УДК 519.711.3
КП
№ДР 0112 U000205
Інв. №

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки
(ХНУРЕ)
61166, м. Харків, пр. Леніна, 14
тел. (057) 702-14-46

ЗАТВЕРДЖУЮ
проректор з наукової роботи
ХНУРЕ, проф., д.ф-м.н.

_______________М.І.Сліпченко
«___»__________2014 р.

ЗВІТ
ПРО НАУКОВО-ДОСЛІДНУ РОБОТУ

«РОЗРОБКА ТЕОРІЇ ТА ПРИНЦИПІВ ПОБУДОВИ МОЗКОПОДІБНИХ СТРУКТУР»
№264
(заключний)

Керівник НДР
пров. наук. співр.
д.т.н., проф. _______________ Шабанов-Кушнаренко С.Ю.

2014

Рукопис закінчений «___»__________2014 р.

Результати роботи розглянуті науково-методичною радою ХНУРЕ,
Протокол № __ від __________.

СПИСОК АВТОРОВ

Руководитель НИР, проф., вед.н.с., д.т.н., проф.	Шабанов-Кушнаренко С.Ю. (реферат, вступление, выводы, разд. 1-2)
Ответственный исполнитель, с.н.с., к.т.н., с.н.с.	Вечирская И.Д. (разд. 3)
м.н.с., к.т.н.	Русакова Н.Е. (подразд. 4.1-4.2)
м.н.с.	Викарчук М.М. (подразд. 4.3-4.4)

РЕФЕРАТ

Отчет по НИР: 149 с., 39 рис., 2 табл., 135 ссылок.

Объектом исследования является процесс параллельной автоматической обработки текстов естественного языка. Предметом исследования являются лингвистические структуры разной степени сложности – предложения, словосочетания, слова и их модели в виде реляционных сетей.

Целью исследования является разработка комплекса методов и моделей для формализации лингвистических структур естественного языка с помощью аппарата алгебры конечных предикатов и теории реляционных сетей для расширения возможностей интеллектуального пользовательского интерфейса, последующая реализация полученных моделей в интеллектуальных естественно-языковых системах.

Методы исследования: В качестве основного математического аппарата выбраны логический анализ, алгебра конечных предикатов. Они представляются удобным средством для логико-математических построений, создания полноценного аппарата лингвистической алгебры. Также использовались основные понятия булевой алгебры и теории графов.

Предложены модели линейных логических операторов первого и второго рода, которые описывают функционирование реляционных сетей и позволяют упростить решение системы логических уравнений на каждом такте реляционной сети.

Разработан метод построения логической ассоциативной структуры реляционной сети, который базируется на использовании аппаратной памяти с ассоциативным доступом и позволяет упростить доступ к данным и повысить быстродействие поиска решений.

Разработан метод построения предикатной модели предложения путем введения дополнительных и несущественных предметных переменных, который основан на аппарате алгебры конечных предикатов и лингвистических экспериментах и обеспечивает однозначность предикатной модели предложения.

Развита лингвистическая алгебра путем разработки метода построения формул алгебры булевых функций, которые моделируют предложения естественного языка, что позволяет упростить формализацию проективных предложений.

Практическая ценность исследования. Разработанные математические методы формализации структуры естественного языка, алгоритмы и программная система предназначены для автоматизированных информационных систем с естественно-языковым интеллектуальным интерфейсом, для логической поддержки проектирования информационных структур. Математические и программные результаты работы могут быть использованы в системах автоматической обработки текстовой информации (биржи услуг, электронная почта и т.д.).

ЛИНГВИСТИЧЕСКИЕ СТРУКТУРЫ, РЕЛЯЦИОННЫЕ СЕТИ, АЛГЕБРА КОНЕЧНЫХ ПРЕДИКАТОВ, ЛОГИЧЕСКИЙ АНАЛИЗ, ЛИНЕЙНЫЙ ЛОГИЧЕСКИЙ ОПЕРАТОР.

СОДЕРЖАНИЕ

ВСТУПЛЕНИЕ

В связи с ускорением развития науки и техники одной из первоочередных задач становится развитие методов создания интеллектуальных систем разнообразного назначения, посредством которых осуществляется автоматизация научных исследований и процессов производства. Применяемые сейчас информационные технологии еще несовершенны, несмотря на то, что интеллектуальные информационные системы (ИИС) уже способны выполнять некоторые функции, считавшиеся ранее исключительно прерогативой человека: доказывать математические теоремы, переводить тексты с одного языка на другой, диагностировать болезни и многое другое. Поэтому вопрос разработки новых технологий, подкрепленных качественно выполненным программным обеспечением персональных компьютеров и другими приложениями, позволяющими пользователю удовлетворять свои информационные запросы и решать поставленные перед ним задачи любой степени сложности, является одним из важнейших в области перспективных направлений развития компьютерных наук.

Для эффективного использования возможностей вычислительной техники (прежде всего в информационно-поисковых, экспертных системах разных областей применения, в библиографических системах, в системах машинного перевода) необходимо, чтобы язык обмена информацией “человек-система” был максимально приближен к естественному. Это значительно облегчит задачу обучения при работе с интеллектуальными системами, упростит и сам процесс работы. Важную роль при разработке математического обеспечения ИИС играют реляционные и логические способы представления знаний [53]. Одним из эффективных универсальных математических средств для описания информации являются алгебры предикатов и предикатных операций. На языке этих алгебр легко и удобно описывать различную формализуемую информацию, моделировать интеллектуальную деятельность человека.

В рамках научного направления бионики интеллекта ведутся научные исследования и разработки теории интеллекта [14, 15]. Теория интеллекта представляет собой науку о математическом описании детерминированных, дискретных и конечных интеллектуальных процессов, воспроизводимых человеческим разумом, и структур, обеспечивающих реализацию таких процессов, которая ориентирована на совершенствование цифровой вычислительной техники и ее практическое использование. В качестве формального языка, на котором можно было бы математически описывать структуры и функции человеческого интеллекта, в теории интеллекта разработан язык алгебры конечных предикатов и предикатных операций [19, 34, 35]. Язык алгебры конечных предикатов и предикатных операций эффективен и удобен для описания различной формализуемой информации, а также моделирования деятельности человека.

В настоящее время последние разработки в области теории интеллекта связаны с проектированием реляционных сетей [12-15]. Реляционные сети представляют собой схемную реализацию формул алгебры конечных предикатов. Методы синтеза реляционных сетей базируются на принципе параллельной обработки символьной информации. Эффективность работы реляционной сети и экономность ее аппаратной реализации в виде программируемой матрицы FPGA [21, 80, 104] во многом определяется правильным выбором метода построения архитектуры реляционной сети для заданной задачи. Актуальность данного исследования определяется перспективностью использования полученных алгебрологических моделей структур русского языка, позволяющих существенно расширить класс задач, решаемых на ЭВМ в реальном темпе времени. В частности, для формализации многих информационных процессов, в том числе и разработки интеллектуальных систем общения с компьютером на естественном языке; развития аппарата формального описания объектов различной природы.

1 РАЗРАБОТКА УНИВЕРСАЛЬНОГО АЛГЕБРОЛОГИЧЕСКОГО АППАРАТА ДЛЯ ФОРМАЛИЗАЦИИ СТРУКТУР ЕСТЕСТВЕННОГО ЯЗЫКА

1.1. Исследования в области теории информатизации

В качестве математического фундамента изучения информационных процессов развивается специфический формальный аппарат, называемый теорией интеллекта [14, 15]. Задачей теории интеллекта является описание естественных информационных процессов, протекающих в человеке, и технических информационных процессов [1, 8, 25, 80, 86, 87, 90, 106, 119]. Наиболее полным и доступным для изучения носителем этих информационных процессов является человек, его интеллект. В результате изучения естественного интеллекта создаются многие устройства и системы, способные воспринимать, хранить, обрабатывать и передавать информацию. Во многих случаях они выполняют эту работу быстрее и лучше человека. В качестве языка теории интеллекта используется логическая математика [116, 121]. Общая теория интеллекта совпадает с той частью логической математики, которая перспективна для информатизации; она изучает математические структуры, которые выявляются при математическом описании конкретных механизмов реальных интеллектуальных систем [75, 77, 80, 81]. Однако логическая математика гораздо шире теории интеллекта. В ней имеются такие разделы, которые пока что еще не востребованы информатизацией [115].

Если для описания объектов внешнего мира требуется язык функций, то для моделирования интеллекта человека, его внутреннего мира необходим более общий язык отношений и действий над ними [34, 35, 36, 68, 77]. Понятие отношения по сути эквивалентно понятию предиката. Алгебраический аппарат предикатов и предикатных операций оказывается наиболее эффективным и удобным для изучения интеллектуальной деятельности человека и представляется наиболее перспективным в этой области [1, 4, 17, 19]. В настоящее время логическая математика представляет собой совокупность разнообразных алгебр, наличие каждой из которых определяется соответствующими предметными областями [35, 36, 40, 45, 47, 49]. В первую очередь отметим алгебру конечных предикатов, предложенную в работе [15]. Она представляет собой обобщение аппарата булевых функций и аппарата многозначной логики. Использование этой алгебры позволило приступить к формальному описанию абстрактных понятий, которыми пользуется человек в своей интеллектуальной деятельности. При этом возникла необходимость в разработке алгебр конечных предикатов более высокого порядка, которые позволили бы в гораздо более компактной и естественной форме математически описывать многочисленные и разнообразные понятия высокой степени абстракции, широко используемые человеческим интеллектом. В работах [29, 35] были предложены алгебры подстановочных операций, предикатных операций, прикладная и фундаментальная алгебры предикатных операций.

В настоящее время для развития логической математики и ее приложений в искусственном интеллекте особую роль играет алгебро-логический анализ естественного языка и мышления человека [2, 7, 23, 24, 30, 32, 37, 53]. Язык и мышление человека можно рассматривать как созданную природой реально действующую алгебро-логическую систему. Важно досконально разобраться в механизме языка и мышления. Как показывают уже имеющиеся достижения в этой области, такие знания приводят к развитию аппарата логической математики. Так, Джордж Буль, изучая в середине 19-го столетия логическую структуру сложных предложений естественного языка, открыл алгебру логики, которая впоследствии стала теоретической базой построения схем первых ЭВМ и программ для них. В конце того же века Готлоб Фреге, алгебраизируя естественный язык математика, извлек из него кванторы, играющие в современной логической математике важную роль. В 70-х годах 20-го столетия попытки формального описания морфологии естественного языка привели к открытию алгебры предикатов. Наконец, работы 90-х годов по абстрактному описанию смысла текстов естественного языка содействовали отысканию целого семейства алгебр предикатных операций. На сегодняшний день логическая математика представляет собой богатую содержанием и быстро развивающуюся область знания. Основной метод ее разработки в настоящее время – это алгебраизация логики, воплощенной в механизмах естественного языка и мышления человека.

Уже сегодня логическая математика доказала свою высокую эффективность в роли абстрактного средства анализа естественного интеллекта и синтеза искусственного интеллекта. Можно ожидать, что широкое внедрение в практику информатизации аппарата логической математики и его дальнейшее развитие позволят существенно повысить темпы информатизации. Освоение же естественного языка и мышления вычислительной машиной многократно усилит мощь искусственного интеллекта. Роль алгебрологического аппарата неизмеримо возрастает в связи с последними достижениями в разработке сверхбыстродействующих компьютеров, функционирующих по принципам работы мозга [81, 97, 106]. Ожидается, что изыскания в этой области приведут в обозримом будущем к отказу от компьютеров последовательного действия, повсеместно используемых в настоящее время, и переходу к машинам параллельного действия. А это, в свою очередь, радикально изменит методы программирования и общения человека с машиной.

Развитие логической математики выявило глубокое родство архитектуры логической и числовой ветвей математики и вместе с тем обнаружило существенные различия между ними в деталях. Интенсивно развивается кванторная алгебра [22], являющаяся, как выяснилось, логическим аналогом интегрального исчисления и функционального анализа. Обнаружился логический аналог аналитической геометрии и линейной алгебры – так называемая логическая алгебра [36]. Алгебра предикатов выполняет роль логического аналога школьной алгебры. Буквально на наших глазах рождаются новые разделы логической математики.

В настоящее время логическая математика широко применяется в физике интеллекта – при формальном описании систем естественного интеллекта [93, 94, 99, 104] и в технике интеллекта – при разработке систем искусственного интеллекта [6, 31, 46, 49, 100]. Эти применения порождают массу новых запросов со стороны практики к разработкам в области логической математики. В частности, в последнее время на стыке логической математики и теории категорий начали появляться новые алгебрологические результаты, которые имеют хорошие перспективы для применений в деле автоматизации программирования [120, 127]. Ожидается, что эти применения приведут к существенному ускорению роста производительности труда программиста, разработке принципиально новых высокоэффективных методов написания компьютерных программ, формированию нового взгляда на программирование [27].

Изыскания в области логической математики содействуют лучшему уяснению природы интеллекта самого человека, перспектив его дальнейшего развития. Одной из важных разработок такого рода является аксиоматическое описание структур, обнаруживаемых в самой логической математике. Абстрактное описание объектов логической математики требует опережающей конструктивной разработки гораздо более сложных инструментальных логических средств, которые впоследствии сами становятся предметом аксиоматического анализа.

1.2. Исследования в области формального описания механизмов естественного языка

Самые сложные проблемы современной лингвистики – проблема языка и речи и проблема предложения. Они до сих пор не получили удовлетворительного решения. Решение этих проблем способно создать основу для решения многочисленных практических задач, выдвигаемых перед лингвистической наукой научно-технической революцией. Эта направленность лингвистических исследований получает все более широкое признание [54, 57, 58, 79, 80, 92].

Знания, представленные в автоматизированных информационных системах в формализованном виде, являются одним из основных средств, обеспечивающих человеку возможность общения с компьютером на естественном языке. Эта проблема поставила задачу создания интеллектуального интерфейса. Общая структура и основные функции естественно-языкового интерфейса (общения, автоматического синтеза программ, обоснования и обучения) достаточно глубоко исследованы Поспеловым Г.С., Половым Э.В., Поспеловым Д.А., Шенком Р., Вейшедлом Р.М., Слокумом Дж., Нариньяни А.С. и другими учеными. Результаты их исследований достаточно широко представлены в литературе [21, 24, 45, 49, 80, 82, 105]. В зависимости от целей создания естественно-языковых систем состав вышеперечисленных функций может варьироваться, но независимо от проблемной области, в которой работает пользователь, функционирование интеллектуального естественно-языкового интерфейса обязательно предполагает реализацию функции общения. При этом в функции общения основным компонентом является понимание, включающее анализ и интерпретацию знаний.

В области искусственного интеллекта под знаниями понимается "совокупность сведений (у индивидуума, общества или системы искусственного интеллекта) о мире (конкретной предметной области, совокупности объектов или объекта), включающих в себя информацию о свойствах объектов, закономерностях процессов и явлений, правилах использования этой информации для принятия решений. Модель предметной области включает в себя, как правило, два вида знаний – общие знания об окружающей действительности и знания, характерные только для моделируемой предметной области.

Для эффективного использования знаний в информационных системах они должны быть формализованы в соответствии с определенными правилами, т.е. объединены в некоторую структуру, организация которой позволяет системе понимать, каким образом отдельные части знаний связаны между собой и как манипулировать этими взаимосвязями. Знания могут быть закодированы в различных формах в зависимости от используемого метода представления знаний. Наиболее удачный его выбор, позволяющий адекватно описывать некоторую предметную область и хорошо понимаемый и используемый системой обработки информации, является ключом к успешному функционированию автоматизированной информационной системы. Представление знаний предопределяет возможности базы знаний, используемой для хранения информации. Однако верно и обратное утверждение: чтобы информационная система удовлетворяла определенным прикладным потребностям, должно быть создано соответствующее представление знаний.

Большое влияние на скорость и качество обработки информации интеллектуальными системами оказывает форма представления этой информации в компьютере [85]. Вообще говоря, в различных информационных системах используются различные способы представления знаний в зависимости от конкретных областей применения систем [123]. Для описания знаний используются различные языки представления знаний (ЯПЗ), классы которых соответствуют различным методам представления знаний. В настоящее время широкое распространение получили ЯПЗ четырех типов: сетевые, фреймовые, продукционные и логические [49].

Сетевые ЯПЗ в качестве модели используют семантические сети [21], представляющие собой достаточно удобное средство описания информации в вычислительных системах. Они могут применяться при моделировании человеческой памяти, а также для формализации семантики, естественно-языковых единиц различных уровней в том числе и свободных словосочетаний. Семантические сети эффективно используются при доказательстве теорем [121], оказывают немалую помощь в организации ввода новых элементов и полного хранения задачи в памяти машины.

Семантические сети, используемые в информационных системах, в зависимости от уровня сложности решаемых задач могут быть простыми и расширенными [50]. Простые сети допускают простую и весьма конкретную содержательную интерпретацию. Однако они не обладают достаточными средствами для представления переменных, функциональных символов, n-арных предикатов. В расширенных сетях ребра могут представлять собой сложные математические модели преобразования одних переменных в другие [45].

В настоящее время существуют системы обработки ЕЯ информации, применяющие модель предметной области в качестве основного средства семантического анализа [45]. При этом в качестве ЯПЗ используются фреймы, дающие представление о стереотипной ситуации посредством множества слотов, принимающих определенные значения и в случае необходимости имеющих некоторые присоединенные процедуры [46]. Фреймы предложены Минским в 1975 году и представляют собой сложные структуры для формализации каких-либо стандартных ситуаций. В зависимости от класса стереотипной ситуации различают фреймы визуальных образов, семантические, фреймы-сценарии и др. [73]. Фреймы, являющиеся ЯПЗ в системах искусственного интеллекта, могут быть представлены в виде графа, имеющего несколько уровней вершин, иерархически связанных между собой. Описание так называемых терминалов, то есть вершин нижнего уровня, состоит из имени конкретного терминала, заполнителя и характеристики данного заполнителя.

Возможность перехода к иерархической семантической сети относится к основным достоинствам фреймовых ЯПЗ, так же, как и компактная, наглядная и единообразная запись информационных единиц и использование концепции "умолчания" при желании избежать кванторных утверждений.

Фреймовые ЯПЗ являются в настоящее время одним из основных методов формального представления знаний в системах искусственного интеллекта. Однако они имеют ряд существенных недостатков, среди которых немаловажное место занимает их практическая непригодность для детального изучения словосочетаний, и более мелких единиц естественного языка.

Продукционные ЯПЗ ориентированы преимущественно на представление процедурных знаний [49]. Использование продукционных ЯПЗ сталкивается с проблемой поиска упорядоченного набора продукций, при котором достигается намеченная цель. Таким образом, решение задачи в этом случае может быть сведено к поиску правил дедуктивного вывода. Выбор нужного правила имеет важное значение для качественного производительного функционирования системы, а также для обеспечения возможности ее совершенствования. Положительными сторонами продукционных систем являются простота создания и понимания отдельных правил, простота пополнения и модификации, реализации механизма логического вывода. К недостаткам таких систем можно отнести неясность взаимных отношений между правилами, сложность оценки целостного образа знаний, отличие от человеческой структуры знаний, отсутствие гибкости в логическом выводе.

ЯПЗ, опирающиеся на использование синтаксически правильных формул некоторой формальной логической системы, называются логическими [49]. Формальная система представляет собой множество чисто абстрактных, объектов, в котором заданы правила оперирования набором символов в сугубо синтаксической трактовке без учета смыслового содержания [116].

Чаще всего в качестве формальной логической системы для представления знаний используется аппарат математической логики [79] и в частности логика предикатов как один из ее разделов [85]. Логика предикатов составляет математический фундамент, лежащий в основе моделирования естественного языка. На стыке математической логики и реляционной алгебры находится алгебра конечных предикатов [15], позволяющая описывать конечные, детерминированные и дискретные информационные процессы с помощью уравнений.

Большинство логических ЯПЗ базируется на исчислении предикатов [75], которое уже в 50-е года применялось в программах искусственного интеллекта. В основе представления знаний с использованием аппарата исчисления предикатов лежит преобразование информации об объектах моделируемой предметной области в формулы логики предикатов и добавление их в систему в качестве аксиом.

Логические ЯПЗ обладают высокой степенью формализма, что существенно облегчает решение вопроса об организации знаний в автоматизированных информационных системах. Несомненное преимущество логического подхода для решения задач обработки естественного языка состоит в том, что построенная математическая модель в этом случае описывается тем же языком, что и знания. Используемое в логической модели исчисление предикатов можно легко сочетать с достаточно эффективным механизмом вывода, например резолюцией. Достоинствами логических моделей является единственность теоретического обоснования и возможность реализации системы формально точных определений и выводов. Именно по этой причине интеллектуальные системы, использующие логическую модель знаний, получили достаточно широкое распространение [80].

К недостаткам использования логического построения систем относится неконструктивность и семантическая ограниченность. В то же время человеческая логика часто не ограничивается жесткими рамками формализма логических языков и является интеллектуальной моделью с нечеткой структурой. Тем не менее, рамки формальной логики все более расширяются, подтверждением чему является появление модальных, многозначных и вероятностных логик. Это позволяет расширить возможности применения логики в информационных системах [22].

Многочисленные современные практические приложения абстрактной алгебры и логического анализа в базах данных и интеллектуальных системах обусловили возросший интерес к возможности формального описания текстов естественного языка [21]. На основе применения алгебраических методов в теоретическом программировании были разработаны различные трансляторы с языков высокого уровня и различные алгоритмические алгебры [123]. В языках логического типа использовались либо исчисление высказываний, либо исчисление предикатов первого порядка.

Проведенный анализ показывает, что описанные ЯПЗ во многом близки между собой. По сути, они обладают одинаковыми возможностями для представления знаний в информационных системах. Ни один из рассмотренных ЯПЗ не является настолько универсальным, чтобы можно было бы пренебречь другими. Однако к настоящему времени наметилась явная тенденция преимущественного развития математических методов. Математический аппарат имеет развитый теоретический фундамент и позволяет на формальном уровне описать взаимосвязи между компонентами предметной области. Его описательная мощность превышает возможности других методов.

