую методологию анализа продукционных систем в плане эквивалентности, эквивалентных преобразований, верификации и оптимизации.

Рассмотрение особенностей продукционных систем привело к их новой общей алгебраической модели. Ее изучение показало, что аналогичной семантикой обладают и другие системы в информатике, которые ранее никогда не относились непосредственно к продукционным. В результате построена математическая теория, имеющая широкую область приложений в информатике.

В целях практического применения теория LP-структур развита до эффективных способов компьютерных представлений бинарных отношений на решетках. На основе этой теории разработаны новые методы обратного вывода. Представленная в диссертации программная реализация интегрированной среды разработки экспертных систем, созданные с её помощью базы знаний, результаты их верификации и оптимизации демонстрируют работоспособность теории LP-структур. Реализована

эффективная система логического программирования, обладающая новыми автоматизированными возможностями.

В кратком изложении основные теоретические результаты диссертационного исследования составляют следующие положения.

1. Введено и обосновано понятие эквивалентности продукционно-логических систем на основе их логического замыкания.

2. Доказаны возможности автоматических локально-эквивалентных преобразований систем продукционного типа.

3. Рассмотрены продукционно-логические уравнения и предложены способы их решения, что в применении соответствует полному обратному выводу.

4. Доказано существование и получен эффективный способ построения логической редукции LP-структур, что в приложениях означает минимизацию соответствующих баз знаний.

5. Исследован новый тезис о продукционной семантике иерархии типов с отношением агрегирования; обоснованы автоматизированные решения важных задач верификации типов и рефакторинга.

6. Показана возможность применения продукционно-логических структур к исследованиям новых свойств императивных алгоритмов.

7. Предложена новая концепция трехмерной архитектуры расширяемой программной системы, развивающая используемую ранее двумерную модель.

8. Разработаны и реализованы оригинальные компьютерные представления основанных на решетках алгебраических структур.

9. Предложены и реализованы эффективные методы обратного вывода и верификации для систем продукционного типа, основанные на решении логических уравнений.

10. Для иллюстрации применения LP-структур первого порядка изложены элементы теории неклассических псевдодифференциальных операторов, в свою очередь содержащие новые результаты в соответствующей области.

К числу практических результатов диссертации относятся следующие.

11. Представлена компьютерная реализация теории LP-структур (проведенные эксперименты показали ее работоспособность и эффективность).

12. Разработана и реализована интегрированная среда для продукционно-логических систем LPExpert, обладающая новыми возможностями исследования, оптимизации и верификации знаний.

13. Практическое использование теории LP-структур и ее компьютерной реализации подтверждается выполненными автором на их основе исследованиями ряда баз знаний, в том числе промышленного уровня, а также имеющимися патентами и актами внедрения.

С учетом изложенного есть основания полагать, что созданная в рамках диссертационной работы теория LP-структур вносит весомый вклад в решение крупной, национального уровня научной проблемы создания математической базы для верификации, оценки надежности и оптимизации программного обеспечения. Технологические решения на её основе обладают несомненными инновационными перспективами и практической значимостью для экономики страны.

Теория LP-структур имеет возможности дальнейшего развития, а ее применения могут охватить более широкие области информатики. В этом плане можно назвать следующие направления.

Усложнение моделей. Это, в частности, относится к LP-структурам на решетках типов. В работе принята стратегия отказа от «подъема» общего атрибута при наличии конфликта, однако возможен более тонкий учет особенностей конкретных иерархий типов. Также при использовании в данных структурах свойства \wedge -дистрибутивности окажется доступным моделирование других методов рефакторинга. Усложняя эквивалентные LP-системы, можно ввести дополнительные операции, которые позволят учитывать структуру термов в равенствах. Класс LPStructure может быть реорганизован в абстрактный класс и применен в качестве основы иерархии типов, охватывающих теорию LP-структур целиком.

Построение новых моделей. Например, функциональные зависимости в реляционных базах данных можно описать LP-структурой с продукционно-логическими свойствами отношений, основанными на правилах Армстронга. Представляются возможными построения аналогичных моделей для мультиагентных систем, некоторых классов полуструктурированных данных и так далее.

Использование эффективных алгоритмов. При компьютерной реализации LP-структур могут быть применены быстрые алгоритмы построения транзитивного замыкания и редукции отношений, а также специальные алгоритмы их обновления.