1.3. Алгебрологическая формализация семантики естественного языка

Попытки математического описания естественного языка возникли в середине 50-х годов в связи с общим стремлением автоматизировать (и соответственно математизировать) различные процессы человеческой деятельности. Сейчас математические методы изучения языка превратились в самостоятельную дисциплину. Поскольку в естественном языке проявляется весь механизм мышления человека, то, изучая механизм языка, мы выходим на познание механизма мышления. Верно, видимо, также и то, что только полное познание человеческого интеллекта даст исчерпывающее описание лингвистических структур. Содержательное изучение языка и мышления человека для этой цели недостаточно. Дело в том, что мы сами являемся носителями языка, мы им свободно владеем, и очевидность многих лингвистических связей порождает иллюзию простоты языка. Изучение языка содержательным способом субъективно, это результат интроспективного психологического изучения языка. Субъективные методы исследования пользуются в науке плохой репутацией как недостаточные. Необходим переход на объективное изучение языка и мышления.

Опыт объективного изучения явлений во внешнем мире свидетельствует о том, что для получения качественных результатов необходимо переходить на язык математики, изложение результатов исследований внешнего мира на естественном языке недостаточно для достаточно глубокого познания явлений внешнего мира. Чтобы описывать объекты исследования на языке математики, необходимо иметь в своем распоряжении достаточный математический аппарат описания. Опыт развития физики свидетельствует о том, что математический язык приходится постоянно развивать параллельно с развитием физических исследований. Обнаруживается, что математического языка постоянно не хватает. Только при достаточном его развитии можно надеяться на успешное развитие исследований реальных объектов.

Логическая математика имеет тесные и глубокие связи с языкознанием. Во всяком случае, естественный язык – это такой механизм, который относится к компетенции логической математики. Как алгебраическая система, он развит гораздо лучше, чем современный аппарат логической математики. Логической математике есть, что перенять у естественного языка. Человек никогда не откажется от естественного языка в пользу языка математики или языков программирования, вообще – искусственных языков, разрабатываемых информатикой. Логическая математика проявляет глубокий интерес к изучению механизма естественного языка [37]. Формализация семантики привела к ее логизации. Это, в первую очередь, компонентный анализ: изучается смысл текста как класс всех текстов одинакового смысла [116].

Алгебра логики появилась в результате изучения в середине 19-го столетия логической структуры сложных предложений Джорджем Булем. Алгебра предикатов появилась в результате попыток математического описания процессов склонения и спряжения слов (то есть процессов словоизменения) в 70-х годах 20-го столетия. Если удастся вскрыть алгебро-логическую структуру естественного языка, то тогда сам алгебрологический аппарат, развиваемый наукой и используемый на практике информатикой в процессе компьютеризации и информатизации, будет поднят на неизмеримо более высокий уровень. Важно извлечь из естественного языка алгебро-логический аппарат, который может оказаться намного более совершенным, чем известный и используемый в настоящее время. Пока это не удалось сделать в полном объеме. Алгебра предикатов появилась, как результат длительного процесса попыток описания некоторых простейших синтаксических механизмов языка [7, 8, 33, 74]. Алгебра логики появилась в результате изучения смысловой структуры сложносочиненных предложений [93]. Идут два встречных процесса: усиление алгебры предикатных операций и выявление алгебраической структуры естественного языка. Вместе взятые, они ведут к единой цели: разработке более совершенного аппарата логической математики. Логическая математика должна уже сейчас пытаться формализовать лингвистические механизмы, чтобы выявить то, чего ей еще недостает. Таким образом, формальное описание механизмов естественного языка сильно стимулирует развитие логической математики.

Сейчас в информатике распространяется и утверждается в правах мнение, что естественный язык с математической точки зрения представляет собой некоторую алгебру. Эта алгебра называется лингвистической. Она пока в значительной мере гипотетична. Изучение лингвистической алгебры показывает, что она относится к классу логических алгебр, а именно: в ней можно обнаружить и алгебру предикатов, и алгебру предикатных операций [34, 35]. Смысл текста можно представить в виде формулы алгебры предикатов, а его синтаксическую структуру – в виде формулы алгебры предикатных операций. Механизм естественного языка можно формально описать только средствами логической математики. Какой-то (по-видимому – не самый лучший) алгебро-логический язык в логической математике мы уже имеем. Мы знаем, что он универсален. С его помощью можно вскрыть и формально описать алгебро-логическую структуру естественного языка. Главная задача логической математики в языкознании – узнать, какой именно вариант алгебры предикатных операций реализован в естественном языке, убедительно показать алгебраическую природу языка и мышления (или развить какую-либо еще более удачную альтернативную теорию, если алгебро-логический подход окажется недостаточным). Если это удастся сделать, то далее надо будет научиться переводить фразы естественного языка на язык логической математики. Формальное описание всех механизмов естественного языка в полном объеме – это пока неподъемная задача для логической математики и теории интеллекта. Чтобы стать способной решить эту задачу, логическая математика должна многократно усилить свой алгебраический инструментарий, развивать язык формального описания.

Однако ни одна из современных моделей общения на естественном языке не базируется на формальном аппарате, который мог бы гарантировать истинность выдвигаемых идей, теорий, принципов и методов. Это обстоятельство и сложность самой проблемы общения на естественном языке приводит к утверждению, что критерием истины может служить только действующая модель участника общения, т.е. устройство или система программ, подтверждающая выдвигаемые положения.

Этому важному вопросу моделирования структуры естественного языка – критериям истинности модели – посвящена работа [104]. В основе работы лежит идея компараторной идентификации. Ее отличие от классической, прямой идентификации в том, что идентифицируется объект, выходные сигналы которого недоступны для непосредственного измерения – именно таково реальное положение с идентификацией естественного языка. Суть ее заключается в том, что испытуемый (носитель естественного языка) в специально поставленных экспериментах своими физическими реакциями формирует значения некоторых предикатов P_1, P_2,…, P_r. В этих экспериментах выявляются свойства предикатов P_1, P_2,…, P_r, которые формально записываются в виде логических уравнений, связывающих предикатные переменные X_1, X_2,…, X_r. Некоторые из этих уравнений используются в роли аксиом или исходных постулатов модели. Из аксиом, как из уравнений, находятся значения предикатных переменных X_1, X_2,…, X_r, которыми являются соответственно предикаты P_1, P_2,…, P_r. Внутренняя структура найденных предикатов характеризует те или иные детали структуры естественного языка.

Для проведения эксперимента вводится ситуационно-текстовый предикат. Это бинарный предикат $P(X, Y)$, заданный на декартовом произведении $A \times B$ множеств A и B. В роли сигнала X исследователь предъявляет испытуемому ситуации. Под ситуациями понимается физическая реальность, данная испытуемому в ощущениях. Множество A всех ситуаций, используемых в данном эксперименте, исследователь может выбрать произвольно, сообразуясь с конкретными задачами. В роли сигнала Y исследователь предъявляет испытуемому тексты. Мысль, заключенная в тексте, называется смыслом этого текста. Под текстом понимается физический объект, а не субъективный результат его восприятия. Предполагается, что тексты описывают свойства ситуаций. Поэтому, когда испытуемый воспринимает пару (X, Y), образованную из ситуации X и текста Y, то он может установить, соответствуют друг другу или нет данные ситуация и текст. Предполагается, что все используемые в эксперименте ситуации и тексты таковы, что испытуемый всегда однозначно реагирует ответом 0 или 1 на любую пару (*X*, *Y*), принадлежащую множеству A×B. Значение t=P(*X*, *Y*) предиката P называется истинностным значением текста *Y* для ситуации *X*.

Далее вводятся функция восприятия ситуации $x=f(X)$ и функция понимания текста $y=g(Y)$. Эти функции описывают соответственно процессы преобразования ситуации в восприятие этой ситуации и текста в смысл этого текста (при этом восприятий *x* может оказаться меньше, чем ситуаций, а мыслей *y* меньше, чем текстов, т.е. происходит факторизация). Ниже мы еще вернемся к рассмотрению результатов работы [104].

Одна из основных проблем формализации естественного языка – большое отличие понятия отношения в математике и в естественном языке. В математике формальное отношение вводится через декартово произведение множеств. Оно определяется на конкретном множестве и носит чисто экстенсиональный (формальный) характер. В естественном языке используется неформальное (интенсиональное) понятие отношения, не использующее явной привязки к конкретным множествам. Например, понятие «больше» или отношение «краснота» обычно понятны людям без определения множества объектов, на котором они применяются. Предполагается, что людям известна некоторая операция F, позволяющая для данной последовательности N определить, выполняется ли на N рассматриваемое отношение R. Кроме того, с точки зрения целесообразности, отношения задаются или экстенсионально или интенсионально. Отсюда возникает задача преобразования одного отношения в другое. В этом преобразовании наибольшую сложность представляет преобразование формального отношения в интенсиональное.

Для эффективного проникновения в механизм языка нужен какой-то принципиально новый подход к его изучению. Такой подход мы видим в сравнении естественного языка с языком математики. Основанием к этому служит предположение о существовании глубокой аналогии, близкого родства между этими двумя языками. Математический язык, в отличие от естественного, хорошо формализован, его механизм понятен. Он создавался, в отличие от естественного языка, не стихийно сам по себе, а целенаправленно отдельными людьми и поэтому весь на виду. Если гипотеза о наличии глубокой аналогии между этими двумя языками оправдается, тогда можно будет найти ответы на многие трудные вопросы, касающиеся механизма естественного языка, обращаясь за подсказкой к языку математики [30].

Авторы работ [30, 36] пришли к выводу, что предложения естественного языка, как и математические утверждения, в содержательном плане представляют собой отношения и ничего сверх этого. Исходя из гипотезы о тесном родстве естественного и математического языков, они обнаружили в предложении естественного языка не только истинностную переменную (о чем науке уже давно было известно), но также и не замеченные никем до сих пор (если полагаться на выполненный нами анализ литературных источников) предметные переменные. В результате подтвердилось предположение, что предложение естественного языка является, как и математическое высказывание, носителем некоторого предиката, представляющего собой зависимость истинностной переменной предложения от его предметных переменных. Этот предикат принимается в качестве формального эквивалента смысла предложения естественного языка. Подход также не является завершенной формальной теорией естественного языка, имеет множество белых пятен и предлагает только фундамент, базовые идеи, на которых предполагается огромная длительная работа по формализации смысловых структур естественного языка, однако он представляется нам наиболее перспективным.

Следующим шагом в этом же направлении можно считать введение понятия лингвистической алгебры. Язык человека рассматривается как некая алгебра, а его тексты – как формулы этой алгебры. Важно выяснить, что собой представляет лингвистическая алгебра. Если это удастся сделать, тогда на ее языке можно будет математически записывать содержание естественно-языковых текстов. Решая уравнения лингвистической алгебры, можно будет по запросам извлекать необходимые знания из информационных массивов, записанных на естественном языке. Перевод естественно-языкового текста на формальный язык лингвистической алгебры можно интерпретировать как его понимание, а обратный перевод формул лингвистической алгебры в естественно-языковый текст – как словесное выражение мысли.

Для математического описания строения текстов авторы работ [34, 35] предлагают рассматривать предложение как алгебраическую формулу и выяснить, каковы принципы построения формул данной алгебры. Если естественный язык – это некая алгебра, а тексты – ее формулы, то тогда в чем заключается значение текста? Поскольку значением обычной математической формулы является функция, выраженная ею, то по аналогии каждый текст выражает некоторый предикат, т.е. функцию $P(x_1, x_2,..., x_m)=\xi$, аргументами $x_1, x_2,..., x_m$ которой служат предметные переменные, результатом же ξ действия этой функции является одно из двух значений: 0 или 1. Символ 0 выражает факт ложности предложения, символ 1 – факт его истинности. Любое утверждение формирует с помощью предложения некоторое уравнение $P(x_1, x_2,..., x_m)=1$, связывающее между собой предметные переменные $x_1, x_2,..., x_m$.

Лингвистическая алгебра представляет собой алгебру многоместных предикатов. Арность предикатов, выражаемых ее формулами, может быть очень большой. Например, роман "Война и мир" Толстого, несомненно, представляет собой единый связный текст (т.е. одну формулу), в нем же речь идет о многих тысячах различных предметов. Лингвистическая алгебра имеет очень большое число базисных элементов, в роли которых выступают предикаты, представленные различными словами естественного языка. В качестве базисных операций в ней используются отрицание, дизъюнкция и конъюнкция предикатов, а также кванторы существования и общности.

Многие слова (типа «очень») выражают специальные предикатные операции. Весьма вероятно, что в базисе лингвистической алгебры содержатся и иные, не выявленные нами элементы и операции, но даже такой неполный их ассортимент весьма внушителен и выглядит вполне достаточным для придания ей свойства универсальности. Тем не менее, вопрос о полноте лингвистической алгебры нуждается в дополнительном изучении.

Еще один аспект проблемы алгебрологического моделирования естественного языка рассматривается в [30]. Авторы изучают булеву алгебру мыслей. Под мыслью понимается некоторая факторизующая функция, реализуемая человеком – носителем естественного языка. Аргументами этой функции могут быть тексты естественного языка или ситуации физического мира. Рассматривается два вида возможных действий над мыслями – операции и предикаты. Результатом операции над мыслью является новая мысль. Например, одноместная операция отрицания мысли. Другой вид действия на мысли – это предикаты. Предикаты, как и операции, могут быть одно- и многоместными. В первом случае они воздействуют на одну мысль, во втором – на набор мыслей. При определении значения предиката человек проверяет выполнение некоторого условия, которое и определяет вид предиката. Например, двуместный предикат следования – это предикат, проверяющий, вытекает ли логически вторая мысль из первой.

Из того факта, что мысли эффективно описываются текстами естественного языка, авторы делают вывод, что тексты выполняют роль формул некоторой алгебраической системы. Значительная часть работы [30] посвящена построению и изучению этой алгебры.

Продолжение исследований в этом направлении можно найти в работе [71]. В ней изучаются предикаты первого и высших порядков. По сути, ведется работа по расширению лингвистической алгебры. Вводятся правила построения логических формул произвольного порядка. Разрабатывается теория моделей и операций над ними. Исследуются свойства предикатов с произвольно выбранной областью определения.

1.4. Методы построения реляционных сетей

Процесс обработки данных в интеллектуальных системах основан на конкретных знаниях о предметной области. В настоящее время разработано достаточно много различных способов представления знаний, каждый из которых обладает своими достоинствами и недостатками. Универсальным языком представления знаний является язык алгебры конечных предикатов и предикатных операций, разработанный в рамках научного направления теории интеллекта [15]. Язык алгебры предикатов и предикатных операций эффективен и удобен для описания различной формализуемой информации, моделирования деятельности человека и формирования запросов в базах данных.

Последние разработки в теории интеллекта, в области развития математического аппарата алгебры предикатов и предикатных операций, связаны с проектированием реляционных логических сетей [18, 43, 59, 103]. Методы синтеза реляционных сетей базируются на принципе параллельной обработки символьной информации. Реляционная сеть выполняет функции своего рода базы данных. В роли атрибутов базы данных в сети выступают предметные переменные, представленные полюсами сети. В роли доменов атрибутов базы данных в сети выступают области изменения предметных переменных сети. В роли таблиц, хранящихся в базе данных, выступают бинарные отношения, представленные ветвями сети. В роли исходных данных, поступающих в базу данных, для решаемой задачи выступают подмножества некоторых из областей изменения предметных переменных реляционной сети. В роли запроса в базу данных выступают некоторые из предметных переменных реляционной сети. Таким образом, входная и выходная информация в реляционных сетях представлена в виде принадлежности к множеству, т.е. речь идет о представлении не данных, а знаний.

В области построения реляционных сетей получен ряд важных результатов. Первая работа, опубликованная на эту тему [12] посвящена анализу путей повышения производительности электронных вычислительных машин. Для решения этой проблемы предложен математический аппарат для формульного представления отношений и действий над ними на базе алгебры конечных предикатов. Отношения интерпретируются как мысли интеллекта, а действия над ними – как мышление. Схемная реализация формул, описывающих алгебро-логические структуры, приводит к характерным инженерным сетям, которые называются реляционными. Основная идея построения реляционных сетей заключается в бинарной конъюнктивной декомпозиции многоместных предикатных моделей, т.е. в переходе от системы многоместных предикатов к системе бинарных предикатов, что позволяет параллельно обрабатывать задачи, требующие вычислений большого объема или в реальном темпе времени. Примером таких задач является обработка связного текста – перевод с одного языка на другой и распознавание речи.

В работе [51] предложены методы декомпозиции морфологических предикатов и пространств и декартовой декомпозиции многоместных предикатов с помощью унарных предикатов, которые являются необходимым этапом построения формальной модели в виде бинарной реляционной сети. Методы были практически применены для декомпозиции предиката, моделирующего склонение полных непритяжательных имен прилагательных. Разработан метод бинаризации алгебраической формульной записи морфологических структур, позволяющий преобразовать любую модель, представленную системой логических уравнений в алгебре конечных предикатов, в систему бинарных предикатов. Это позволяет построить реляционную сеть для параллельной обработки символьной информации.

В работе [78] предложена общая алгебраическая модель реляционных сетей, использованная для схемной реализации формул, которые описывают алгебрологические модели естественного языка. Разработан метод формульной записи отношений, алгебраически формализованы различные типы соответствий и основные операции над отношениями. Разработаны варианты сокращения базиса элементов дизъюнктивно-конъюнктивной алгебры предикатов. Разработана схемная реализация предикатных операций замены, перестановки и подстановки, а также метод схемного вычисления значений кванторов.

В работе [65] предложен метод потактовой работы бинарных реляционных сетей средствами алгебры конечных предикатов. Метод основан на бинарной декомпозиции отношений и позволяет параллельно обрабатывать символьную информацию в узлах бинарных логических сетей. Средствами алгебры предикатных операций разработан метод функционирования бинарных реляционных сетей, который базируется на линейном логическом операторе и позволяет решать систему бинарных уравнений логических сетей, а также схемно их реализовывать. Средствами алгебры конечных предикатов и алгебры предикатных операций разработана математическая модель бинарной реляционной сети склонения регулярных имен существительных русского языка, что позволяет аппаратно их реализовывать и использовать для автоматического решения задач анализа, синтеза и нормализации словоформ.

В работе [43] предложен метод формального описания операций реляционной алгебры на языке алгебры конечных предикатов, который использует операции подстановки дизъюнктивно-конъюнктивной подстановочной алгебры предикатов, операции объединения и проекции алгебры функциональных данных, а также отображение операций реляционной алгебры в систему предикатов. Метод позволяет значительно повысить скорость вычислений операций реляционной алгебры путем аппаратной реализации этих операций в виде логических сетей параллельного действия. Разработан и обоснован метод бинарной декомпозиции функциональных предикатов, основанный на декартовой декомпозиции предикатов, их дизъюнктивном разложении и бинарной декомпозиции с помощью кванторов существования. Метод позволяет сократить объем необходимых вычислений при реализации логических сетей за счет уменьшения количества значений вспомогательной переменной. Построена модель логической сети для флективной обработки глагольных словоформ.

Научное направление, к которому относится настоящее исследование, – математическое моделирование функциональной структуры человеческого интеллекта. Основной проблемой в этой области является способность компьютера к общению на естественном языке. С математической точки зрения естественный язык представляет собой некоторую алгебру, названную лингвистической. Она относится к классу логических алгебр, в ней можно усмотреть и алгебру предикатов, и алгебру предикатных операций. Мысли – это предикаты, предложения – это формулы алгебры предикатов, выражающие эти предикаты. Грамматическая структура предложений выражается на языке алгебры предикатных операций. Когда мы обмениваемся предложениями, то, по сути дела, обмениваемся предикатами, отношениями. Когда изучается семантика предложения, то оно предстает перед исследователем как формула алгебры предикатов. Если же изучается строение предложения, то удобно представлять его как формулу алгебры предикатных операций.

Естественный язык человека представляет собой какой-то алгебраический аппарат в действии, по-видимому, одну из разновидностей алгебры предикатных операций, точнее – физическую реализацию этого алгебраического аппарата. Алгебра предикатов тоже в естественном языке присутствует, но она вложена в алгебру предикатных операций. Нужно узнать, какой именно вариант алгебры предикатных операций реализован в естественном языке и нет ли в естественном языке еще каких-нибудь алгебраических структур сверх этого. Если это удастся сделать, то далее надо научится переводить фразы естественного языка на язык логической математики, то есть осуществить, математическое описание смысловой структуры текстов, их грамматической структуры. Грамматика так легко охватывает всю семантику текста потому, что грамматическая алгебра охватывает семантическую алгебру.

Имеется и еще одна важная проблема – перевод с одного языка на другой. Это не только практическая, но и теоретическая проблема [31]. Два различных варианта естественного языка (например, русский и английский) – это разные лингвистические алгебры, заданные на одном и том же носителе – множестве мыслей. Синтаксически же алгебры у разных естественных языков могут существенно отличаться, но все равно они будут алгебрами предикатных операций.

В лингвистической алгебре имеются два яруса – семантический и синтаксический. Ставится задача как можно конкретнее и детальнее описать механизм лингвистической алгебры. Каждое предложение и образуемый из предложений текст выражают некоторую мысль, поэтому его следует рассматривать как элементы носителя лингвистической алгебры, а соответствующие им предложения (тексты) – как описывающие их формулы. Оказывается, что предложения и тексты строятся тем же способом, который используется при образовании формул. Предложения, выражающие одну и ту же мысль, можно рассматривать как тождественные формулы лингвистической алгебры.

Мысли интернациональны, каждую из них можно выразить на любом естественном языке. Разные языки (например, русский и английский) можно рассматривать как различные лингвистические алгебры, заданные над одним и тем же носителем – множеством всевозможных мыслей, которыми способны оперировать люди. Предложения разных языков, выражающие одну и ту же мысль, – это тождественные формулы. Перевод текстов с одного языка на другой следует считать переходом от формул одной лингвистической алгебры к тождественным им формулам другой алгебры, заданной над тем же носителем. В роли носителя любой лингвистической алгебры выступает множество всех доступных человеку мыслей. Смысл текста можно представить в виде формулы алгебры предикатов, а его синтаксическую структуру – в виде формулы алгебры предикатных операций. Механизм естественного языка можно формально описать только средствами логической математики. Какой-то (по-видимому – не самый лучший) алгебро-логический язык в логической математике мы уже имеем. Мы знаем, что он универсален. С его помощью можно вскрыть и формально описать алгебро-логическую структуру естественного языка.

Главная задача логической математики в языкознании – узнать, какой именно вариант алгебры предикатных операций реализован в естественном языке, убедительно показать алгебраическую природу языка и мышления (или развить какую-либо еще более удачную альтернативную теорию, если алгебро-логический подход окажется недостаточным). Если это удастся сделать, то далее надо будет научиться переводить фразы естественного языка на язык логической математики.

Важно извлечь из естественного языка алгебро-логический аппарат, который может оказаться намного более совершенным, чем известный и используемый в настоящее время. Пока это не удалось сделать в полном объеме. Алгебра предикатов появилась, как результат долгих безуспешных попыток описать некоторые простейшие синтаксические механизмы языка. Алгебра логики появилась в результате изучения смысловой структуры сложносочиненных предложений. Усиление алгебры предикатных операций и выявление алгебраической структуры языка, вместе взятые, ведут к единой цели – разработке совершенного аппарата логической математики. Пытаться формализовать лингвистические механизмы логическая математика должна уже сейчас, чтобы выявить то, чего ей недостает.

Если удастся извлечь из анализа языка более совершенный алгебро-логический язык, то это сильно увеличит возможности разработчиков информационных систем, новых информационных технологий. Представляется, что нет такой другой области знания, которая в большей степени, чем эта, могла бы способствовать уяснению глубинной природы человека и повышению темпов компьютеризации и информатизации общества. Как в математическом, так и в естественном языках мысли выражаются в виде предложений. О математическом языке мы имеем формализованных знаний неизмеримо больше, чем о естественном, поэтому математический язык может дать нам подсказку при математическом описании строения естественного языка. Наша главная цель – математически описать смысл (или значение, содержание) предложения естественного языка. Для этого сначала рассмотрим, как формализуется смысл математического предложения.

Предложение математического языка называются высказываниями. Высказывания содержат константы и переменные. Константы высказывания выражают вполне определенные математические объекты. Переменная распространяется на некоторое множество. Она принимает значения из какого-то множества констант. Переменные бывают трех видов: параметры, связанные и свободные переменные. Переменные, используемые в роли констант, называются параметрами высказывания. Переменные, стоящие при знаках кванторов, называются связанными переменными. Связанные переменные – это вспомогательные переменные, они выполняют служебную роль, нужны они для записи кванторов. Используются они в роли индексов логического суммирования (при кванторах существования) и логического перемножения (при кванторах общности). Остальные переменные в высказываниях называются свободными. Они также называются переменными высказываний. Они (и только они) являются настоящими переменными в высказываниях.

Развитие теории линейных логических преобразований является актуальной задачей для дальнейшей разработки теории реляционных сетей, поскольку функционирующая реляционная сеть представляет собой систему взаимодействующих линейных логических преобразований [25, 59, 88]. В данной работе развивается математический аппарат линейных логических преобразований за счет введения двойственных линейных логических преобразований для обеспечения эффективной работы реляционных сетей.

2 ПРЕДСТАВЛЕНИЕ ОТНОШЕНИЙ И ДЕЙСТВИЙ НАД НИМИ СРЕДСТВАМИ АЛГЕБРЫ КОНЕЧНЫХ ПРЕДИКАТОВ

В разделе описаны способы формульного представления отношений и действий над ними. Рассмотрены методы решения логических уравнений реляционных сетей с помощью линейных логических операторов. Сформулированы и доказаны основные свойства логических операторов первого и второго рода.

2.1. Формульное представление отношений в алгебре предикатов

Теория интеллекта, в отличие от классической физики, изучает и математически описывает явления, наблюдаемые во внутреннем мире человека или машины. Информационные процессы не сводятся к количественным изменениям, они имеют, в основном, не числовую, а логическую природу. Поэтому теория интеллекта нуждается в аппарате, соответственно, логической, а не числовой, математики, т.е. в таком алгебраическом языке, на котором можно было бы описывать информационные объекты.

В роли главных объектов в теории интеллекта выступают мысли и мышление. Мысли формально выражаются в виде отношений, а процесс мышления – в виде последовательности операций над отношениями [34-36]. Таким образом, логическая математика должна включать в себя алгебру отношений и алгебру операций над отношениями. Математика определяет отношения, как подмножества какого-либо декартова произведения. Несмотря на простоту определения (а может быть, именно благодаря ей), понятие отношения обладает ничем не ограниченной описательной силой. Имеет место поразительный факт: за всю историю развития науки в ней не встретилось ни одного объекта, который нельзя было бы формально выразить в виде какого-либо отношения. Этот факт позволяет рассматривать логическую математику как универсальное средство формального описания любых объектов, в том числе – любых информационных процессов.

Классическая математика оперирует функциями. Для их записи она разработала специальный язык формул. К сожалению, отношения – не функции, их не удается непосредственно выразить формулами. Отношения записывают в виде множеств наборов элементов, а также в виде таблиц, графиков или графов. Гильберт в 1920 году предложил обойти эту трудность, описывая формулами не сами отношения, а некоторые взаимно однозначно связанные с ними функции. Этим достигается (хотя и опосредовано) формульное представление отношений. Он показал, как это делается, на следующем простейшем примере и дальше развивать эту тему не стал.

Дано множество $P=\{a, b\}$, состоящее из двух элементов a и b. (Множество P можно понимать как одноместное отношение $P=\{(a), (b)\}$, состоящее из двух однокомпонентных наборов (a) и (b), образованных из элементов a и b. Требуется записать его в виде формулы. Гильберт заменяет множество P функцией $P(x)$, обращающейся в 1 при $x=a$ или $x=b$ и в 0 – при любых других значениях переменной x. Ее он записывает в виде формулы $P(x)=(x=a)\vee(x=b)$. Запись $x=a$ в ней понимается как функция $a(x)$, обращающаяся в 1 при $x=a$ и в 0 – при любом $x\neq a$. Аналогично понимается и выражение $x=b$. Знак $\vee$ обозначает операцию дизъюнкции двоичных знаков: $0\vee 0=0$, $0\vee 1=1\vee 0=1\vee 1=1$. Имея функцию $P(x)$, всегда можно по ней восстановить множество P. Таким образом, записанная формула, действительно, выражает множество $\{a, b\}$. Ясно, что таким способом можно записать в виде формулы любое множество.

Метод, указанный Гильбертом, можно легко распространить на любые отношения. Любая функция $P(x_1, x_2,..., x_m)$, отображающая множество $A=A_1\times A_2\times ...\times A_m$ в множество $\sum=\{0, 1\}$, называется предикатом, заданным на A. Пусть L – множество всех отношений на A, M – множество всех предикатов на A. Отношение $\boldsymbol{P}\in L$ и предикат $P\in M$ называются соответствующими друг другу, если при любых $x_1\in A_1$, $x_2\in A_2$,..., $x_m\in A_m$

$$P(x_1, x_2,..., x_m) = \begin{cases} 1, \text{ если } (x_1, x_2,..., x_m) \in \boldsymbol{P}; \\ 0, \text{ если } (x_1, x_2,..., x_m) \notin \boldsymbol{P}. \end{cases} \tag{2.1}$$

Соотношение (2.1) устанавливает биективное соответствие между всеми отношениями множества L и всеми предикатами множества M [36].

К дизъюнкции, используемой Гильбертом, добавляем конъюнкцию двоичных знаков: $0\wedge0=0\wedge1=1\wedge0=0$, $1\wedge1=1$.

Введенных средств достаточно для формульного представления любого отношения. Делается это следующим образом. Пусть, к примеру, дано бинарное отношение:

$$P=\{(3, 0), (2, 1), (1, 2), (0, 3), (2, 2)\}.$$

Соответствующий ему предикат запишется формулой:

$$P(x, y)=x^3y^0\vee x^2y^1\vee x^1y^2\vee x^0y^3\vee x^2y^2.$$

В общем случае формула для любого предиката $P\in M$ отыскивается по правилу:

$$\mathrm{P}(x_1, x_2,..., x_m) = \bigvee_{\substack{a_1\in A_1 \\ a_2\in A_2 \\ \cdots\cdots \\ a_m\in A_m}} (P(a_1, a_2,..., a_m) x_1^{a_1} x_2^{a_2} ... x_m^{a_m}). \tag{2.2}$$

Справа от знака равенства фигурируют константы 0 или 1 под видом выражения $P(a_1, a_2,..., a_m)$. Логическая сумма $\vee$ берется по всем наборам (a_1,

$a_2, \ldots, a_m) \in A$, формула (2.2) называется совершенной дизъюнктивной нормальной формой (СДНФ) предиката P.

Приведенная здесь система формульного представления предикатов называется алгеброй предикатов с базисными элементами в виде предикатов узнавания элементов и базисными операциями дизъюнкции и конъюнкции предикатов. Алгебра предикатов полна в том смысле, что ее формулами можно выразить любой предикат. Алгебру предикатов можно дополнить операцией отрицания предикатов, однако это не расширяет ее выразительные возможности. Формулы алгебры предикатов допускают тождественные преобразования по законам булевой алгебры, которые дополняются законами истинности

$$\bigvee_{a \in A} x_i^a = 1, \quad (i = \overline{1, m}); \tag{2.3}$$

ложности

$$\text{если } a \neq b, \text{ то } x_i^a x_i^b = 0, \quad (i = \overline{1, m}; \ a, b \in A_i) \tag{2.4}$$

и отрицания

$$\overline{x_i^a} = \bigvee_{\substack{b \in A_i \\ b \neq a}} x_i^b = 1, \quad (i = \overline{1, m}; \ a \in A_i). \tag{2.5}$$

Кроме описанного, возможен и другой способ опосредованного формульного описания отношений, основанный на замене их отображениями. Сначала рассмотрим одноместные отображения, а затем и многоместные. Будем говорить, что каждому предикату $F(x, y)$, определенному на $A \times B$, соответствует отображение $F(x) = y$, действующее из A в B. Если для элемента $x \in A$ существует элемент $y \in B$ такой, что $F(x_1, y) = 1$, то будем писать, что $F_{.}(x) = y$, в противном случае будем писать $F(x) \neq y$ и говорить, что отображение

F сопоставляет элементу $x \in A$ элемент $y \in B$. Обратим внимание на то, что знак = здесь обозначает не отношение равенства, а нечто иное. Это следует из того, что одному и тому же элементу x некоторые отображения могут сопоставлять два (и более) различных элемента y_1 и y_2, и тогда получится, что $y_1 \neq y_2$, что невозможно для равенства. Такие отображения называются многозначными. Знак = употреблен в нестандартной роли с той целью, чтобы подчеркнуть, что понятие отображения есть результат обобщения понятия функции. Существуют и такие отображения, которые некоторым элементам $x \in A$ не сопоставляют ни одного элемента $y \in B$. Такие отображения называются частичными. Некоторые из отображений являются функциями, они называются всюду определенными и однозначными или иначе – функциональными отображениями. В математической литературе иногда допускается неточное словоупотребление, в результате которого появляются термины “частичная функция” и “многозначная функция”, где слово “функция” не вполне корректно употребляется вместо точного термина “отображение”.

Формульное представление отображений основано на импликативном разложении предиката:

$$F(x, y) = \bigwedge_{a \in A} (x^a \supset F(a, y)). \tag{2.6}$$

Согласно (2.6) любое отображение $F(x)=y$, действующее из A в B, которое соответствует предикату $F(x, y)$ на $A \times B$, можно записать системой условий:

$$x^a \supset F(a, y), \qquad (a \in A).$$

Многоместные отображения получаем, переходя от предиката $F(x, y)$ на $A \times B$ к предикату $F(x_1, x_2, \ldots, x_n, y)$ на $A_1 \times A_2 \times \ldots \times A_n \times B$ и от отображения $F(x)=y$, действующего из множества A в множество B, к отображению $F(x_1, x_2, \ldots, x_n)=y$

из $A_1 \times A_2 \times ... \times A_n$ в B. Все сказанное выше об одноместных отображениях легко распространяется также и на многоместные отображения. Импликативное разложение многоместного предиката имеет вид:

$$F\ x_1, x_2, ..., x_n, y\ = \bigvee_{\substack{a_1 \in A_1 \\ a_2 \in A_2 \\ \cdots\cdots \\ a_n \in A_n}} (x_1^{a_1} x_2^{a_2} ... x_n^{a_n} \supset F(a_1, a_2, ..., a_n, y)). \qquad (2.7)$$

Отображение $F(x_1, x_2, ..., x_n)$=y, соответствующее предикату $F(x_1, x_2, ..., x_n, y)$, записывается в виде системы условий:

$$x_1^{a_1} x_2^{a_2} ... x_n^{a_n} \supset F(a_1, a_2, ..., a_n, y), \qquad (2.8)$$

где $a_1 \in A_1$, $a_2 \in A_2, ..., a_n \in A_n$. Образ $N \subseteq B$ множества $M \subseteq A_1 \times A_2 \times ... \times A_n$ и прообраз M' множества N относительно отображения $F(x_1, x_2, ..., x_n)=y$, определяемого предикатом $F(x_1, x_2, ..., x_n, y)$, отыскиваются по формулам:

$$N(y) = \exists x_1 \in A_1 \exists x_2 \in A_2 ... \exists x_n \in A_n (F(x_1, x_2, ..., x_n, y) \wedge M(x_1, x_2, ..., x_n)); \qquad (2.9)$$

$$M'(x_1, x_2, ..., x_n) = \exists y \in B (F(x_1, x_2, ..., x_n, y) N(y)). \qquad (2.10)$$

Векторные отображения получаем, переходя от предиката $F(x_1, x_2, ..., x_n, y)$ на $A_1 \times A_2 \times ... \ ... \times A_n \times B$ к предикату $F(x_1, x_2, ..., x_n, y_1, y_2, ..., y_p)$ на $A_1 \times A_2 \times ... \times A_n \times B_1 \times B_2 \times ... \times B_p$ и от отображения $F(x_1, x_2, ..., x_n)$=y, действующего из $A_1 \times A_2 \times ... \times A_n$ в B, к отображению $F(x_1, x_2, ..., x_n) = (y_1, y_2, ..., y_p)$ из $A_1 \times A_2 \times ... \times A_n$ в $B_1 \times B_2 \times ... \times B_p$. Для случая векторных отображений зависимости (2.7)-(2.10) обобщаются следующим образом:

$$F(x_1, x_2,..., x_n, y_1, y_2,..., y_p) = \bigvee_{\substack{a_1 \in A_1 \\ a_2 \in A_2 \\ \cdots\cdots \\ a_n \in A_n}} (x_1^{a_1} x_2^{a_2} \ldots x_n^{a_n} \supset$$

$$\supset F(a_1,\ a_2,...,\ a_n,\ y_1,\ y_2,...,\ y_p));$$

$$x_1^{a_1} x_2^{a_2} \ldots x_n^{a_n} \supset F(a_1, a_2,...,a_n, y_1, y_2\ ,..., y_p),$$

где $a_1 \in A_1$, $a_2 \in A_2$,..., $a_n \in A_n$;

$$N(y)=\exists x_1 \in A_1 \exists x_2 \in A_2 \ldots \exists x_n \in$$
$$A_n(F(x_1, x_2,..., x_n, y_1, y_2,..., y_p)M(x_1, x_2, ..., x_n));$$

$$M'(x_1, x_2,..., x_n)=\exists y_1 \in B_1 \exists y_2 \in B_2 \ldots \exists y_p \in$$
$$B_p(F(x_1, x_2,...,x_n, y_1, y_2,..., y_p)N(y_1, y_2,..., y_p)).$$

Операции над отношениями можно записывать формулами, заменяя их операциями над соответствующими предикатами [65].

2.2. Линейные логические операторы

Отношения, задаваемые бинарными предикатами, называются соответствиями. Совокупность Q всех предметов $s \in A_2$, удовлетворяющих уравнению $K(r,\ s) = 1$, где $r \in A_1$, называется полным образом предмета r относительно соответствия K. Полный образ Q предмета r может быть вычислен по формуле:

$$\exists x_1 \in A_1(x_1^{\,r} \cdot K(x_1,\ x_2)) = Q(x_2).$$

Здесь $Q(x_2)$ – предикат, соответствующий множеству Q.

Совокупность P всех предметов $r \in A_1$, удовлетворяющих уравнению $K(r,\ s) = 1$, где $s \in A_2$, называется полным прообразом предмета s относительно соответствия K. Полный прообраз P предмета s может быть вычислен по формуле

$$\exists x_2 \in A_2 (x_2{}^s \cdot K(x_1,\ x_2)) = P(x_1).$$

или по формуле

$$\forall x_2 \in A_2 (x_2{}^s \supset K(x_1,\ x_2)) = P(x_1).$$

Максимальным образом множества $P \subseteq A_1$ относительно соответствия $K(r,\ s) = 1$ называется множество $Q_{\max} \subseteq A_2$, представляющее собой объединение образов всех предметов $r \in P$. Он вычисляется по формуле:

$$\exists x_1 \in A_1 (P(x_1) \cdot K(x_1,\ x_2)) = Q_{\max}(x_2) \qquad (2.11)$$

Минимальным образом множества $P \subseteq A_1$ относительно соответствия $K(r,\ s) = 1$ называется множество $Q_{\min} \subseteq A_2$, представляющее собой пересечение образов всех предметов $r \in P$. Он вычисляется по формуле:

$$\forall x_1 \in A_1 (P(x_1) \supset K(x_1,\ x_2)) = Q_{\min}(x_2). \qquad (2.12)$$

Преобразование $F(P) = Q_{\max}$ вида (2.11) обладает аддитивностью $F(P_1 \vee P_2) = F(P_1) \vee F(P_2)$ относительно операции дизъюнкции и однородностью $F(\alpha P) = \alpha F(P)$ относительно операции конъюнкции,

$\alpha \in \{0,\ 1\}$. Оно называется линейным логическим оператором первого рода.

Преобразование $Y(P) = Q_{\min}$ вида (2.12) обладает аддитивностью $Y(P_1 \wedge P_2) = Y(P_1) \wedge Y(P_2)$ относительно операции конъюнкции и однородностью $Y(\alpha \vee P) = \alpha \vee Y(P)$ относительно операции дизъюнкции. Оно называется линейным логическим оператором второго рода. Можно доказать, что любой линейный логический оператор первого рода выражается в виде (2.11), а второго – в виде (2.12) при подходящем выборе предиката $K(x_1, x_2)$, который называется ядром линейного логического оператора.

Линейные логические операторы играют определяющую роль при моделировании психологического мира человека. Именно они описывают основные действия реляционной сети, реализующей процессы мышления в числовой природе и в технике.

Линейные логические операторы можно наглядно изображать при помощи двудольных графов. Возьмем, к примеру, следующее ядро линейного логического оператора:

$$K(x_1, x_2) = x_1^{\,1}(x_2^{\,a} \vee x_2^{\,b}) \vee x_1^{\,2} x_2^{\,d} \vee \vee x_1^{\,3} x_2^{\,c} \vee x_1^{\,4}(x_2^{\,d} \vee x_2^{\,e}) \qquad (2.13)$$

Ему соответствует двудольный граф, изображенный на рис. 2.1.

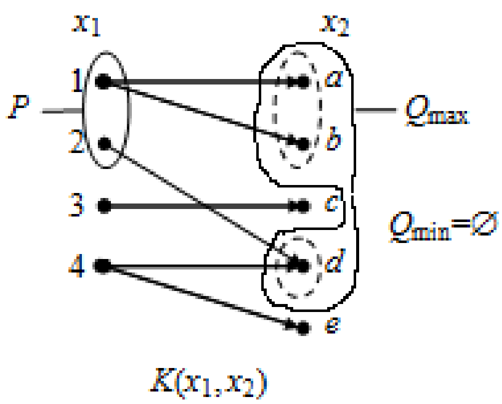

Рис. 2.1 Двудольный граф линейного логического оператора

Образуем линейный логический оператор первого рода с ядром (2.13) и подадим на его вход множество $P=\{1, 2\}$, изображенное на рис. 2.1 в виде овала. Из точек этого множества вправо простираются стрелки, оканчивающиеся элементами a, b, d, образующими максимальный образ $Q_{\max}$ множества P. Полным образом элемента 1 служит множество $\{a, b\}$, элемента 2 – множество $\{d\}$. Эти множества не пересекаются. Поэтому выходным сигналом $Q_{\min}$ линейного логического оператора второго рода будет пустое множество. Если поменять в двудольном графе направление всех стрелок на обратные, получаем дуальный граф с тем же ядром $K(x_1, x_2)$. Дуальному графу соответствуют линейный логический оператор первого рода

$$\exists x_2 \in A_2(Q(x_2)K(x_1, x_2)) = P_{\max}(x_1) \tag{2.14}$$

и второго рода

$$\forall x \in A_2(Q(x_2) \supset K(x_1, x_2)) = P_{\min}(x_2), \tag{2.15}$$

называемые дуальными по отношению к операторам (2.11) и (2.12). Важно отметить, что дуальные логические операторы далеко не всегда возвращают исходное множество к первоначальному виду. Так, например, множество $Q_{\max} = Q$, формируемое оператором (2.11) (рис. 2.1), возвращается дуальным оператором (2.14) в виде более широкого множества $P_{\max}$, чем исходное множество P (рис. 2.2).