Динамические LP-структуры. Данная идея связана с возможностями динамического преобразования самой решетки, в то время как в моделях настоящей работы может изменяться лишь вторичное отношение на ней. Таким образом, например, можно более полно реализовать методы рефакторинга в иерархиях типов.

Нечеткие LP-структуры. Это направление способно существенно расширить области применения разработанных в диссертации методов.

ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

(в изданиях Перечня ВАК РФ)

1. *Махортов С. Д.* LP-структуры на решетках типов и некоторые задачи рефакторинга / С. Д. Махортов // Программирование. – 2009. – Т. 35, № 4. – С. 5–14.
2. *Махортов С. Д.* Основанный на решетках подход к исследованию и оптимизации множества правил условной системы переписывания термов / С. Д. Махортов // Интеллектуальные системы. – 2009. – Т. 13, № 2. – С. 31–37.
3. *Махортов С. Д.* Интегрированная среда логического программирования LPExpert / С. Д. Махортов // Информационные технологии. – 2009. – № 12. – С. 65–66.
4. *Махортов С. Д.* Алгебраический подход к исследованию и оптимизации баз знаний продукционного типа / С. Д. Махортов, С. Л. Подвальный // Информационные технологии. – 2008. – № 8. – С. 55–60.
В работе [4] Махортову С.Д. принадлежат постановки задач, формулировки и доказательства теорем.
5. *Глушко В. П.* Операторы неглавного типа и задача с косою производной / В. П. Глушко, С. Д. Махортов // Успехи мат. наук. – 1986. – Т. 41, № 4. – С. 202–203.
В работе [5] Махортову С.Д. принадлежат формулировки и доказательства теорем.
6. *Махортов С. Д.* Продукционная логика первого порядка и ее алгебраическая интерпретация / С. Д. Махортов // Системы управления и информационные технологии. – 2007. – № 3(29). – С. 21–26.
7. *Махортов С. Д.* Продукционно-логические отношения на полных решетках / С. Д. Махортов // Системы управления и информационные технологии. – 2008. – № 3(33). – С. 44–48.

8. *Махортов С. Д.* LP-структуры и возможности их применения для эквивалентных преобразований условных систем переписывания термов / С. Д. Махортов // *Обозрение прикладной и промышленной математики.* – 2008. – Т. 15, вып. 3. – С. 504–505.
9. *Махортов С. Д.* LP-структуры на полных решетках и возможности их применения в системах продукционного типа / С. Д. Махортов // *Обозрение прикладной и промышленной математики.* – 2008. – Т. 15, вып. 4. – С. 671–672.
10. *Махортов С. Д.* Продукционно-логические уравнения на полных решетках / С. Д. Махортов // *Обозрение прикладной и промышленной математики.* – 2009. – Т. 16, вып. 2. – С. 368–369.
11. *Махортов С. Д.* О технологии многоуровневой разработки программных систем / С. Д. Махортов // *Вестник ВГУ. Серия Физика, математика.* – Воронеж. – 2002. – № 1. – С. 159–162.
12. *Махортов С. Д.* Порождающие множества в продукционных системах / С. Д. Махортов // *Вестник ВГУ. Серия Физика, математика.* – Воронеж. – 2002. – № 2. – С. 69–76.
13. *Махортов С. Д.* Логические отношения на решетках / С. Д. Махортов // *Вестник ВГУ. Серия Физика, математика.* – Воронеж. – 2003. – № 2. – С. 203–209.
14. *Махортов С. Д.* Логические уравнения на решетках / С. Д. Махортов // *Вестник ВГУ. Серия Физика, математика.* – Воронеж. – 2004, № 2. – С. 170–178.
15. *Махортов С. Д.* Некоммутативные решетки и немонотонные логические отношения / С. Д. Махортов // *Вестник ВГУ. Серия Физика, математика.* – Воронеж. – 2006. – № 1. – С. 166–173.
16. *Махортов С. Д.* О сравнении возможностей императивного и логического программирования / С. Д. Махортов // *Вестник ВГУ. Серия Системный анализ и информационные технологии.* – Воронеж. – 2006. – № 1. – С. 78–86.
17. *Махортов С. Д.* Методы исследования и преобразования иерархий типов на основе логических структур / С. Д. Махортов // *Вестник ВГУ. Серия Системный анализ и информационные технологии.* – Воронеж. – 2006. – № 2. – С. 24–27.
18. *Махортов С. Д.* Алгебры весовых и вырождающихся псевдодифференциальных операторов / С. Д. Махортов // *Вестник ВГУ. Серия Физика, математика.* – Воронеж. – 2007. – № 1. – С. 73–80.
19. *Махортов С. Д.* Об алгебраической интерпретации продукционной логики нулевого порядка / С. Д. Махортов // *Вестник ВГУ. Серия Системный анализ и информационные технологии.* – Воронеж. – 2007. – № 1. – С. 56–63.