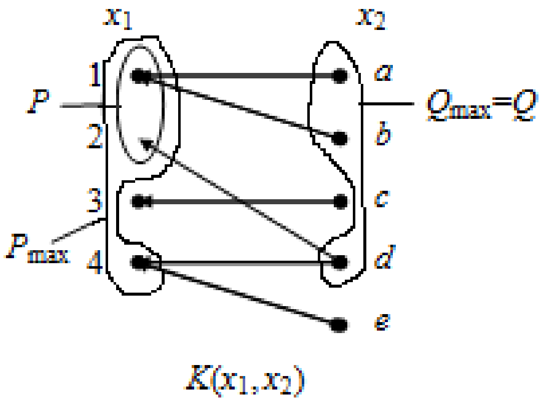

Рис. 2.2. Двудольный граф дуального оператора

Реляционная сеть представляет собой систему взаимодействующих линейных логических преобразований [25]. Целью данного раздела является дальнейшее развитие математического аппарата линейных логических преобразований за счет введения двойственных линейных логических преобразований для обеспечения эффективной работы реляционных сетей.

Каждая модель реляционной сети характеризуется своим предикатом модели. Работа реляционной сети заключается в решении системы логических уравнений на каждом такте. Для каждого из своих логических уравнений вида

$$K(x, y) = 1, \quad (2.16)$$

где K – это бинарное отношение, заданное соответствующим бинарным предикатом, реляционная сеть в первом полутакте каждого такта отыскивает:

1) по известному знанию $P(x)$ о значении переменной x знание $Q'(y)$ о значении переменной y;

2) по известному знанию $Q(y)$ о значении переменной y знание $P'(x)$ о значении переменной x (рис. 2.3).

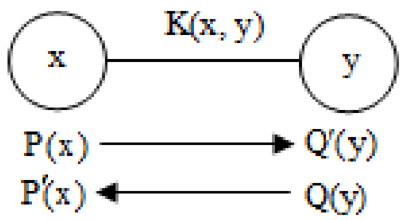

Рис. 2.3. Два полутакта работы реляционной сети

2.3. Линейные логические операторы первого рода

Рассмотрим логическое уравнение (2.16). Ограничим возможные значения переменной x каким-нибудь множеством $P \subseteq U$. Тогда значения переменной y, удовлетворяющей уравнению (2.16), также окажутся ограниченными некоторым множеством $Q \subseteq U$. Множество Q можно выразить через множество P следующей зависимостью:

$$\exists x \in U\ K(x,\ y)P(x) = Q(y), \tag{2.17}$$

где $P(x)$ – предикат, соответствующий множеству P, $Q(y)$ – предикат, соответствующий множеству Q. Запишем предикатную операцию (2.17) в краткой форме $F_K(P) = Q$.

Утверждение 1. Любая предикатная операция F_K *однородна* относительно операции конъюнкции: для любых $K \subseteq U^2$, $\alpha \in \{0,\ 1\}$ и $P \subseteq U$: $F_K(\alpha P) = \alpha F_K(P)$.

Покажем это $F_K(\alpha P) = \exists x \in U\ K(x, y)\alpha P(x) =$ $= \alpha \exists x \in U\ K(x, y)P(x) = \alpha F(P)$.

Утверждение 2. Любая предикатная операция F_K *аддитивна* относительно операции дизъюнкции: для любых $K \subseteq U^2$ и $P, Q \subseteq U$ $F_K(P \vee Q) = F_K(P) \vee F_K(Q)$.

Покажем это

$$F_K(P \vee Q) = \exists x \in U\ K(x,y)(P(x) \vee Q(x)) =$$
$$= \exists x \in U\ K(x,y)P(x) \vee \exists x \in U\ K(x,y)Q(x) = F_K(P) \vee F_K(Q).$$

Преобразование вида (2.17) называется *линейной предикатной операцией* $F_K(P) = Q$.

Утверждение 3. Любая аддитивная и однородная предикатная операция $F_K(P) = Q$ выражается в виде (2.17).

Множество $Q = F_K(P)$ называется *образом* множества P по отношению K.

Рассмотрим пример нахождения образа некоторого множества. Пусть универсум предметов $U = \ a,b,c,d,e\ $, универсум переменных $V = \ x,y\ $, $K(x,y) = x^a y^a \vee x^a y^b \vee x^b y^c \vee x^c y^c \vee x^c y^d \vee x^d y^e$, $P = \ a,b\ $. Находим $Q(y) = F_K(P(x))$ по формуле (2.17):

$$Q(y) = \exists x \in \{a,b,c,d,e\}\ K(x,y)P(x) =$$
$$= \exists x \in \{a,b,c,d,e\}\ (x^a y^a \vee x^a y^b \vee x^b y^c \vee$$
$$\vee x^c y^c \vee x^c y^d \vee x^d y^e)(x^a \vee x^b) =$$
$$= \exists x \in \{a,b\}\ (x^a y^a \vee x^a y^b \vee x^b y^c \vee$$
$$\vee x^c y^c \vee x^c y^d \vee x^d y^e) =$$
$$= (a^a y^a \vee a^a y^b \vee a^b y^c \vee a^c y^c \vee a^c y^d \vee a^d y^e) \vee$$
$$\vee (b^a y^a \vee b^a y^b \vee b^b y^c \vee b^c y^c \vee b^c y^d \vee b^d y^e) =$$
$$= (1 \cdot y^a \vee 1 \cdot y^b \vee 0 \cdot y^c \vee 0 \cdot y^c \vee 0 \cdot y^d \vee 0 \cdot y^e) \vee$$
$$\vee (0 \cdot y^a \vee 0 \cdot y^b \vee 1 \cdot y^c \vee 0 \cdot y^c \vee 0 \cdot y^d \vee 0 \cdot y^e) = = y^a \vee y^b \vee y^c.$$

Итак, $Q = \{a,b,c\}$.

Графическое представление нахождения образа множества

$P = \ a,b\ \xrightarrow{F_K} Q = \ a,b,c\ $ представлено на рис. 2.4.

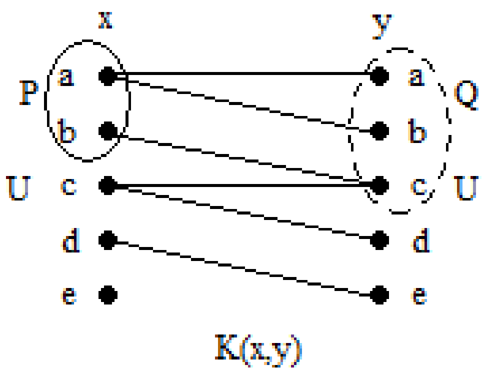

Рис. 2.4. Образ множества $P = \ a,b\ \xrightarrow{F_K} Q = \ a,b,c$

Множество P', определяемое по формуле

$$\exists y \in U \; K(x,\, y) Q(y) = P'(x), \tag{2.18}$$

называется *прообразом* множества Q по отношению K. Предикатная операция (2.18) кратко записывается в виде $F'_K(Q) = P'$, она называется *дуальной* по отношению к операции $F_K(P) = Q$ (2).

Рассмотрим пример отыскания прообраза некоторого множества. Пусть $U = \ a,b,c,d,e\ $, $V = \ x,y\ $, $K(x,y) = x^a y^a \vee x^a y^b \vee x^b y^c \vee x^c y^c \vee x^c y^d \vee x^d y^e$, $Q = \ a,b,c\ $.

Находим $P' = F'_K(Q)$ по формуле (2.18):

$$P'(x) = \exists y \in \{a,b,c,d,e\} \, (x^a y^a \vee x^a y^b \vee x^b y^c \vee$$

$$\vee x^c y^c \vee x^c y^d \vee x^d y^e)(y^a \vee y^b \vee y^c) =$$

$$= \exists y \in \{a,b,c\} \, (x^a y^a \vee x^a y^b \vee x^b y^c \vee x^c y^c \vee$$

$$\vee x^c y^d \vee x^d y^e) = (x^a a^a \vee x^a a^b \vee x^b a^c \vee x^c a^c \vee$$

$$\vee x^c a^d \vee x^d a^e) \vee (x^a b^a \vee x^a b^b \vee x^b b^c \vee x^c b^c \vee$$

$$\vee x^c b^d \vee x^d b^e) \vee (x^a c^a \vee x^a c^b \vee x^b c^c \vee x^c c^c \vee$$

$$\vee x^c c^d \vee x^d c^e) = (x^a \cdot 1 \vee x^a \cdot 0 \vee x^b \cdot 0 \vee x^c \cdot 0 \vee$$

$$\vee x^c \cdot 0 \vee x^d \cdot 0) \vee (x^a \cdot 0 \vee x^a \cdot 1 \vee x^b \cdot 0 \vee x^c \cdot 0 \vee$$

$$\vee x^c \cdot 0 \vee x^d \cdot 0) \vee (x^a \cdot 0 \vee x^a \cdot 0 \vee x^b \cdot 1 \vee x^c \cdot 1 \vee$$

$$\vee x^c \cdot 0 \vee x^d \cdot 0) = x^a \vee x^b \vee x^c .$$

Итак, $P' = \{a,b,c\}$.

Графическое представление нахождения прообраза множества $Q = \ a,b,c \ \xrightarrow{F'_K} P' = \ a,b,c$ представлено на рис. 2.5.

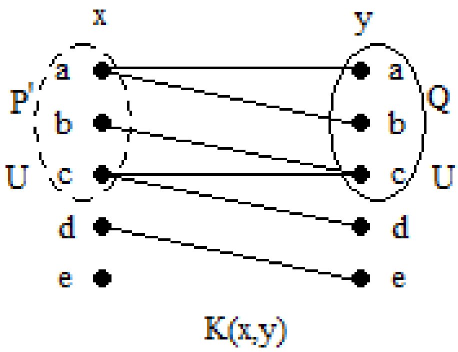

Рис. 2.5. Прообраз множества $Q = \ a,b,c \ \xrightarrow{F'_K} P' = \ a,b,c$

Преобразования вида (2.17)-(2.18) называются *линейными предикатными операциями первого рода* или *линейными логическими операторами первого рода* $F_K(P) = Q$, $F'_K(Q) = P'$.

$K(x,\ y)$ называется *ядром линейного логического оператора.*

2.4. Линейные логические операторы второго рода

Существуют предикатные операции второго рода, которые выражаются в следующем виде:

$$F_K^{\ *}(P) = \neg F_K(\neg P).$$

Получим общий вид предикатной операции второго рода:

$$F_K^{\ *}(P) = \neg F_K(\neg P) = \neg\exists x \in U\ K(x,y)\neg P(x) = \forall x \in U\ \neg(K(x,y) \wedge \overline{P(x)}) =$$
$$= \forall x \in U\ (\overline{K(x,y)} \vee \overline{\overline{P(x)}}) = \forall x \in U\ (\overline{K(x,y)} \vee P(x)) = \forall x \in U\ (K(x,y) \supset P(x)).$$

Т.о., предикатная операция второго рода $F^*(P) = Q$ имеет следующий вид:

$$\forall x \in U\ (K(x,y) \supset P(x)) = Q(y). \qquad (2.19)$$

Утверждение 4. Предикатная операция $F_K^{\ *}(P) = Q$ (4) *однородна* относительно операции дизъюнкции: для любых $K \subseteq U^2$, $\alpha \in \{0,\ 1\}$ и $P \subseteq U$: $F_K^{\ *}(\alpha \vee P) = \alpha \vee F_k^{\ *}(P)$.

Покажем это

$$F_K^{\ *}(\alpha \vee P) = \neg F_K(\neg(\alpha \vee P)) = \neg F_K(\neg\alpha \wedge \neg P)) =$$
$$= \neg(\neg\alpha \wedge F_K(\neg P)) = \neg\neg\alpha \vee \neg F_K(\neg P)) = = \alpha \vee \neg F_K(\neg P)) = \alpha \vee F_K^{\ *}(P).$$

Утверждение 5. Предикатная операция $F_K^{\ *}(P) = Q$ (2.19) *аддитивна* относительно операции конъюнкции: для любых $K \subseteq U^2$ и $P, Q \subseteq U$: $F_K^{\ *}(P \wedge Q) = F_K^{\ *}(P) \wedge F_K^{\ *}(Q)$.

Покажем это

$$F_K{}^*(P \wedge Q) = \neg F_K(\neg(P \wedge Q)) = \neg F_K(\neg P \vee \neg Q)) =$$
$$= \neg(F_K(\neg P) \vee F_K(\neg Q)) = \neg F_K(\neg P) \wedge \neg F_K(\neg Q) == F_K{}^*(P) \wedge F_K{}^*(Q)\,.$$

Преобразование (2.19) называется *линейной предикатной операцией* $F_K{}^*$.

Утверждение 6. Любая аддитивная и однородная предикатная операция $F_K{}^*$ имеет вид (2.19).

Рассмотрим пример нахождения образа некоторого множества с помощью линейного логического оператора второго рода. Пусть $U = \ a,b,c,d,e\ $, $K(x,y) = x^a y^a \vee x^a y^b \vee x^b y^c \vee x^c y^c \vee x^c y^d \vee x^d y^e$, $P = \ a,b\ $. Находим дополнение множества $P = \ a,b$

$$\neg P(x) = \overline{(x^a \vee x^b)} = \overline{x^a} \wedge \overline{x^b} = (x^b \vee x^c \vee x^d \vee x^e)(x^a \vee x^c \vee x^d \vee x^e) =$$
$$= x^c \vee x^d \vee x^e,$$

тогда

$$Q = F_K{}^*(P) = \neg F_K(\neg P) = \neg\exists x \in \{a,b,c,d,e\}\, K(x,y) \neg P(x) =$$
$$= \neg\exists x \in \{a,b,c,d,e\}\,((x^a y^a \vee x^a y^b \vee x^b y^c \vee x^c y^c \vee$$
$$\vee x^c y^d \vee x^d y^e)(x^c \vee x^d \vee x^e)) = \neg\exists x \in \{c,d,e\}\,(x^a \wedge$$
$$\wedge y^a \vee x^a y^b \vee x^b y^c \vee x^c y^c \vee x^c y^d \vee x^d y^e) =$$
$$= \neg((c^a y^a \vee c^a y^b \vee c^b y^c \vee c^c y^c \vee c^c y^d \vee c^d y^e) \vee$$
$$\vee(d^a y^a \vee d^a y^b \vee d^b y^c \vee d^c y^c \vee d^c y^d \vee d^d y^e) \vee$$
$$\vee(e^a y^a \vee e^a y^b \vee e^b y^c \vee e^c y^c \vee e^c y^d \vee e^d y^e)) = \overline{y^c \vee y^d \vee y^e} = \overline{y^c} \wedge \overline{y^d} \wedge \overline{y^e} =$$
$$= (y^a \vee y^b \vee y^d \vee y^e)(y^a \vee y^b \vee y^c \vee y^e) \wedge$$
$$\wedge(y^a \vee y^b \vee y^c \vee y^d) = (y^a \vee y^b \vee y^e) \wedge$$
$$\wedge(y^a \vee y^b \vee y^c \vee y^d) = y^a \vee y^b.$$

Итак, $Q=\{a,b\}$.

Графическое представление нахождения образа множества $P=\{a,b\}\xrightarrow{F_K^{\,*}}Q=\{a,b\}$ представлено на рис. 2.6.

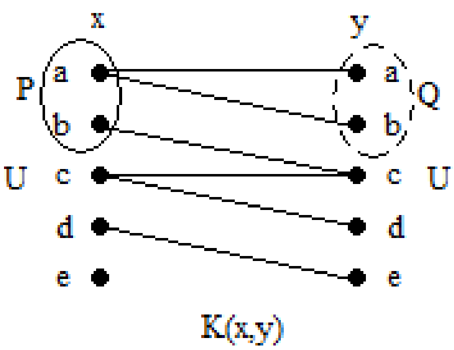

Рис. 2.6. Образ множества $P=\{a,b\}\xrightarrow{F_K^{\,*}}Q=\{a,b\}$

Множество P', определяемое по формуле

$$\forall y\in U\ (K(x,y)\supset Q(y))=P'(x), \tag{2.20}$$

называется *прообразом* множества Q по отношению K. Предикатная операция (2.20) кратко записывается в виде $F_K'^{\,*}(Q)=P'$, она называется *дуальной* по отношению к операции $F_K^{\,*}$.

Рассмотрим пример отыскания прообраза некоторого множества с помощью линейного логического оператора второго рода. Пусть $U=\{a,b,c,d,e\}$, $K(x,y)=x^a y^a\vee x^a y^b\vee x^b y^c\vee x^c y^c\vee x^c y^d\vee x^d y^e$, $Q=\{a,b\}$. Находим дополнение множества $Q=\{a,b\}$

$$\neg Q(y)=\overline{(y^a\vee y^b)}=\overline{y^a}\wedge\overline{y^b}=(y^b\vee y^c\vee y^d\vee y^e)(y^a\vee y^c\vee y^d\vee y^e)=$$
$$=y^c\vee y^d\vee y^e,$$

тогда

$$P' = F_K'^{*}(Q) = \neg\exists y \in U\ K(x,\ y)\neg Q(y) =$$
$$= \neg\exists y \in \{a,b,c,d,e\}\ (x^a y^a \vee x^a y^b \vee x^b y^c \vee x^c y^c \vee x^c y^d \vee x^d y^e)(y^c \vee y^d \vee y^e)$$
$$= \neg\exists y \in \{c,d,e\}\ (x^a y^a \vee x^a y^b \vee \vee x^b y^c \vee x^c y^c \vee x^c y^d \vee x^d y^e)$$
$$= \neg((x^a c^a \vee x^a c^b \vee x^b c^c \vee x^c c^c \vee x^c c^d \vee x^d c^e) \vee$$
$$\vee(x^a d^a \vee x^a d^b \vee x^b d^c \vee x^c d^c \vee x^c d^d \vee x^d d^e) \vee$$
$$\vee(x^a e^a \vee x^a e^b \vee x^b e^c \vee x^c e^c \vee x^c e^d \vee x^d e^e)) =$$
$$= \neg((x^a \cdot 0 \vee x^a \cdot 0 \vee x^b \cdot 1 \vee x^c \cdot 1 \vee x^c \cdot 0 \vee x^d \cdot 0) \vee$$
$$\vee(x^a \cdot 0 \vee x^a \cdot 0 \vee x^b \cdot 0 \vee x^c \cdot 0 \vee x^c \cdot 1 \vee x^d \cdot 0) \vee$$
$$\vee(x^a \cdot 0 \vee x^a \cdot 0 \vee x^b \cdot 0 \vee x^c \cdot 0 \vee x^c \cdot 0 \vee x^d \cdot 1)) =$$
$$= \neg(x^b \vee x^c \vee x^d) = \overline{x^b} \wedge \overline{x^c} \wedge \overline{x^d} = (x^a \vee x^c \vee x^d \vee$$
$$\vee x^e)(x^a \vee x^b \vee x^d \vee x^e)(x^a \vee x^b \vee x^c \vee x^e) =$$
$$= (x^a \vee x^d \vee x^e)(x^a \vee x^b \vee x^c \vee x^e) = x^a \vee x^e.$$

Итак, $P' = \{a,e\}$.

Графическое представление нахождения прообраза множества $Q = \ a,b \xrightarrow{F_K'^{*}} P' = \ a,e$ представлено на рис. 2.7.

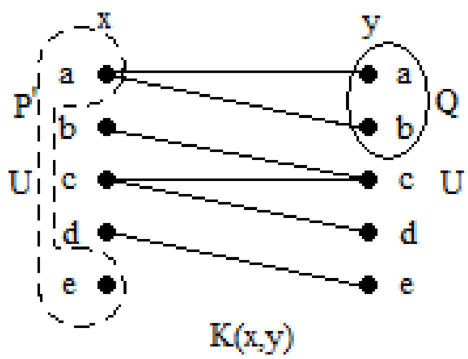

Рис. 2.7. Прообраз множества $Q = \ a,b \xrightarrow{F_K'^{*}} P' = \ a,e$

Т.о., введя в рассмотрение линейные предикатные операции первого и второго рода, мы приходим к следующему выводу: при решения логических уравнений реляционной сети с помощью линейных предикатных операций первого рода или второго рода во втором полутакте каждого такта реляционная сеть находит общую часть $P_i(x)$ всех знаний $P_1'(x), P_2'(x), \ldots, P_l'(x)$ о значении каждой из своих предметных переменных x (l – число ветвей, подходящих к полюсу x) по формулам, соответственно:

$$P_1'(x) \wedge P_2'(x) \wedge \ldots \wedge P_l'(x) = P_i(x);$$
$$P_1'(x) \vee P_2'(x) \vee \ldots \vee P_l'(x) = P_i(x).$$

Использование в реляционной сети линейных логических операторов первого рода имеет преимущество: для их реализации не нужна операция отрицания, а для линейных логических операторов второго рода операции отрицания необходимы – на входе и выходе схемы и для формирования импликации в самом операторе.

2.5. Особенности работы реляционных сетей первого и второго рода

Основные идеи, заложенные в строении реляционных сетей [15], основаны на том, что интеллект человека рассматривается как логика в действии, материальное воплощение механизма логики. Существует ряд работ по алгебраизации логики [36, 88, 91], в которых разработан специальный математический аппарат для формульного представления отношений и действий над ними – алгебро-логические структуры. В них отношения интерпретируются как мысли интеллекта, а действия над ними – как мышление. Механизм, решающий уравнения алгебры предикатов, называется реляционной сетью. Такое название мотивировано тем, что, во-первых, мозг человека реализует нейронную сеть; во-вторых, с психологической точки зрения механизм мышления представляется как ассоциативная сеть; в-третьих, с математической точки зрения механизм мышления предстает как устройство для обработки отношений. Схемная реализация формул, описывающих алгебро-логические структуры, приводит к характерным инженерным сетям. Каждый тип алгебро-логических структур (а таких типов сравнительно немного) приводит к своему типу реляционных сетей. При сопоставлении этих типов сетей с основными типами нейроструктур обнаруживается глубокое сходство строения технических и биологических конструкций. Опираясь на это сходство, можно определить функции различных типов нейронных структур и описать в точных математических и технических терминах принципы функционирования мозга.

Реляционная сеть состоит из полюсов и ветвей, соединяющих полюсы. Каждому полюсу соответствует своя предметная переменная x_i с областью определения A_i $(i=\overline{1,m})$. Пара полюсов x_i и x_j, соединенных ветвью $K(x_i, x_j)$, реализуют линейный логический оператор первого рода

$$\exists x_i \in A_i (P_i(x_i) K_{ij}(x_i, x_j)) = Q_{j\,\max}(x_j) \tag{4.1}$$

или второго рода

$$\forall x_i \in A_i (P_i(x_i) \supset K_{ij}(x_i, x_j)) = Q_{j\,\min}(x_j)\,. \tag{4.2}$$

Сеть называется *первого рода*, если в ней действуют лишь операторы первого рода. Аналогично определяются сети *второго рода*. Если в сети используются операторы обоих видов, сеть называется *комбинированной*. Сеть отыскивает решение уравнения

$$K(x_1, x_2, \ldots, x_m) = 1 \tag{4.3}$$

при ограничениях, накладываемых на области изменения переменных x_i $(i=\overline{1,m})$ $x_i \in P_i$, где $P_i \subseteq A_i$. Если решение уравнения отыскивается при более сложном ограничении

$$L(x_1, x_2, \ldots, x_m) = 1,$$

тогда сеть достраивается таким образом, чтобы она соответствовала уравнению $K' = 1$, где $K' = K \wedge L$. Построению сети, реализующей предикат K, предшествует *бинаризация* предиката K, т.е. представление его в виде

$$K(x_1, x_2, \ldots, x_m) = \bigwedge_{\substack{i=1 \\ j=1 \\ i \neq j}}^{m} K_{ij}(x_i, x_j), \tag{4.4}$$

которая может производиться различными способами. Решение уравнения (4.3) сетью осуществляется по тактам. В течение каждого такт одновременно срабатывают все линейные логические операторы сети. В сети первого рода после каждого такта осуществляется пересечение всех множеств $Q_{j\,\max}$, сходящихся со всех сторон к каждому из полюсов x_j. В сети второго рода множества $Q_{j\,\min}$, наоборот объединяются. Сеть первого рода может формировать лишние решения, а второго – может не найти некоторые из действительных решений. В процессе решения уравнения (4.3) с увеличением номера такта работы сети множества $Q_{j\,\max}$ и $Q_{j\,\min}$ сближаются, причем всегда $Q_{j\,\min} \leq Q_{j\,\max}$, На некотором такте сближение множеств $Q_{j\,\min}$ и $Q_{j\,\max}$ прекращается. Если это достигается одновременно на всех полюсах, то на этом процесс решения уравнения (4.3) заканчивается. Если при всех $j = \overline{1,m}$ оказывается, что $Q_{j\,\min} = Q_{j\,\max}$, то это означает, что сеть нашла все решения, не пропустив ни одного, и не причислила к решению ни одного ошибочного. Если же такое равенство не достигнуто в конце работы сети, это значит, что сеть сработала не вполне эффективно. Этот признак может быть использован при оценке степени доброкачественности метода синтеза сети, в частности – метода бинаризации предиката K (4.3). Важно отметить, что существуют такие методы синтеза сети, которые обеспечивают ее безупречную работу при решении любого уравнения вида (4.3).

3 ЛИНГВИСТИЧЕСКАЯ АЛГЕБРА КАК АППАРАТ ФОРМАЛИЗАЦИИ СМЫСЛА СЛОВОСОЧЕТАНИЙ И ПРЕДЛОЖЕНИЙ ЕСТЕСТВЕННОГО ЯЗЫКА

Идентифицируются некоторые механизмы естественного языка в виде математической структуры, названной лингвистической алгеброй. В ней имеются два яруса – семантический и синтаксический. Первый ярус представлен одним из вариантов алгебры предикатов, второй – алгебры предикатных операций. Рассматривается метод экспериментальной проверки алгебрологических моделей языка. Разрабатывается способ формульной записи смысла словосочетаний и предложений естественного языка.

3.1. Мысли как предикаты

Естественный язык представляет собой сложный объект. Чтобы преуспеть в математическом анализе механизма языка, необходим правильный подход к его изучению. В данном разделе в качестве такового используется гипотеза, гласящая, что естественный язык – это некоторая алгебра, которая называется лингвистической алгеброй. Ставится задача: как можно конкретнее и детальнее описать механизм лингвистической алгебры.

Охарактеризуем общее понятие алгебры, которое затем будет использовано в качестве инструмента математического описания механизма естественного языка. Алгеброй A над множеством A называется любая система формульной записи элементов какого-нибудь множества A. Множество A называется носителем алгебры A. Любая алгебра A над A характеризуется своими базисными операциями, используемыми в роли преобразователей ее элементов, и базисными элементами, выбираемыми из ее носителя A. Множество всех базисных операций алгебры A называется ее базисом операций, множество всех базисных элементов – базисом элементов алгебры A. Базис операций и базис элементов, взятые вместе, образуют базис алгебры A.

Формулой алгебры называется любая запись, которая выражает какую-нибудь суперпозицию базисных операций этой алгебры, примененную к ее базисным элементам. Каждая формула алгебры выражает некоторый элемент ее носителя и в этой роли может использоваться как его имя. Алгебра A называется полной, если каждый элемент ее носителя можно выразить в виде какой-нибудь формулы алгебры A. Базис алгебры называется полным, если эта алгебра полна. Он называется несократимым, если исключение любой операции или элемента из базиса делает его неполным. Любые две формулы алгебры, выражающие один и тот же элемент ее носителя, называются тождественными. Тождеством алгебры A называется любая запись, указывающая какую-нибудь пару тождественных формул алгебры A. Если алгебры A и B заданы над одним и тем же носителем A, то в этом случае допустимо говорить о тождественных формулах различных алгебр. Система тождеств алгебры A называется полной, если из нее можно вывести факт тождественности или нетождественности любых двух формул алгебры A. Система тождеств алгебры называется несократимой, если ни одно из ее тождеств невозможно логически вывести из совокупности остальных. Схемой тождеств алгебры называется любая запись, указывающая какое-нибудь семейство тождеств этой алгебры.

Из гипотезы о том, что естественный язык есть алгебра, можно вывести много разнообразных следствий, которые допускают опытную проверку. Так, каждое предложение и образуемый из предложений текст выражают некоторую мысль, поэтому мысли следует рассматривать как элементы носителя лингвистической алгебры, а соответствующие им предложения (тексты) – как описывающие их формулы. Предложения и тексты должны строиться тем же способом, который используется при образовании формул. Предложения, выражающие одну и ту же мысль, необходимо рассматривать как тождественные формулы лингвистической алгебры.

Мысли интернациональны, каждую из них можно выразить на любом языке. Разные языки (например, русский и английский) необходимо рассматривать как различные лингвистические алгебры, заданные над одним и тем же носителем – множеством всевозможных мыслей, которыми способны оперировать люди. Предложения разных языков, выражающие одну и ту же мысль, – это тождественные формулы. Перевод текстов с одного языка на другой следует считать переходом от формул одной лингвистической алгебры к тождественным им формулам другой, заданной над тем же носителем.

Проверяя в лингвистическом эксперименте следствия, выводимые из этой гипотезы и других допущений, формулируемых ниже, можно подтвердить или опровергнуть исходные положения (аксиомы) создаваемой таким способом теории естественного языка. Если удастся получить большое число разнообразных следствий из аксиом и при этом окажется, что ни одно из них не противоречит фактам языка и речи, то это обстоятельство можно будет расценить как подтверждение развиваемой теории. Если же некоторые из формулируемых гипотез не выдержат испытания или подтвердятся лишь частично, то ничто не помешает заменить их более совершенными, опираясь на полученные отрицательные результаты.

Важно ответить на вопрос: какова структура элементов носителя лингвистической алгебры? Иными словами, какова математическая природа мыслей, выражаемых предложениями и текстами, т.е. что представляет собой смысл (иначе – содержание) любого предложения (текста) естественного языка? На этот счет будем придерживаться следующей гипотезы: мысли – это предикаты. Если это так, тогда лингвистическую алгебру следует рассматривать как алгебру предикатов. Эта гипотеза мотивирована тем, что в математике все мысли (идеи и понятия) выражаются предикатами. Если же естественный язык – тоже математический объект (а мы предположили, что он есть алгебра), то и здесь следует ожидать того же.

Для дальнейшего изложения потребуется понятие предиката и алгебры предикатов. Чтобы убедиться в том, что содержанием любого предложения действительно служит некоторый предикат, достаточно задаться вопросом, что представляет собой содержание какой-нибудь формулы. Ответ очевиден: содержанием формулы является функция, которую она выражает. Но если предложение есть формула, то его содержанием тоже должна быть какая-то функция. Для ответа на этот вопрос заметим, что если предложение используется для характеристики какой-то вполне определенной ситуации (а предложения только для этого и нужны), то оно станет либо истинным, либо ложным. Если же предложение рассматривать вне связи с какой бы то ни было ситуацией, тогда вопрос о его истинности или ложности не возникает. Точно так же, пока в формулу не подставлены значения ее аргументов, нет повода спрашивать, каково конкретное значение функции, выраженной этой формулой. Истинностное значение каждого предложения (т.е. его истинность или ложность) однозначно определяется ситуацией, к которой оно отнесено. Аналогично, значение любой формулы однозначно определяется набором значений всех входящих в нее аргументов.

Таким образом, каждое предложение выражает некоторую функцию с двоичными значениями, иначе говоря, задает какой-то предикат $P(x)=\xi$. Независимой переменной x этой функции служит переменная ситуация, зависимой – истинностная переменная ξ. После подстановки вместо переменной x конкретной постоянной ситуации x=a заданное предложение становится истинным ($\xi=1$) или ложным ($\xi=0$) в зависимости от того, соответствует или нет содержание этого предложения ситуации a, к которой оно отнесено. А что такое переменная ситуация x? Она должна представлять собой, в соответствии с приведенным выше определением понятия предиката, набор $x=(x_1, x_2,..., x_m)$ предметных переменных $x_1, x_2,..., x_m$. Любая постоянная ситуация x=a должна быть набором $a=(a_1, a_2,..., a_m)$ каких-то предметов $x_1=a_1, x_2=a_2,..., x_m=a_m$.

Итак, каждое предложение должно выражать некоторый предикат $P(x_1, x_2,..., x_m)=\xi$, представляющий зависимость истинностной переменной ξ от предметных переменных x_1, x_2,..., $x_{m.}$. Однако, если обратиться к конкретным предложениям естественного языка (например, русского), то никаких предметных переменных в них обнаружить не удается. Объясняется это тем, что предложение естественного языка, в отличие от математической формулы, выражает не всю функцию $P(x_1, x_2,..., x_m)$, а только ее имя Р. Каждый раз человек, преобразуя то или иное предложение в соответствующую ему мысль, достраивает его до предиката, добавляя к нему (как к имени предиката) недостающие предметные переменные. Только после этого предложение становится доступным для понимания. И наоборот, преобразуя некоторую мысль в предложение, человек исключает из нее предметные переменные, передавая другим людям не саму мысль, а только ее имя.

Покажем, как можно дополнить предложение предметными переменными. Пусть дано какое-нибудь предложение, например:

“На парковке стоит автомобиль”. (3.1)

Определяем число предметов, о которых идет речь в предложении (3.1). Очевидно, что таких предметов – два, один из них характеризуется словом “парковка”, а другой – словом “автомобиль”. Из множества V выбираем какие-нибудь две предметные переменные, например x_1 и x_2, и вводим их в предложение (3.1) после указанных слов. В результате получаем следующее утверждение:

“На парковке x_1 стоит автомобиль x_2”. (3.2)

Оно выражает теперь не только имя предиката, как исходное предложение (3.1), но и сам предикат с аргументами x_1 и x_2. Используя предложение (3.1) как имя полученного предиката, последний можем записать в виде:

$$\text{На парковке стоит автомобиль}(x_1, x_2). \tag{3.3}$$

Предложение, дополненное предметными переменными, будем называть высказыванием. Именно так в математической логике называется любое утверждение, выражающее какой-либо предикат. В нашем примере в роли высказывания используется запись (3.2). Предложение, выполняющее роль имени предиката и входящее в его состав, будем записывать жирным шрифтом, чтобы отличить его от исходного предложения, которое записывается нежирным (так сделано в записях (3.1) и (3.3)).

Добавим к предложению (3.1) еще одно:

“Рядом с ним стоит мотоцикл”, (3.4)

образуя из них единый текст. В предложении (3.4) речь идет тоже о двух предметах. Первый из них указан местоимением “ним”, второй – именем существительным “мотоцикл”. Из контекста (а контекстом для предложения (3.4) служит предложение (3.1)) явствует, что слово “ним” является заменителем слова “автомобилем”, относящегося к предмету x_2, который фигурирует в первом предложении. Слово “мотоцикл” вводит третий предмет, отличающийся от первых двух, что требует введения еще одной предметной переменной, в качестве которой берем x_3. В результате получаем высказывание

“Рядом с автомобилем x_2 стоит мотоцикл x_3”, (3.5)

которое выражает предикат

$$\text{Рядом с автомобилем стоит мотоцикл}(x_2, x_3). \tag{3.6}$$

Обратим внимание на тот важный факт, что предложение

“Рядом с автомобилем стоит мотоцикл”, (3.7)

использованное в качестве имени предиката (3.6), становится двусмысленным, если его рассматривать вне контекста и без предметных переменных x_2 и x_3. Теперь нельзя с уверенностью определить, о каком конкретно автомобиле в нем идет речь: о том же самом, что и в предложении (3.1) (т.е. о предмете x_2), или о каком-либо ином (например, о предмете x_4). Еще более многозначным воспринимается предложение (3.4). В нем слово “ним” может относиться к какому угодно предмету, а не только к автомобилю. Введением же предметных переменных подобные неоднозначности полностью устраняются. Этот факт наглядно демонстрирует необходимость дополнения предложений предметными переменными для возможности их однозначного понимания. При переходе от предложений (3.1) и (3.4) к соответствующим им высказываниям (3.2) и (3.5) мы опирались непосредственно на интуицию человека, являющегося носителем русского языка. Однако возможно и формальное выполнение такого перехода, который в этом случае должен осуществляться чисто механически только на основе анализа текста предложения и окружающего его контекста без обращения к их смыслу. Эта задача весьма сложна и, насколько нам известно, никем еще не рассматривалась, однако без ее решения невозможна автоматизация процесса понимания текстов естественного языка.

Обнаруживается следующее несоответствие принятой нами исходной теоретической схемы и фактического положения дел. Алгебраический подход к естественному языку требует, чтобы все предметные переменные x_1, x_2,..., x_m алгебры предикатов присутствовали в каждом предикате $P(x_1, x_2,..., x_m)$. А фактически это не так: в предикатах (3.3) и (3.6) присутствуют не все переменные. Так, в предикате (3.3) отсутствует переменная x_3, а в предикате (3.6) – переменная x_1. Однако точно такое же несоответствие наблюдается и в математике. Там тоже почти во всех используемых на практике формулах присутствует лишь небольшая часть переменных той алгебры, на языке которой они пишутся. В математике это несоответствие преодолевается введением понятия несущественной переменной. Этим понятием воспользуемся и мы для дальнейшего построения теории языка.

Аргумент x_{i} $(i=\overline{1,m})$ предиката $P(x_1, x_2,..., x_{\mathrm{i}},..., x_{\mathrm{m}})$ называется несущественным, если при любых $x_1, x_2,..., x_{\mathrm{i-1}}, x_{\mathrm{i}}', x_{\mathrm{i}}'', x_{\mathrm{i+1}},..., x_{\mathrm{m}} \in U$ $P(x_1, x_2,...,x_{\mathrm{i-1}}, x_{\mathrm{i}}', x_{\mathrm{i+1}},..., x_{\mathrm{m}})=P(x_1, x_2,..., x_{\mathrm{i-1}}, x_{\mathrm{i}}'', x_{\mathrm{i+1}},..., x_{\mathrm{m}})$. Согласно этому определению, значение предиката P не зависит от значения несущественного аргумента x_{i} при любых фиксированных значениях остальных переменных. Если в формуле предиката P переменная x_{i} отсутствует, это свидетельствует о том, что она для этого предиката несущественна. В перечне аргументов предиката $P(x_1, x_2,..., x_{\mathrm{m}})$ несущественные переменные можно опускать. Например, предикат $P(x_1, x_2,..., x_{\mathrm{m}})$, у которого существенны лишь переменные x_2 и x_4, можно записать в виде $P(x_2, x_4)$. Если в формуле предиката присутствует несущественный аргумент, то ее всегда можно так тождественно преобразовать, что он в ней исчезнет. Предикат, у которого все аргументы, кроме одного, несущественны, называется унарным, двух – бинарным, трех – тернарным, n $(n \le m)$ – n-арным. Число n называется арностью предиката. От него надо отличать число m, являющееся размерностью предиката. При m=1 предикат называется одноместным, при m=2 – двухместным и т.д. При произвольном m предикат называется m-местным. При $m \ge 2$ предикат $P(x_1, x_2,..., x_m)$ называется многоместным.

Каждый человек в своей речевой практике использует значительное число предметных переменных. Особенно это отчетливо ощущается при освоении больших по объему связных текстов, таких как роман Толстого “Война и мир” или трехтомный учебник Фихтенгольца по математическому анализу, когда приходится держать в уме одновременно большое количество действующих лиц и событий или понятий. По предварительным оценкам число предметных переменных в подобных случаях достигает многих сотен и даже тысяч. Отсюда следует, что число m, характеризующее количество всех предметных переменных в множестве V лингвистической алгебры, весьма велико. Для оценки его конкретной величины необходимы дополнительные исследования. Формально приходится считать, что каждое предложение, входящее в состав таких текстов, имеет все эти аргументы. Однако существенными из них в отдельных высказываниях всегда будут лишь немногие предметные переменные (обычно не более десятка). В приведенном выше примере текст состоит из двух предложений (3.1) и (3.4), в нем незримо присутствуют три предметные переменные. Существенными же в каждом из этих предложений выступают две переменные (в первом – x_1 и x_2, во втором – x_2 и x_3). Общей для обоих предложений является одна существенная переменная (x_2).

3.2. Естественный язык как булева алгебра

Рассмотрим теперь вопрос о базисных элементах лингвистической алгебры. Предложения строятся из отдельных слов. Поэтому естественно предположить, что в роли базисных элементов в лингвистической алгебре выступают слова. Поскольку любые элементы носителя лингвистической алгебры – предикаты, то и слова, рассматриваемые в качестве базисных элементов, тоже должны быть предикатами. Попытаемся показать, что это так и есть на самом деле. С содержательной стороны любые элементы лингвистической алгебры являются мыслями, а мысли выражаются предложениями. Следовательно, отдельные слова тоже надо рассматривать как предложения. Но возможно ли это? Приведем соображения в пользу положительного ответа на этот вопрос.

Начнем с имен существительных. Возьмем, к примеру, слово “книга”. Его можно употребить в роли предложения. Чтобы это показать, произведем следующий мысленный эксперимент. Исследователь предъявляет испытуемому слово “книга” и при этом указывает на предмет x, который выбирается им произвольно. Испытуемый должен определить, является ли предъявленный ему предмет x книгой или нет. Если испытуемый способен дать правильный ответ на предъявление любого предмета, то, тем самым, он демонстрирует знание смысла слова “книга”. И наоборот, тот, кто такой ответ дать не может, не проявляет полного знания смысла этого слова.

В описанном эксперименте испытуемый реализует предикат $P(x)$, выраженный высказыванием “Предмет x есть книга”. Его имя можно кратко записать одним словом P=“книга”. В данном употреблении слово “книга” играет роль целого предложения. Выразим предикат $P(x)$, реализуемый испытуемым в этом эксперименте, записью книга(x). В ней слово книга используется в роли имени P предиката $P(x)$, а переменная x – в роли его аргумента. Значениями переменной x служат предметы, предъявляемые испытуемому. Множество всевозможных предметов, на которые способен отреагировать испытуемый, играет роль универсума предметов U. Таким образом, каждое имя существительное P можно понимать как имя некоторого предиката $P(x)$, заданного на множестве всевозможных предметов U. Предикат этот реален и вполне определен, поскольку его может воспроизвести на практике любой человек, владеющий русским языком, отвечая на вопрос: “Подходит ли предмет x под понятие, выраженное именем существительным P?”. В приведенном выше примере этот вопрос будет выглядеть следующим образом: “Является ли предмет x книгой?”.

Аналогичные соображения применимы также и к словам, относящимся к другим частям речи. Возьмем, к примеру, имя прилагательное “большой”. Его можно понимать как предикат большой(x), выражаемый высказыванием “Предмет x – большой” (хотя бы в одной из возможных ролей: стола, стула и т.д.). Такое расширительное понимание смысла имен прилагательных представляется неизбежным, если исходить из того, что каждое слово, взятое само по себе (т.е. вне контекста), что-то означает. Языковая же интуиция человека ясно свидетельствует, что это так и есть. Предъявляя испытуемому, владеющему русским языком, предметы из множества U и предлагая ему ответить на вопрос “Большой ли предмет x?”, можно убедиться, что слову “большой” соответствует вполне определенный предикат и именно тот, о котором говорилось выше. То же относится и к любым другим именам прилагательным. С причастиями (например, “едущий”) и порядковыми числительными (например, “второй”) поступаем аналогично.