(монографии)

20. *Махортов С. Д.* Математические основы искусственного интеллекта: теория LP-структур для построения и исследования моделей знаний продукционного типа / С. Д. Махортов ; под ред. В. А. Васенина. – М. : МЦНМО, 2009. – 300 с.

(прочие публикации)

21. *Махортов С. Д.* Продукционно-логические уравнения на полных решетках / С. Д. Махортов // *Проблемы информатики.* – 2009. – № 2. – С. 28–38.
22. *Makhortov S.* Multi-level LP-Structures in Rewriting Systems / S. Makhortov // *Mathematical Modelling and Computational Physics (MMCP'2009): Book of Abstracts of the International Conference (Dubna, July 7–11, 2009).* – Dubna : JINR, 2009. – P. 123.

23. *Махортов С.Д.* Модели логических систем продукционного типа, основанные на решетках / С. Д. Махортов // Международная конференция «Мальцевские чтения», посвященная 100-летию со дня рождения академика А.И.Мальцева (Mal'tsev Meeting, August 24-28, 2009): Тезисы докладов. – Новосибирск, 2009. – С. 228.
24. *Махортов С. Д.* Релевантный обратный вывод и верификация логических программ на основе решения уравнений в LP-структурах / С. Д. Махортов // Методы и средства обработки информации: Третья Всероссийская научная конференция. Москва, 6-8 октября 2009 г.: Труды конференции / Под ред. Л.Н. Королева. – М. : ВМиК МГУ, 2009. – С. 143–148.
25. *Махортов С.Д.* LP-структуры представления знаний и принципы их реализации / С. Д. Махортов // Вторая Всероссийская конференция с международным участием «Знания – Онтологии – Теории» (ЗОНТ-09). Новосибирск, 22-24 октября 2009г.: Материалы конференции. – Новосибирск: Институт математики им. С.Л. Соболева СО РАН, 2009. – Т.2. – С. 11–19.
26. *Калечиц В. Е.* Система интерактивной и пакетной отладки в режиме эмуляции / В. Е. Калечиц, Н. И. Лободин, С. Д. Махортов // Математическое моделирование и программное обеспечение САПР. – Горький, 1984. – С. 42–47.
В работе [26] Махортову С.Д. принадлежат разработка алгоритмов и программная реализация.
27. *Махортов С. Д.* Об одном классе псевдодифференциальных операторов неглавного типа / С. Д. Махортов // Операторные уравнения в функциональных пространствах. – Воронеж : ВГУ, 1986. – С. 127–129.
28. *Махортов С. Д.* Разрешимость некоторых вырождающихся граничных задач для эллиптических уравнений / С. Д. Махортов // Неклассические уравнения математической физики. – Новосибирск, 1986. – С. 171–175.
29. *Махортов С. Д.* О задаче с косою производной при произвольном множестве вырождения / С. Д. Махортов // Тезисы XIII Всесоюзной школы по теории операторов в функциональных пространствах. – Куйбышев, 1988. – С. 128–129.
30. *Махортов С. Д.* Алгебра вырождающихся псевдодифференциальных операторов / С. Д. Махортов // Применение методов функционального анализа к неклассическим уравнениям математической физики. – Новосибирск, 1988. – С. 93–101.
31. *Махортов С. Д.* О разрешимости одной некоэрцитивной задачи для вырождающегося эллиптического уравнения / С. Д. Махортов // Тезисы XIV Всесоюзной школы по теории операторов в функциональных пространствах. – Новгород, 1989. – С. 71.
32. *Махортов С. Д.* О задаче с косою производной / С. Д. Махортов // Краевые задачи для неклассических уравнений математической физики. – Новосибирск, 1989. – С. 141–143.
33. *Махортов С. Д.* О фредгольмовости одного псевдодифференциального оператора неглавного типа / С. Д. Махортов // Тезисы XV Всесоюзной школы по теории операторов в функциональных пространствах. – Ульяновск, 1990. – С. 11.
34. *Махортов С. Д.* О задаче с косою производной / С. Д. Махортов // Материалы международного семинара «Дифференциальные уравнения и их приложения». – Самара, 27–30 июня 1995 г. – С. 63.
35. *Махортов С. Д.* О задаче с косою производной со специальным случаем вырождения / С. Д. Махортов // Материалы межвузовской научно-практической конференции «Актуальные проблемы борьбы с преступностью в современных условиях». – Воронеж : ВИ МВД, 2000. – С. 185.

36. *Махортов С. Д.* О теоретико-множественном подходе к формализации логического вывода / С. Д. Махортов // Вестник факультета ПММ. – Воронеж : ВГУ. – 2003. – Вып. 4. – С. 178–185.
37. *Махортов С. Д.* О редукции логических отношений на решетках / С. Д. Махортов // Вестник факультета ПММ. – Воронеж : ВГУ, 2004. – Вып. 5. – С. 172–179.
38. *Махортов С. Д.* О разрешимости логических уравнений на решетках / С. Д. Махортов // Материалы Международной конференции «Образование, наука, производство и управление в XXI веке». – Ст. Оскол, 2004. – С. 308–311.
39. *Махортов С. Д.* Алгебраическая интерпретация продукционной логики / С. Д. Махортов // Вестник факультета ПММ. – Воронеж : ВГУ. – 2006. – Вып. 6. – С. 86–98.
40. *Махортов С. Д.* Теория LP-структур и возможности ее применения в интеллектуальных системах / С. Д. Махортов // Материалы 7-й международной конференции «Информатика: проблемы, методология, технологии». – Воронеж : ВГУ, 2007. – С. 269–272.
41. *Махортов С. Д.* Алгебраический подход к исследованию и оптимизации продукционных баз знаний / С. Д. Махортов // Сб. трудов международной школы-семинара «Современные проблемы механики и прикладной математики». – Воронеж : ВГУ, 2007. – С. 237–241.
42. *Махортов С. Д.* О приложениях LP-структур в теории программирования / С. Д. Махортов // Вестник ВГУ. Серия Системный анализ и информационные технологии. – Воронеж. – 2007. – № 2. – С. 40–49.
43. *Махортов С. Д.* О логической редукции условных систем переписывания термов / С. Д. Махортов // Сб. трудов XLIV Всероссийской конференции по проблемам математики, информатики, физики. – Москва : РУДН, 2008. – С. 50–51.
44. *Махортов С. Д.* О редукции продукционно-логических отношений на полных решетках / С. Д. Махортов // Современные проблемы информатизации в анализе и синтезе технологических и программно-телекоммуникационных систем : сб. трудов / под ред. О. Я. Кравца. – Воронеж : Научная книга, 2009. – Вып. 14. – С. 322–325.
45. *Махортов С. Д.* Модель немонотонной продукционной логики для систем компьютерной алгебры / С. Д. Махортов // Вестник факультета ПММ. – Воронеж : ВГУ, 2009. – Вып. 7. – С. 127–141.
46. *Махортов С. Д.* Кластерно-релевантный обратный вывод на основе решения логических уравнений / С. Д. Махортов // Сб. трудов Международной конференции «Актуальные проблемы прикладной математики, информатики и механики». – Воронеж : ВГУ, 2009. – Ч. 2. – С. 37–41.
47. *Махортов С. Д.* LP-структуры и возможности их применения в некоторых задачах рефакторинга / С. Д. Махортов, В. А. Погореленко // Сб. трудов XLIV Всероссийской конференции по проблемам математики, информатики, физики. – Москва : РУДН, 2008. – С. 52–53.
48. *Махортов С. Д.* Решение некоторых задач рефакторинга на основе алгебраических структур / С. Д. Махортов, В. А. Погореленко // Сб. трудов XII Международной научно-практической конференции-выставки «Актуальные проблемы информатики и информационных технологий». – Тамбов : ТГУ, 2008. – С. 147–149.
49. *Махортов С. Д.* Алгебраический подход к решению некоторых задач рефакторинга / С. Д. Махортов, В. А. Погореленко // Вестник ВГУ. Серия Системный анализ и информационные технологии. – Воронеж. – 2008. – № 2. – С. 28–36.

В работах [47–49] Махортову С.Д. принадлежат постановки задач, формулировки и доказательства теорем.

50. *Морозова И. С.* Программы понижения и повышения уровня структурного описания сложной системы / И. С. Морозова, С. Д. Махортов // Математическое обеспечение ЭВМ вузов. – Воронеж : ВГУ, 1980. – С. 115–120.

В работе [50] Махортову С.Д. принадлежат разработка алгоритмов и программная реализация.