Переходим к глаголам. Берем, к примеру, слово “лежит”. Ему ставим в соответствие предикат лежит(x), выражаемый предложением “Предмет x лежит”. Количественные числительные, например, “два”, выражаем предикатом два(x), где аргумент x определен теперь уже не на множестве всех предметов U, а на системе имен всех подмножеств множества U. Испытуемый, реализующий предикат два(x), должен отвечать на вопрос: “Состоит ли множество x из двух предметов?”. Обращаемся к наречиям. Слово “темно” понимаем как предложение, относящееся к ситуации x. Испытуемый, отвечающий на вопрос “Темно ли в ситуации x?”, будет реализовать предикат темно(x). Слово «очень» выражает предикат очень(x), реализуемый испытуемым, которому предложено отвечать на вопрос “Обладает ли предмет x каким-либо свойством в высокой степени?”. Например, можно ли утверждать, что предмет x очень большой или очень пушистый и т.п. (далее следует перечисление всевозможных свойств предмета, доступных для понимания испытуемым). Предлоги также можно понимать как предикаты, но не унарные, как это было до сих пор, а бинарные. Например, предлог «на» понимаем как предикат на(x, y), соответствующий предложению “Предмет x находится на предмете y”. Мы предполагаем, что любое слово (за исключением небольшого количества слов, выражающих операции над предикатами, таких как “не”, “и”, “или”) можно представить подобным способом в виде некоторого предиката. Для прочного обоснования этой гипотезы необходимы специальные исследования в области анализа смысла слов.

Рассмотрим теперь вопрос о базисных операциях лингвистической алгебры. Из любого предложения можно образовать его отрицание, поставив перед ним частицу “не” или выражение “ложно, что”. Например, из фразы

“Идет дождь” образуем ее отрицание “Не идет дождь” (“Ложно, что идет дождь”). Отрицанием можно действовать также и на отдельные слова и словосочетания, например: “не стул”, “не синий”, “не очень”, “не два”, “не едет”, “не синий платок” и т.п. Естественно предположить, что с алгебраической точки зрения отрицание предложения Р – это булева операция отрицания предиката Р(x), выражающего содержание этого предложения. Обозначая операцию отрицания словом не, можем, к примеру, записать не(стол(x))=(не стол)(x).

В результате ее выполнения получаем новый предикат с именем не стол. В общем случае имеем

$$\text{не}(\mathrm{P}(x_1, x_2,..., x_m))=(\text{не}P)(x_1, x_2,..., x_m). \qquad (3.8)$$

Здесь $P(x_1, x_2,..., x_m)$ – произвольное высказывание; $x_1, x_2,..., x_m$ – его предметные переменные; P – предложение, соответствующее этому высказыванию; не(P) – предложение, получаемое из предложения P действием на него операции отрицания не.

Аналогичным образом рассматриваем союзы “и” и “или”, с помощью которых можно соединять любые предложения, получая в результате новые предложения (в общем случае – тексты). Естественно предположить, что слова и и или соответствуют двухместным операциям конъюнкции и дизъюнкции, действующим на высказывания $P(x_1, x_2,..., x_n)$ и $Q(x_1, x_2,..., x_n)$, которые выражают смысл предложений P и Q. Можем записать:

$$(P(x_1, x_2,..., x_m))\text{и}(Q(x_1, x_2,..., x_m))=(P\text{и}Q)(x_1, x_2,..., x_m); \qquad (3.9)$$

$$(P(x_1, x_2,..., x_m))\text{или}(Q(x_1, x_2,..., x_m))=(P\text{или}Q)(x_1, x_2,..., x_m), \qquad (3.10)$$

где P и Q – исходные предложения; PиQ и PилиQ – предложения, получаемые в результате соединения исходных предложений союзами “и” и “или”. Например, слова “стол” и “стул” превращаем в словосочетания “стол и стул”, “стол или стул”, предложения “Идет дождь” и “Светит солнце” – в предложения “Идет дождь и светит солнце”, “Идет дождь или светит солнце”.

Обратим внимание на возможность двоякого употребления союзов “и” и “или”. Словосочетание “стол и стул” можно понимать как предикат

$$(\text{стол}(x))\text{и}(\text{стул}(x)). \tag{3.11}$$

В этом случае имеется в виду, что один и тот же предмет x используется как в роли стола, так и в роли стула. Другое понимание дается высказыванием

$$(\text{стол}(x))\text{и}(\text{стул}(\text{y})), \tag{3.12}$$

которое выражает следующую мысль: «предмет x есть стол, а предмет у – стул». В первом случае речь шла об одном предмете, во втором – о двух. Такое же двойное понимание возможно и для словосочетания «стол или стул». Эти примеры наглядно показывают, что в результате алгебраизации естественного языка появляется возможность легко отвечать на вопросы, представляющиеся весьма трудными для традиционного анализа языка и речи. Присвоив предикату (3.12) имя стол и стул, приходим к следующему равенству: (стол(x))и(стул(y))=стол и стул(x, y).

Предикат (3.11) является производным от предиката (3.12), поскольку его можно получить из предиката (3.12), заменяя в нем переменную y на x: стол и стул(x, x)=(стол(x))и(стул(x)).

Введем понятие булевой алгебры предикатов. Пусть $P(x_1, x_2,..., x_m)$ и $Q(x_1, x_2,..., x_m)$ – предикаты на U. Отрицанием $\neg P$, конъюнкцией $P\vee Q$ и дизъюнкцией $P\wedge Q$ предикатов P и Q называются такие операции над предикатами, которые для любых $x_1, x_2,..., x_m \in U$ определяются равенствами:

$$(\neg P)(x_1, x_2,..., x_m)=\neg(P(x_1, x_2,..., x_m));$$

$$(P\wedge Q)(x_1, x_2,..., x_m)=P(x_1, x_2,..., x_m)\wedge Q(x_1, x_2,..., x_m);$$

$$(P\vee Q)(x_1, x_2,..., x_m)=P(x_1, x_2,..., x_m)\vee Q(x_1, x_2,..., x_m).$$

Этими равенствами операции $\neg$, $\wedge$ и $\vee$ над предикатами сводятся к операциям $\neg$, $\wedge$ и $\vee$ над их значениями, т.е. к операциям над логическими элементами 0 и 1. Последние называются отрицанием, дизъюнкцией и конъюнкцией логических элементов и определяются следующим образом: $\neg 0=1$, $\neg 1=0$; $0\wedge 0=0\wedge 1=1\wedge 0=0$, $1\wedge 1=1$; $0\vee 0=0$, $0\vee 1= =1\vee 0=1\vee 1=1$. Булевой алгеброй предикатов называется любая алгебра предикатов с базисом операций, состоящим из отрицания, конъюнкции и дизъюнкции предикатов.

Операции отрицания, конъюнкции и дизъюнкции называются булевыми. Выше мы определили булевы операции конструктивно, но их можно задать и абстрактно (т.е. системой свойств) с помощью понятия булевой алгебры. Булевой алгеброй называется любое множество M вместе с заданными на нем одноместной операцией $\neg$ и двухместными операциями $\wedge$ и $\vee$. По определению эти операции обладают следующими свойствами [68]: для любых x, y, $z\in M$ $x\wedge x=x$, $x\vee x=x$; $x\wedge y=y\wedge x$, $x\vee y=y\vee x$; $(x\wedge y)\wedge z=x\wedge(y\wedge z)$, $(x\vee y)\vee z=x\vee(y\vee z)$; $(x\vee y)\wedge z=(x\wedge z)\vee(y\wedge z)$, $(x\wedge y)\vee z=(x\vee z)\wedge(y\vee z)$; $x\vee(y\wedge\neg y)=x$, $x\wedge(y\vee\neg y)=x$; $\neg(\neg x)=x$; $\neg(x\vee y)=\neg x\wedge\neg y$, $\neg(x\wedge\ y)=\neg x\vee\neg y$. Эти свойства называются аксиомами булевой алгебры. При заданном M булевы операции в абстрактном смысле (т.е. с точностью до обозначений элементов множества M) определяются единственным образом. Важно отметить, что указанная система аксиом избыточна. В ней все аксиомы, кроме одной, – парные. В каждой паре одну из аксиом можно исключить (либо все левые, либо все правые) без ущерба для полноты системы. Таким образом, для исчерпывающей характеристики понятия булевой алгебры достаточно указать всего семь аксиом.

Вводим еще одну гипотезу: лингвистическая алгебра есть булева алгебра. В роли булевых операций $\neg$, $\wedge$ и $\vee$ в ней выступают операции над словами и словосочетаниями, предложениями и текстами, выраженные словами не, и и или. Не всегда используются именно эти слова для выражения указанных операций. Два предложения, на которые действует операция и, могут соединяться запятой или точкой, например, “Идет дождь, светит солнце”, “Идет дождь. Светит солнце”. Вместо союза “и” в роли конъюнкции могут использоваться соединительные слова “а”, “однако”, “тем не менее” и т.п., например: “Идет дождь, однако светит солнце”. Но слова “не”, “и” и “или” не всегда выражают операции отрицания, конъюнкции и дизъюнкции. Например, в предложении “Подождем, пока не пройдет дождь” частица “не” означает не отрицание, а утверждение. В предложении “Дождь идет и идет” союз “и” (вместе со вторым вхождением слова “идет”) выражает вовсе не операцию конъюнкции, а смысл слова “долго”, так что ту же мысль можно выразить фразой “Дождь идет долго”. Союз “или” может использоваться в разделительном смысле “или – или”, например: “Выбирай: он или я”. Как дизъюнкция союз “или” используется в объединительном смысле “или также”.

Обобщая, можно сказать, что в естественном языке тексты и соответствующие им смыслы не связаны взаимно однозначно. Смысл одного и того же текста может меняться в зависимости от выбора предметных переменных и от контекста. Важно заметить, что один и тот же смысл можно выразить различными текстами. Смыслы можно изучать и формально описывать вне связи с соответствующими им текстами, а тексты – вне связи с их смыслами. Кроме того, можно формально описывать связь между текстами и их смыслами. Смыслы можно записывать на языке высказываний, тексты же проще всего выражать непосредственно в их естественно-языковой форме. Для описания связи между текстами и смыслами, по-видимому, необходим специальный математический язык.

Имеются случаи, когда слова “не”, “и” и “или” выражают булевы операции над текстами, но при этом используются по-разному или же с дополнительным смыслом. Например, словосочетание “не яркое солнце” можно понимать двояко: как “(не яркое) солнце” в смысле “предмет x есть солнце, и он неярок” и как “не (яркое солнце)” в смысле “ложно, что предмет x есть яркое солнце”. Ясно, что указанные смыслы этого словосочетания различны. Далее, союз “и”, употребленный в роли конъюнкции, может выражать еще и противопоставление событий, как, например, в предложении “Дождь идет и солнце светит”. Следующий пример. Рассмотрим два предложения: “Джейн вышла замуж и родила ребенка” и “Джейн родила ребенка и вышла замуж”. По законам булевой алгебры конъюнкция коммутативна, а, следовательно, оба предложения должны иметь один и тот же смысл. Но очевидно, что это не так. Объясняется это противоречие тем, что в данном случае союз “и” выражает не только конъюнкцию высказываний, но еще и последовательность двух событий во времени, описываемых этими высказываниями.

3.3. Предложения как формулы

Из принятых выше гипотез следует, что предложения естественного языка должны быть формулами алгебры предикатов. Сопоставим этот вывод с фактами языка. Чтобы это сделать, вначале придется обратиться к языку математики. Любая формула имеет вполне определенную структуру, которую можно выразить в виде некоторой схемы. Возьмем, к примеру, формулу алгебры булевых функций $\bar{X}_1 X_2 \vee X_3 \bar{X}_4$. Ее можно выразить графически схемой, изображенной на рис. 3.1. Кружки со знаками булевых операций $\neg$, $\wedge$ и $\vee$ изображают преобразователи формул.

Схема синтезирует формулу $\bar{X}_1 X_2 \vee X_3 \bar{X}_4$ из ее аргументов X_1, X_2, X_3, X_4. Так, проходя через крайний справа блок дизъюнкции, формулы $\bar{X}_1 X_2$ и $X_3 \bar{X}_4$ преобразуются в формулу $\bar{X}_1 X_2 \vee X_3 \bar{X}_4$.

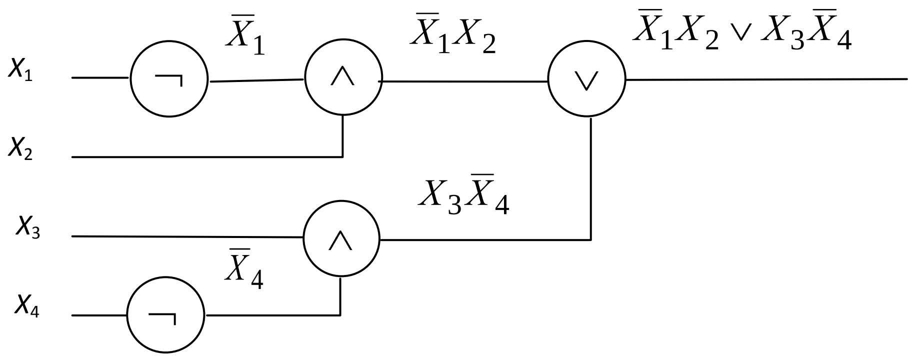

Рис. 3.1. Графическое представление формулы алгебры булевых функций

Та часть формулы, на которую бинарная операция ($\wedge$ или $\vee$) действует первой, поступает на преобразующий блок по горизонтальному входу, второй – по вертикальному. Схема формулы представляет собой древовидный граф.

Попытаемся подойти к разработке метода построения подобных графов для предложений естественного языка. В грамматике для наглядного представления структуры предложений используются деревья синтаксического подчинения [19]. Их мы и примем в качестве отправного пункта при решении поставленной задачи. Пример дерева синтаксического подчинения изображен на рис. 3.2.

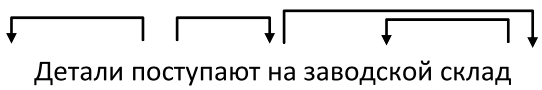

Рис. 3.2. Пример дерева синтаксического подчинения

Слова предложения соединяются в пары стрелками, называемыми дугами. Результат такого соединения называется словосочетанием. Слово, из которого дуга исходит, называется главным, а слово, в которое она входит – зависимым. Корнем предложения называется слово, в которое не входит ни одна из дуг. Подразумевается, что человек, понимающий смысл предъявленного ему предложения, способен построить для него дерево синтаксического подчинения. В основе этой способности лежит ощущение того, что зависимое слово каждой пары в процессе порождения предложения возникает в уме непосредственно за главным.

Построение деревьев синтаксического подчинения для большого числа предложений показало, что связи между словами всегда образуют древовидную структуру, аналогичную той, которая изображена на рис. 3.2. Этот факт свидетельствует о том, что в предложении, кроме линейного порядка слов, существуют еще и направленные связи между словами. Для подавляющего большинства предложений, используемых на практике, деревья синтаксического подчинения характеризуются тем, что дуги в них не пересекаются, а корень не лежит ни под одной из дуг. Такие деревья, как и соответствующие им предложения, называются проективными. В отличие от них, непроективные предложения воспринимаются как неестественные. Предложение, представленное на рис. 3.2, является проективным. На рис. 3.3 изображена структура непроективного предложения.

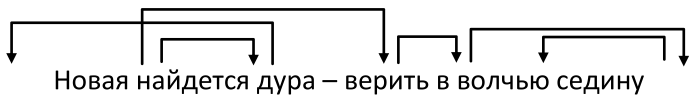

Рис. 3.3. Структура непроективного предложения

Для деловой прозы непроективные предложения обычно неприемлемы стилистически. Однако в поэтических текстах они встречаются часто.

Ниже описывается метод построения схемы формулы предложения. Начнем с конкретного предложения, представленного на рис. 3.2. Схема его формулы изображена на рис. 3.4. Значениями аргументов $X_1 \div X_5$ формулы служат слова “Детали”÷“склад” (точнее – их словоформы). Кружки, помеченные номерами, изображают преобразователи слов и словосочетаний. Они выполняют операции соединения слов и словосочетаний. Схема синтезирует предложение из отдельных слов.

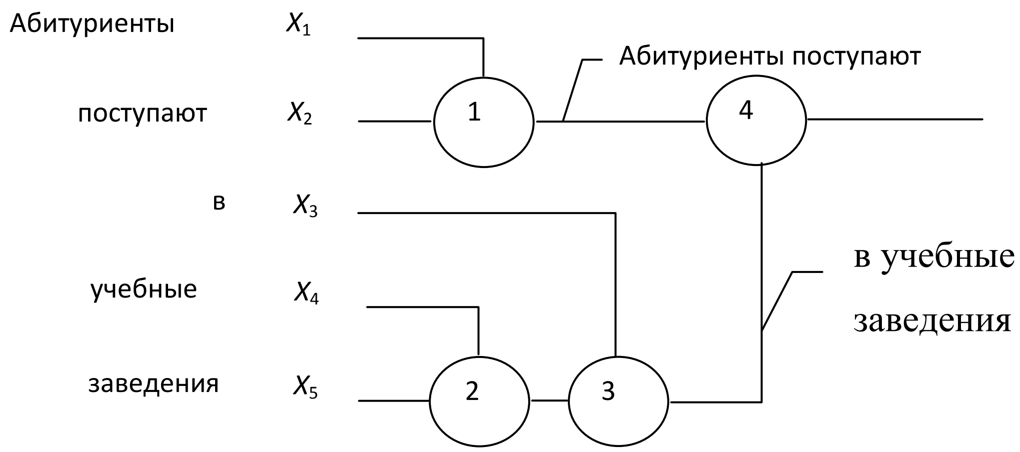

Рис. 3.4. Схема формулы предложения

Так, проходя через блок 1, слова “Абитуриенты” и “поступают” преобразуются в словосочетание “Абитуриенты поступают”. Блок 3 формирует словосочетание “в учебные заведения”. На выходе схемы (после блока 4) получаем готовое предложение “Абитуриенты поступают в учебные заведения”. Номера блоков указывают последовательность выполнения операций преобразователями схемы.

Эта последовательность определяется по следующему алгоритму. Движемся отдельными шагами вдоль предложения слева направо, перебирая слова и выполняя на каждом шаге одну за другой все возможные операции. При этом отступаем от слова, рассматриваемого на данном шаге (первичном), двигаясь по шагам (вторичным) назад и устанавливая каждый раз, соединяется ли слово, стоящее впереди, с тем, которое находится сзади него. В результате выполнения каждой операции должна получиться некая связная законченная последовательность слов (словосочетание). Некоторые из получаемых словосочетаний будут иметь вид предложений. Последние формируются операцией, расположенной на уровне корня предложения, в роли которого выступает сказуемое (в случае, когда оно имеется в наличии; если же его нет, то корнем предложения становится подлежащее). Главное слово в каждой паре считается первым вне зависимости от того, расположено оно впереди зависимого слова или позади него. Следует оговориться, что приведенный выше алгоритм несовершенен. Он не учитывает всех возможных вариантов предложений и поэтому его нельзя применить к любому из них. Кроме того, в нем не достигнута полная формализация действий, поскольку при выполнении алгоритма существенно используется интуиция человека – носителя языка. Необходимо проведение дополнительных исследований для доводки данного алгоритма до такого вида, чтобы с его помощью ЭВМ сама смогла строить схему формулы произвольного предложения.

Применяем только что приведенный метод к рассматриваемому примеру. На первом шаге обращаемся к слову “Абитуриенты”, здесь никакой операции выполнить не удается. На втором шаге обращаемся к слову “поступают”. С помощью операции 1 образуем предложение “Абитуриенты поступают”. Операцию 1 помещаем на линии сказуемого “поступают”. Слово “поступают” считаем первым аргументом операции, а слово “абитуриенты” – вторым. На третьем шаге обращаемся к слову “в”, которое никуда присоединить не удается. То же происходит и на четвертом шаге со словом “учебные”. На пятом шаге к слову “заведения” присоединяем слово “учебные” с помощью операции 2, образуя словосочетание “учебные заведения”. Далее, с помощью операции 3 присоединяем слово “в” к словосочетанию “учебные заведения”, в результате получаем словосочетание “в учебные заведения”. Наконец, посредством операции 4 присоединяем полученное словосочетание к предложению “Абитуриенты поступают”. Результатом будет искомое предложение “Абитуриенты поступают в учебные заведения”.

Схема формулы предложения построена по его дереву синтаксического подчинения, так что, при желании, всегда можно возвратиться от схемы к дереву (т.е. перейти от рис. 3.4 к рис. 3.2). Однако схема содержит в себе и нечто новое, а именно: блоки, синтезирующие текст предложения из его отдельных элементов; полюсы, на которых появляются предложения и словосочетания; очередность выполнения синтеза формулы предложения блоками схемы. По схеме можно построить формулу предложения. В рассматриваемом примере она будет иметь следующий вид:

(Абитуриенты(поступают))4(в3(учебные2(заведения))). (3.13)

Номера выполняют в формуле роль имен операций, скобки указывают очередность их выполнения и последовательность применения каждой операции к словам, а формы слов представляют собой значения аргументов формулы. Перейдем к аргументам формулы от их значений, заменяя слово “Абитуриенты” на переменную X_1, слово “поступают” – на X_2, “в” – X_3, “учебные” – X_4, “заведения” – X_5. В результате формула запишется в виде:

$$(X_1 1(X_2))4(X_3 3(X_4 2(X_5))). \quad (3.14)$$

В развиваемой здесь теории естественного языка принято, что отдельные слова выражают предикаты. Отсюда следует, что символы $X_1 \div X_5$ выражают предикатные переменные, номера $1 \div 4$ – предикатные операции, а само выражение (3.14) – формулу алгебры предикатных операций. Формула же (3.13) выражает имя предиката предложения. Важно отметить, что любое непроективное предложение в математическом плане характеризуется тем, что его формулу невозможно записать без перестановки слов. Для проективного же предложения это сделать всегда возможно. Например, записывая формулу непроективного предложения, представленного на рис.3.3, придется предварительно переставить некоторые из его слов:

(Найдется2(новая1(дура)))6(верить5(в4(волчью3(седину)))).

После дополнения предложения предметными переменными по методике, описанной выше, оно превращается в формулу алгебры предикатов. Итак, мы видим, что естественный язык имеет двухъярусное строение. Первый ярус представлен некоторой алгеброй предикатов, второй – алгеброй предикатных операций. Семантика предложения, т.е. его содержание, формально описывается на языке алгебры предикатов, синтаксис, т.е. строение предложения, – на языке алгебры предикатных операций. Формула (3.14) показывает, в какой последовательности и из каких слов (не важно – каких) образуется предложение типа “Абитуриенты поступают в учебные заведения”.

Более сложный пример схемы формулы предложения представлен на рис. 3.5.

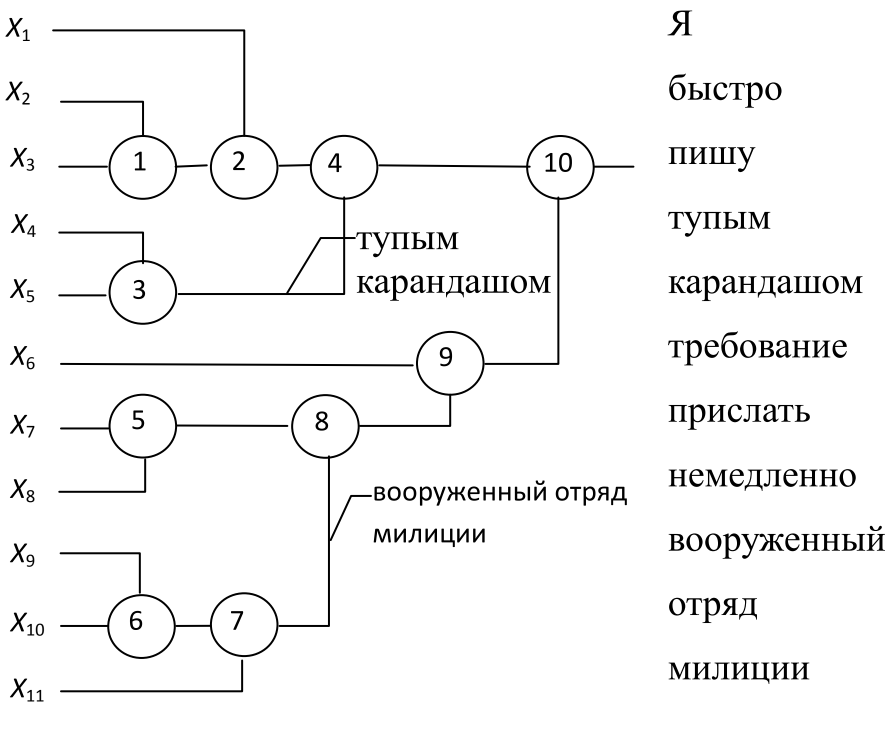

Рис. 3.5. Схема формулы предложения

Переходя от схемы к формуле, получаем следующую формулу алгебры предикатов, выражающую смысл предложения:

$$((\text{Я}2(\text{быстро}1(\text{пишу})))4(\text{тупым}3(\text{карандашом})))10$$
$$(\text{требование}9((\text{прислать}5\text{немедленно})8 \qquad (3.15)$$
$$((\text{вооруженный}6(\text{отряд}))7\text{милиции}))).$$

Заменяя в выражении (3.15) слова “Я”÷“милиции” соответствующими им предикатными переменными $X_1 \div X_{11}$, приходим к формуле алгебры предикатных операций

$$((X_1 2(X_2 1(X_3)))4(X_4 3(X_5)))10(X_6 9((X_7 5X_8)8((X_9 6(X_{10}))7X_{11})))), \quad (3.16)$$

выражающей синтаксическую структуру рассматриваемого предложения.

Только что при рассмотрении синтаксической структуры предложения нам пришлось обратиться к понятию алгебры предикатных операций. Дадим его формальное определение. Пусть U – универсум предметов; $x_1, x_2, ..., x_m$ – предметные переменные; P – множество всех предикатов $P(x_1, x_2, ..., x_m)$ на предметном пространстве U^m. Множество P называется универсумом предикатов. Переменные $X_1, X_2, ..., X_k$, определенные на множестве P, называются предикатными. Их значениями служат предикаты, заданные на U^m. Множество P^k называется предикатным пространством размерности k над предметным пространством U^m. Элементы множества P^k (k-компонентные наборы предикатов) называются предикатными векторами. Предикатное пространство представляет собой двухэтажную конструкцию: на ее первом этаже находятся предметы, на втором – предикаты. Любая функция $F(X_1, X_2, ..., X_k)=Y$, отображающая множество P^k в множество P, называется предикатной операцией. Образуем множество R всех предикатных операций. Алгеброй предикатных операций над R называется любая алгебра, заданная на носителе R.

Пусть $F(X_1, X_2, ..., X_k)=$ – предикатная операция, отображающая множество P^k в множество P. Здесь $X_1, X_2, ..., X_k$ – предикатные переменные, выступающие в роли аргументов операции F; Y – предикатная переменная, являющаяся значением операции F. Отрицанием $\neg F=\overline{F}$ предикатной операции F называется такая предикатная операция, значения которой определяются по правилу $(\neg F)(X_1, X_2, ..., X_k)=\neg F(X_1, X_2, ..., X_k)$ для любых $X_1, X_2, ..., X_k \in P$. Пусть F и G – предикатные операции, отображающие P^k в P.

Дизъюнкцией $F \vee G$ предикатных операций F и G называется предикатная операция, значения которой определяются по правилу $(F \vee G)(X_1, X_2,..., X_k)=F(X_1, X_2,..., X_k) \vee G(X_1, X_2,..., X_k)$ для любых $X_1, X_2, ..., X_k \in M$.

Конъюнкцией $F \wedge G$ предикатных операций F и G называется предикатная операция, значения которой определяются по правилу $(F \wedge G)(X_1, X_2,..., X_k)=F(X_1, X_2,..., X_k) \wedge G(X_1, X_2,..., X_k)$ для любых $X_1, X_2, ..., X_k \in M$.

В последних трех равенствах слева от знака равенства фигурируют операции $\neg$, $\vee$, $\wedge$ над предикатными операциями; справа знаки $\neg$, $\vee$, $\wedge$ обозначают операции над предикатами. Булевой алгеброй предикатных операций называется любая алгебра предикатных операций с базисом операций, состоящим из отрицания, конъюнкции и дизъюнкции.

3.4. Словосочетания как формулы

Выше было выяснено, что предложения строятся из отдельных слов с помощью булевых операций и операций соединения слов и словосочетаний. Последние были лишь обозначены номерами, а о существе этих операций еще ничего не было сказано. Рассмотрим теперь вопрос о том, что конкретно представляют собой операции соединения слов и словосочетаний.

Принимая в роли сказуемого глагол, представим его как некий предикат, для чего дополним его предметными переменными. Предметные переменные эффективно выявляются с помощью вопросов, которые порождаются сказуемым. Например, в предложении “Абитуриенты поступают в учебные заведения” слово “поступают” порождает вопросы: 1) Кто поступает? – Абитуриенты; 2) Куда поступают? – В учебные заведения. В предложении “Я быстро пишу тупым карандашом требование прислать немедленно вооруженный отряд милиции” слово “пишу” порождает вопросы:

1) Кто пишет? – Я; 2) Как пишет? – Быстро; 3) Чем пишет? – Тупым карандашом; 4) Что пишет? – Требование прислать немедленно вооруженный отряд милиции. В соответствии с этим заключаем, что слово “поступают” в первом предложении представляет собой имя предиката поступают(x_1, x_2) с двумя предметными переменными, слово “пишу” – имя предиката пишу(y_1, y_2, y_3, y_4) с четырьмя предметными переменными. Таким образом, вопросы характеризуют собой отдельные предметные переменные предиката, выраженного сказуемым данного предложения.

Глагол в роли сказуемого предложения характеризуется, как никакая другая часть речи, большим числом предметных переменных. Вследствие этого он выполняет роль соединителя отдельных частей предложения в единое целое. Каждое предложение выражает некую мысль, которую можно выразить вполне определенным многоместным предикатом $P(x_1, x_2,..., x_n)$. Механизм образования предложения представляем следующим образом. Переменные x_1, x_2,..., x_n предложения сначала связываются сказуемым, реализующим предикат $S(x_1, x_2,..., x_n)$, где n – число предметных переменных при глаголе S. Дополняем предложение теми или иными словосочетаниями, отвечающими на вопросы, соответствующие предметным переменным x_1, x_2,..., x_n. Этим достигается постепенное приближение смысла текста предложения к требуемому.

Аналогично этому, расширением текста предложения достигается сужение (уточнение) его смысла и доведение его до того содержания, которое намеревался передать человек своей фразой. Может создаться впечатление, что с расширением текста предложения расширяется и его содержание, но это не так. На самом же деле, после присоединения к предложению каждого последующего слова его содержание всегда сужается, например: “Абитуриенты поступают” $\supseteq$ “Абитуриенты поступают в заведения” $\supseteq$ “Абитуриенты поступают в учебные заведения”. Если же двигаться в обратном направлении, отсоединяя одно за другим слова от готового предложения, то мы получим цепочку вложенных друг в друга предложений с расширяющимся содержанием, например: “Я быстро пишу тупым карандашом требование прислать немедленно вооруженный отряд милиции”$\subseteq$“Я быстро пишу тупым карандашом требование прислать отряд милиции”$\subseteq$“Я пишу требование прислать отряд”$\subseteq$“Я пишу требование”$\subseteq$“Я пишу”.

Из этого свойства логически следует, что соединение слов выражается операцией конъюнкции. Действительно, если известно, что предикаты P и Q находятся в отношении $P\subseteq Q$, то всегда найдется такой предикат R, что $P=QR$. Присоединяемое к предложению Q слово или словосочетание R выполняет, таким образом, роль конъюнктивного множителя. Наоборот, в результате выполнения операции $QR=P$ присоединения слова или словосочетания R к предложению Q получаем предложение P, удовлетворяющее условию $P\subseteq Q$.

Пусть $T_1(x_{i_{11}}, x_{i_{21}},..., x_{i_{s_1 1}})$, $T_2(x_{i_{12}}, x_{i_{22}},..., x_{i_{s_2 2}})$, ..., $T_r(x_{i_{1r}}, x_{i_{2r}},..., x_{i_{s_r r}})$ – предикаты, выражаемые словами (или словосочетаниями) предложения, которые присоединяются к его сказуемому. Здесь r – число всех слов (или словосочетаний), присоединяемых к сказуемому предложения. Аргументами каждого из предикатов T_j $(j=\overline{1,r})$ служат некоторые из аргументов предиката $S(x_1, x_2,..., x_n)$ сказуемого S. Символом s_j обозначено число существенных переменных предиката T_j. Тогда предикат предложения выразится в виде:

$$P(x_1, x_2,..., x_n)=S(x_1, x_2,..., x_n)T_1(x_{i_{11}}, x_{i_{21}},..., x_{i_{s_1 1}})\wedge$$
$$\wedge T_2(x_{i_{12}}, x_{i_{22}},..., x_{i_{s_2 2}})...T_r(x_{i_{1r}}, x_{i_{2r}},..., x_{i_{s_r r}}). \quad (3.17)$$

Равенство (3.17) показывает, как содержание предложения P образуется из содержания сказуемого S путем ограничения его содержанием словосочетаний T_1, T_2,...,T_r. Расчленяя мысль P на части S, T_1, T_2,..., T_r в соответствии с формулой (3.17), говорящий, по существу, производит конъюнктивную декомпозицию предиката P. А слушающий осуществляет композицию предиката P из предикатов S, T_1, T_2,..., T_r, выполняя операцию их конъюнкции. Для примера возьмем предикат (в) предложения “На столе лежит книга”. В соответствии с равенством (3.17) его можно представить следующей формулой:

$$\text{На столе лежит книга}(x_1, x_2)= \\ =\text{На}(x_1, x_2)\wedge\text{столе}(x_1)\wedge\text{лежит}(x_1, x_2)\wedge\text{книга}(x_2). \quad (3.18)$$

Ее содержание можно выразить высказыванием: “На предмете x_1 располагается предмет x_2, и предмет x_1 есть стол, и предмет x_2 находится в лежачем положении относительно предмета x_1, и предмет x_2 есть книга”. Конъюнктивной декомпозицией предиката P называется его представление в виде P=$Q_1\wedge Q_2\wedge...\wedge Q_l$, где Q_1, Q_2,..., Q_l – некоторые предикаты; l – число предикатов, получаемых в результате декомпозиции.

Обратим внимание на следующее важное обстоятельство. В предложении, каким бы обширным оно ни было, обычно даются ответы далеко не на все те вопросы, которые потенциально содержатся в сказуемом. Например, восприняв предложение “Детали поступают на заводской склад”, человек может задать множество вопросов, на которые он не получил ответа из данного предложения. К ним относятся: “Откуда поступают детали?”, “Когда они поступают?”, “С какой целью?” и т.п. Вернее, ответы на поставленные вопросы в предложении имеются, но они неинформативны (бессодержательны): “Откуда угодно”, “Когда угодно”, “С любой целью”. Это означает, что предметные переменные, соответствующие указанным вопросам, в данном предложении несущественны. В языке всегда имеются возможности для конкретизации этих ответов посредством расширения текста предложения. Управление этим расширением осуществляется постановкой соответствующих вопросов. После формирования конкретных ответов на поставленные вопросы соответствующие им предметные переменные в расширенном предложении превращаются из несущественных в существенные.

Выше мы рассмотрели, как из словосочетаний образуется предложение. Теперь рассмотрим конкретные способы образования словосочетаний из отдельных слов. Тема эта обширна и требует специального исследования. Здесь мы ограничимся лишь несколькими характерными примерами. Подлежащее T соединяется со сказуемым S обычно по схеме

$$P(x)=T(x)\wedge S(x), \tag{3.19}$$

в результате получается предложение P. Например,

$$\text{книга лежит}(x)=\text{книга}(x)\wedge\text{лежит}(x). \tag{3.20}$$

Формула, стоящая справа от знака равенства в выражении (3.20), означает: “Предмет x есть книга, и этот предмет лежит”. Согласование имени прилагательного T_1 с именем существительным T_2 осуществляется по аналогичной схеме:

$$T(x)=T_1(x)\wedge T_2(x). \tag{3.21}$$

В результате получаем словосочетание T. Например,

$$\text{толстая книга}(x)=\text{толстая}(x)\wedge\text{книга}(x). \tag{3.22}$$

Формула, стоящая справа от знака равенства в выражении (3.22), означает: “Предмет x – толстый, и этот предмет есть книга”.

Управление одного имени существительного T_2 другим именем существительным T_1 часто осуществляется по схеме:

$$T(x)=T_1(x)\wedge\exists y(T_2(y)\wedge\text{деталь}(x, y)). \qquad (3.23)$$

В результате получаем словосочетание T. Предикат деталь(x, y) имеет следующее содержание: “Предмет x является деталью (составной частью) предмета y”. К примеру, по этой схеме получаем:

$$\text{страница книги}(x)=\text{страница}(x)\wedge\exists y(\text{книга}(y)\wedge\text{деталь}(x, y)). \qquad (24)$$

Смысл правой части равенства (3.24) выражается высказыванием: “Предмет x есть страница, и существует предмет y, являющийся книгой, такой что предмет x служит его деталью”. Управление имени существительного T_2 количественным числительным T_1 может осуществляться по схеме:

$$T(x)=T_1(x)\wedge\forall y(y\in x\supset T_2(y)). \qquad (3.25)$$

Например,

$$\text{две книги}(x)=\text{два}(x)\wedge\forall y(y\in x\supset \text{книга}(y)). \qquad (3.26)$$

Содержание правой части равенства (3.26) можно передать следующим высказыванием: “Множество x состоит из двух предметов, и каждый из предметов y, если он принадлежит множеству x, есть книга”.

Выполненный выше анализ естественного языка как некоторой алгебры, называемой нами лингвистической, свидетельствует о том, что такой подход предоставляет исследователю определенные возможности для формального описания механизма интеллекта. Обнаруживается, что смысл текста можно представить в виде формулы алгебры предикатов, а его синтаксическую структуру – в виде формулы алгебры предикатных операций. Открываются перспективы для дальнейшего проникновения в механизм естественного языка. В частности, представляются перспективными исследования по выявлению различных видов предметных переменных предложения (например, временнóго и пространственного характера) и изучению способов их практического использования в языке. Нуждаются в изучении механизмы выражения смысла отдельных слов сочетаниями слов и сочетаниями морфов. Есть основания предположить, что и эти механизмы подчиняются зависимости (3.17). Важно установить, как изменяется смысл слова при переходе к той или иной его словоформе (например, “стул” – “стула”), а также при образовании из него нового слова (например, “синий” – “синева”). Особую задачу составляет изучение смысла отдельных морфов слова.

Важными объектами формального описания в языке являются анафора, эллипсис, омонимия и синонимия. При алгебраическом подходе к языку смысл этих, во многом пока загадочных, явлений становится достаточно ясным. Требует формального описания связь текста с его смыслом, смысловое взаимодействие текста и контекста, аксиоматическое выражение смысла первичных слов (категорий), механизм смысловой декомпозиции предложения на составные части при синтезе и его композиции из морфов, слов и словосочетаний при анализе предложений. Интересна задача выявления тождеств и схем тождеств (законов) лингвистической алгебры. Важно понять, каков механизм участия естественного языка при узнавании (зрительном, слуховом, осязательном и т.п.) предметов внешнего мира. Представляется, что нет другой такой области знания, которая в большей степени, чем эта, могла бы способствовать выяснению природы человека как сознательного существа и повышению темпов информатизации общества.

3.5 Лингвистическая интерпретация алгебры одноместных предикатов

Алгебра одноместных предикатов стоит особо в ряду алгебр предикатов, поскольку для достижения ее полноты достаточно единственной базисной операции дизъюнкции. Введем в рассмотрение необходимые определения. Дизъюнктивной алгеброй предикатов называется любая алгебра многоместных предикатов [1] типа $P(x_1, x_2, \ldots, x_m)$, $x_i \in A_i, i = \overline{1,m}$, базис операций которой образует единственная операция дизъюнкции, а базис элементов состоит из предикатов 0, 1 и всевозможных предикатов узнавания предмета x_i^a ($a \in A_i$). Любая дизъюнктивная алгебра многоместных предикатов ($m \geq 2$) неполна. Дизъюнктивная же алгебра одноместных предикатов типа $P(x)$, $x \in A$ полна при любом A. В этой алгебре каждый ненулевой предикат выражается формулой

$$P(x) = \bigvee_{a \in P} x^a, \tag{3.27}$$

кроме того, каждый предикат (в том числе и нулевой) выражается формулой :

$$P(x) = \bigvee_{a \in A} P(a) x^a. \tag{3.28}$$

Формулой дизъюнктивной алгебры одноместных предикатов можно записать любое множество. Пусть, например, $P = \{a, b, c\}$. Тогда $P(x) = x^a \vee x^b \vee x^c$. Множество P выражаем равенством $P(x) = 1$: $x^a \vee x^b \vee x^c = 1$; $x \in \{a, b, c\}$. Перечислим основные законы дизъюнктивной алгебры предикатов: идемпотентности $P \vee P = P$; коммутативности $P \vee Q = Q \vee P$; ассоциативности $(P \vee Q) \vee R = P \vee (Q \vee R)$; сохранения единицы $P \vee 1 = 1$; исключения нуля $P \vee 0 = P$; истинности $\underset{a \in A}{\vee} x^a = 1$. Система всех основных законов дизъюнктивной алгебры одноместных предикатов полна при любом A. Основываясь на этих законах можно доказать, к примеру, что $\{a,a,b\} = \{a,b\}$, $\{a,b\} = \{b,a\}$, $\{a,b\} \cup \{c\} = \{a,c\} \cup \{b,c\}$, $\varnothing \cup \{a,b\} = \{a,b\}$. Чтобы сделать это, составляем соответствующие доказываемым соотношениям равенства алгебры одноместных предикатов:

$$x^a \vee x^a \vee x^b = x^a \vee x^b,\ x^a \vee x^b = x^b \vee x^a,$$

$$(x^a \vee x^b) \vee x^c = (x^a \vee x^c) \vee (x^b \vee x^c),$$

$$0 \vee (x^a \vee x^b) = x^a \vee x^b$$

и устанавливаем, что эти равенства являются ее тождествами.

Можно консервативно расширить дизъюнктивную алгебру одноместных предикатов, дополнительно введя в ее базис операцию конъюнкции. Благодаря этому, получаем избыток в базисе вновь сформированной дизъюнктивно-конъюнктивной алгебры одноместных предикатов. Избавиться от этого избытка можно за счет консервативного сужения базиса элементов полученной алгебры. Кроме того, можно также консервативно расширить дизъюнктивно-конъюнктивную алгебру одноместных предикатов, дополнительно введя в ее базис операцию отрицания. Благодаря этому, мы получим булеву алгебру одноместных предикатов с избыточным базисом. Избавиться от этого избытка можно за счет консервативного сужения базиса элементов в этой алгебре.

Практическое значение алгебры одноместных предикатов определяется прежде всего тем, что она может выполнить роль ключа, открывающего доступ к системному формальному описанию и воспроизведению на ЭВМ механизма естественного языка, на котором люди общаются друг с другом.

Это утверждение попытаемся обосновать на примере русского языка. Как известно, основу любой разновидности естественного языка (в том числе – и русского) составляют слова, выражающие определенные понятия, такие, к примеру, как *стол, ручка*, *прическа*, *теорема*, *едет*, *синий*, *очень*, *давно*. Назовем их понятийными словами. Каждое из понятийных слов распространяется на свое множество конкретных предметов. Именно этим множеством определяется объем и *содержание* каждого понятийного слова, а в конечном счете, его *смысл* и значение. Преимущественно из понятийных слов составляются предложения, которыми люди выражают свои мысли. В предложениях могут встречаться слова, которые не относятся к понятийным словам, но таких слов в любом языке гораздо меньше, чем понятийных слов. Примерами не понятийных слов могут служить предлоги и союзы.

Встречаются предложения, составленные только из понятийных слов. Приведем пример такого предложения: *Я быстро пишу тупым карандашом требование прислать немедленно вооруженный отряд милиции*. На рис. 3.6 изображено *дерево грамматических зависимостей* [2] для этого предложения.

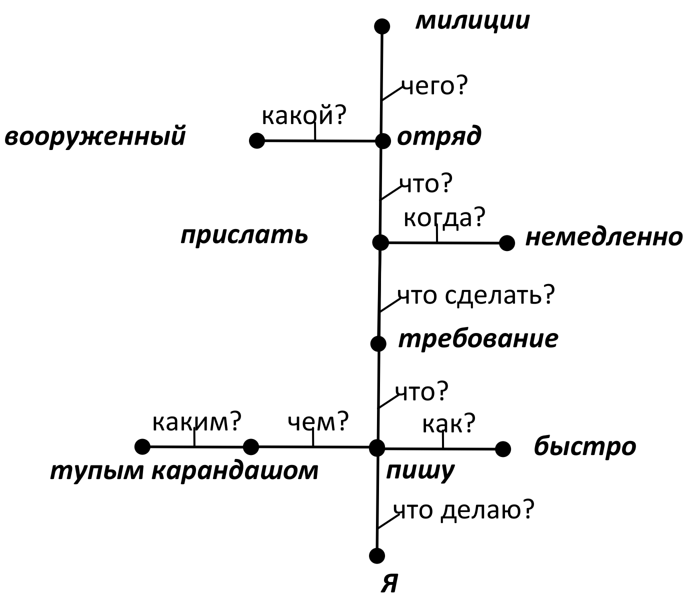

Рис. 3.6. Дерево грамматических зависимостей

С формальной точки зрения это дерево представляет собой реляционную сеть, которая состоит из полюсов и ветвей. Каждому полюсу соответствует свое слово, каждой ветви отвечает своя связь между соответствующими словами, которая характеризуется вопросами (рис. 3.6). Каждому понятийному слову соответствует свое множество предметов, а каждому множеству – свой одноместный предикат.

Таким образом, каждому понятийному слову можно поставить во взаимно однозначное соответствие свой одноместный предикат. Этот одноместный предикат можно принять за полноценное описание смысла соответствующего слова [3]. Убедиться в справедливости этого утверждения можно с помощью следующего эксперимента, который ставится на человеке, называемом испытуемым. В этом эксперименте для каждого понятийного слова (для конкретного испытуемого носителя языка) отыскивается соответствующий ему одноместный предикат (если таковой вообще существует). Для этого исследователь должен образовать множество U всех предметов, охватываемых содержанием любых понятийных слов, которые используются в естественном языке. Кроме того, исследователь формирует множество M всех понятийных слов той же разновидности естественного языка, которой владеет испытуемый (например, русским языком).

Далее исследователь предъявляет испытуемому по очереди из множества U предметы, спрашивая, знакомы ли они ему. Из знакомых для испытуемого предметов исследователь образует множество $A \subseteq U$. Он также предъявляет испытуемому слова из множества M, обращаясь с тем же вопросом. Из знакомых для испытуемого слов исследователь образует множество $B \subseteq U$. Далее, он образует декартово произведение $A \times B$ пар (a, b) и начинает предъявлять эти пары испытуемому, предлагая ему ответить на вопрос, можно ли считать, что предмет a охватывается содержанием слова b. Если испытуемый правильно узнаёт предметы $a \in A$ и в совершенстве понимает смысл слов $b \in B$, то он своими ответами будет реализовывать любые предикаты, заданные на $A \times B$.

Предметы, принадлежащие множеству A, всегда можно однозначно охарактеризовать словесным описанием определенной длины, например словосочетанием: "Я, быстро пишущий тупым карандашом требование прислать немедленно вооруженный отряд милиции". Присоединение каждого нового слова к словосочетанию очень быстро сужает его объем до одного предмета. Для достижения этого обычно бывает достаточно одного словосочетания сравнительно небольшой длины. Если же испытуемый не владеет в совершенстве естественным языком, то он неизбежно будет давать противоположные ответы при повторном предъявлении одной и той же пары "предмет-слово" или же вообще не сформирует никакого ответа. Этим он продемонстрирует, что не является носителем полноценного предиката каждого слова данного языка. Отсюда следует непреложный вывод: предикаты понятийных слов полноценно формально описывают содержательную сторону слова (его объем, содержание, смысл, значение).

Для каждого слова из множества B исследователь образует некоторое свое высказывание. Если это, к примеру, имя существительное, например, слово *стол*, он ставит ему в соответствие высказывание "Предмет x есть стол", если глагол (*едет*) – высказывание "Предмет x едет", если имя прилагательное (*красный*) – "Предмет x красный". Для других частей речи (*темно*, *быстро*, *очень*) высказывания тоже можно построить, однако несколько более сложным образом. Затем исследователь знакомит с этим высказыванием испытуемого (например, "Предмет x есть стол") и по очереди предъявляет ему различные предметы из множества A (например, "стул, стоящий в углу комнаты", "стол, стоящий посередине комнаты", "тетрадь, лежащая на тумбочке"). Он ставит перед испытуемым задание определить, будет ли предъявленное высказывание истинным или ложным для тех или иных предметов. Своим чисто внешним (объективным) поведением испытуемый реализует предикат $P(x)$, соответствующий значению слова *стол*. Мы его будем записывать в виде стол(x), принимая само слово за имя предиката $b(x)$.

Эксперименты показывают, что высказывание "предмет x есть стол или стул" соответствует дизъюнкции предикатов стол(x) и стул(x):

$$(\text{стол или стул})(x)=\text{стол}(x)\vee\text{стул}(x)=(\text{стол}\vee\text{стул})(x).$$

Аналогично находим:

$$(\text{стол и стул})(x)=\text{стол}(x)\wedge\text{стул}(x)=(\text{стол}\wedge\text{стул})(x),$$

$$(\text{не стол})(x)=\neg\text{стол}(x)=(\neg\text{стол})(x).$$

Мы видим, что на множестве A понятийных слов заданы булевы операции, то есть мы имеем *булеву алгебру слов*. Базисными элементами в ней служат предикаты, соответствующие понятийным словам.

Описанные опыты можно проводить не только со словами, но и со словосочетаниями. Берем, к примеру, словосочетание *большой стул*. Опыты со словами *большой*, *стул* и словосочетанием *большой стул* приводят к тождеству:

$$(\text{большой стул})(x)=\text{большой}(x)\wedge\text{стул}(x)$$

В результате появляется возможность приступить к практически беспрепятственному системному математическому описанию механизма естественного языка.

4 РАЗРАБОТКА КОНЦЕПЦИИ ПАРАЛЛЕЛЬНОЙ АССОЦИАТИВНОЙ ОБРАБОТКИ СИМВОЛЬНОЙ ИНФОРМАЦИИ

Описана концепция параллельной обработки символьной информации, сущность которой заключается в обращении к механизмам ассоциативной обработки информации, родственным процессам анализа и синтеза, протекающим в мозге человека. На примере реляционной сети склонения полных непритяжательных имен прилагательных русского языка построена логическая ассоциативная структура, представленная в виде системы априорных отношений.

4.1 Разработка ассоциативной структуры для аппаратной реализации реляционной сети

В настоящее время актуальной является задача повышения эффективности работы систем обработки естественно-языковой информации за счет аппаратной реализации реляционных сетей памятью с ассоциативным доступом, применение которой позволяет существенно повысить скорость выборки и упростить доступ к данным. Так как устройство с ассоциативной памятью предназначено для повышения скорости доступа при работе с базами данных, то наиболее целесообразно выполнить его в виде отдельной платы расширения для компьютера. Впоследствии на основании этой платы может быть создан сопроцессор данных.

Предлагается рассмотреть процесс аппаратной реализации памятью с ассоциативным доступом на примере разработанной методами алгебры конечных предикатов и предикатных операций [12, 15] модели реляционной сети склонения полных непритяжательных имен прилагательных русского языка [9]. Строение данной реляционной сети приведено на рис. 4.1

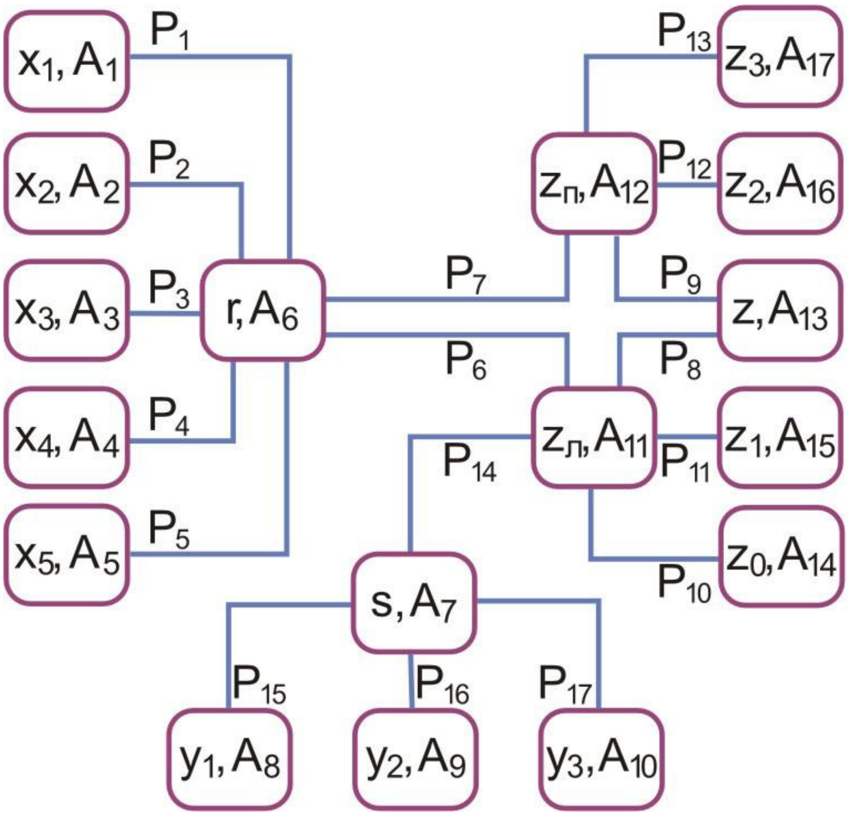

Рис. 4.1. Строение модели реляционной сети склонения полных непритяжательных имен прилагательных русского языка

Каждому полюсу реляционной сети на рис. 4.1 поставлена в соответствие своя предметная переменная модели и область ее задания: x_1 – род формы слова со значениями М – мужской, Ж – женский, С – средний ($x_1 \in A_1 = \{$М, Ж, С$\}$); x_2 – число формы слова со значениями Е – единственное, М – множественное ($x_2 \in A_2 = \{$Е, М$\}$); x_3 – падеж формы слова со значениями И – именительный, Р – родительный, Д – дательный, В – винительный, Т – творительный, П – предложный ($x_3 \in A_3 = \{$И, Р, Д, В, Т, П$\}$); x_4 – признак одушевленности формы слова со значениями О – одушевленный, Н – неодушевленный ($x_4 \in A_4 = \{$Н, О$\}$); x_5 – признак употребляемости формы слова со значениями С – современная, А – архаичная ($x_5 \in A_5 = \{$С, А$\}$); r – номер влияний контекста со значениями 1,2,…,19 ($r \in A_6 = \{1, 2, \ldots, 19\}$); s – тип склонения формы слова со значениями 1, 2, …, 7 ($s \in A_7 = \{1, 2, \ldots, 7\}$); y_1 – последняя буква основы слова со значениями Б, В, Г, Д, Е, Ж, З, К, Л, М, Н, П, Р, С, Т, Х, Ц, Ч, Ш, Щ ($y_1 \in A_8 = \{$Б, В, Г, Д, Е, Ж, З, К, Л, М, Н, П, Р, С, Т, Х, Ц, Ч, Ш, Щ$\}$); y_2 – признак ударности слова со значениями Б - безударный, У – ударный ($y_2 \in A_9 = \{$Б, У$\}$); y_3 – признак смягчения слова со значениями Т - твердый, М – мягкий ($y_3 \in A_{10} = \{$Т, М$\}$); $z_{\text{л}}$ – левая часть окончания формы слова со значениями А, А′, Я, У, У′, Ю, О, О′, Е, Ы, Ы′, И, И′ ($z_{\text{л}} \in A_{11} = \{$А, А′, Я, У, У′, Ю, О, О′, Е, Ы, Ы′, И, И′$\}$); $z_{\text{п}}$ – правая часть окончания формы слова со значениями Я, Ю, Е, Й, М, ГО, МУ, Х, МИ ($z_{\text{п}} \in A_{12} = \{$Я, Ю, Е, Й, М, ГО, МУ, Х, МИ$\}$); z – окончание формы слова со значениями АЯ, А′Я, ЯЯ, УЮ, У′Ю, ЮЮ, ОЕ, О′Е, ЕЕ, ОЙ, О′Й, ЕЙ, ОМ, О′М, ЕМ, ОГО, О′ГО, ЕГО, ОМУ, О′МУ, ЕМУ, ОЮ, О′Ю, ЕЮ, ЫЙ, ИЙ, ЫМ, Ы′М, ИМ, И′М, ЫЕ, Ы′Е, ИЕ, И′Е, ЫХ, Ы′Х, ИХ, И′Х, ЫМИ, Ы′МИ, ИМИ, И′МИ ($z \in A_{13} = \{$АЯ, А′Я, ЯЯ, УЮ, У′Ю, ЮЮ, ОЕ, О′Е, ЕЕ, ОЙ, О′Й, ЕЙ, ОМ, О′М, ЕМ, ОГО, О′ГО, ЕГО, ОМУ, О′МУ, ЕМУ, ОЮ, О′Ю, ЕЮ, ЫЙ, ИЙ, ЫМ, Ы′М, ИМ, И′М, ЫЕ, Ы′Е, ИЕ, И′Е, ЫХ, Ы′Х, ИХ, И′Х, ЫМИ, Ы′МИ, ИМИ, И′МИ$\}$); z_0 – знак ударности окончания ($z_0 \in A_{14} = \{*,\ '\}$); z_1 – первая буква окончания ($z_1 \in A_{15} = \{$А, Я, У, Ю, О, Е, Ы, И$\}$); z_2 – вторая буква окончания ($z_2 \in A_{16} = \{$Я, Ю, Е, Й, М, Г, Х $\}$); z_3 – третья буква окончания ($z_3 \in A_{17} = \{*$, О, У, И$\}$).

$$
\begin{gathered}
P_1(x_1, r) = x_1^{\text{М}}(r^1 \vee r^4 \vee r^5) \vee (x_1^{\text{М}} \vee x_1^{\text{С}})(r^2 \vee r^3 \vee r^6 \vee r^7) \vee \\
\vee x_1^{\text{Ж}}(r^8 \vee r^9 \vee \vee r^{10} \vee r^{11} \vee r^{12}) \vee x_1^{\text{С}} r^{13} \vee \\
\vee (x_1^{\text{М}} \vee x_1^{\text{Ж}} \vee x_1^{\text{С}})(r^{14} \vee r^{15} \vee r^{16} \vee r^{17} \vee r^{18} \vee r^{19}), \\
P_{12}(z_{\text{п}}, z_2) = z_{\text{п}}^{\text{Я}} z_2^{\text{Я}} \vee z_{\text{п}}^{\text{Ю}} z_2^{\text{Ю}} \vee z_{\text{п}}^{\text{Е}} z_2^{\text{Е}} \vee z_{\text{п}}^{\text{Й}} z_2^{\text{Й}} \vee \\
\vee z_{\text{п}}^{\text{ГО}} z_2^{\text{Г}} \vee z_{\text{п}}^{\text{Х}} z_2^{\text{Х}} \vee (z_{\text{п}}^{\text{М}} \vee z_{\text{п}}^{\text{МУ}} \vee z_{\text{п}}^{\text{МИ}}) z_2^{\text{М}}.
\end{gathered}
$$

Каждой ветви реляционной сети на рис. 4.1 поставлено в соответствие бинарное отношение модели P_1–P_{17}. Ниже приведены примеры формульного и графического представления отношений реляционной сети (рис. 4.2, 4.3):

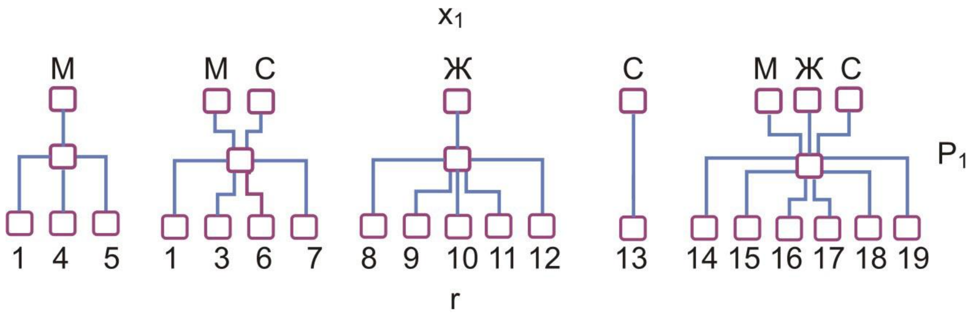

Рис. 4.2. Отношение P_1, связывающее переменные x_1 и r

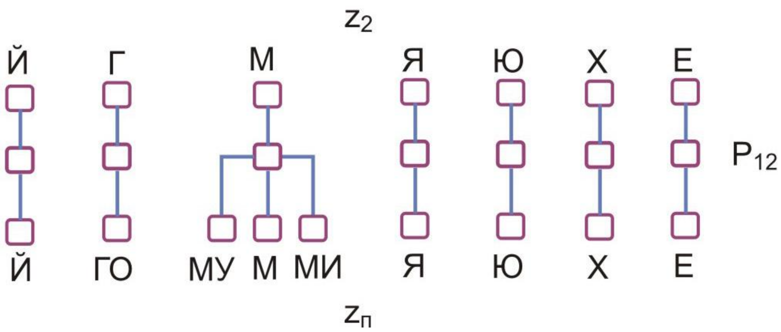

Рис.4.3. Отношение P_{12}, связывающее переменные $z_п$ и z_2.

Предикат данной модели реляционной сети имеет следующий формульный вид:

$$P(x_1, x_2, x_3, x_4, x_5, r, s, y_1, y_2, y_3, z_л, z_n, z, z_0, z_1, z_2, z_3) =$$
$$= P_1(x_1,r) \wedge P_2(x_2,r) \wedge P_3(x_3,r) \wedge P_4(x_4,r) \wedge P_5(x_5,r) \wedge P_6(z_л,r) \wedge$$
$$\wedge P_7(z_п,r) \wedge P_8(z_л,z) \wedge P_9(z_п,z) \wedge P_{10}(z_л,z_0) \wedge P_{11}(z_л,z_1) \wedge P_{12}(z_п,z_2) \wedge$$
$$\wedge P_{13}(z_п,z_3) \wedge P_{14}(s,z_л) \wedge P_{15}(y_1,s) \wedge P_{16}(y_2,s) \wedge P_{17}(y_3,s).$$

Память с ассоциативным доступом (ассоциативная память) представляет собой вид машинной памяти, используемой в приложениях очень быстрого поиска. В отличие от обычной памяти произвольного доступа или RAM, в которой пользователь задает адрес памяти и ОЗУ возвращает слово данных, хранящееся по этому адресу, ассоциативная память разработана таким образом, чтобы пользователь задавал слово данных. После чего, ассоциативная память ищет его во всей памяти, чтобы выяснить, хранится ли оно где-нибудь в нем. Если слово данных найдено, ассоциативная память возвращает список одного или более адресов хранения, где слово было найдено (и в некоторых архитектурах, также возвращает само слово данных, или другие связанные части данных).

Для аппаратной реализации выше описанной модели реляционной сети в виде памяти с ассоциативным доступом были составлены таблицы, соответствующие 3 группам, в которые можно естественно объединить полюсы реляционной сети:

– в первую группу входят полюсы с предметными переменными, значения которых характеризуют контекст, окружающий склоняемое слово (табл. 4.1);

Таблица 4.1

r	x_1	x_2	x_3	x_4	x_5
1	М	Е	И	Н, О	С, А
2	М, С	Е	Р	Н, О	С, А
3	М, С	Е	Д	Н, О	С, А
4	М	Е	В	Н	С, А
5	М	Е	В	О	С, А
6	М, С	Е	Т	Н, О	С, А
7	М, С	Е	П	Н, О	С, А
8	Ж	Е	И	Н, О	С, А
9	Ж	Е	Р, Д, П	Н, О	С, А
10	Ж	Е	В	Н, О	С, А
11	Ж	Е	Т	Н, О	С
12	Ж	Е	Т	Н, О	А
13	С	Е	И, В	Н, О	С, А
14	М, Ж, С	М	И	Н, О	С, А
15	М, Ж, С	М	Р, П	Н, О	С, А
16	М, Ж, С	М	Д	Н, О	С, А
17	М, Ж, С	М	В	Н	С, А
18	М, Ж, С	М	В	О	С, А
19	М, Ж, С	М	Т	Н, О	С, А

– во вторую группу входят полюсы, характеризующие само склоняемое слово (табл. 4.2);

Таблица 4.2

s	y_1	y_2	y_3
1	Б, В, Д, З, Л, М, П, Т, Н, Р, С	У	Т
2	Г, К, Х	У	М
3	Ц	У	Т
4	Ж, Ч, Ш, Щ	У	М
5	Е, Н, Р, С	У	М
6	Б, В, Д, З, Л, М, П, Т, Н, Р, С	Б	Т
7	Ж, Ч, Ш, Щ, Г, К, Х	Б	Т

– в третью группу входят полюсы, которые характеризуют результат склонения слова, то есть словоформу, соответствующую данному контексту и данному слову (табл. 4.3).

Таблица 4.3

z	$z_л$	z_0	z_1	$z_п$	z_2	z_3	s	R
АЯ	А	*	А	Я	Я	*	1,2,3,4	8
А′Я	А′	′	А	Я	Я	*	6,7	8
ЕЕ	Е	*	Е	Е	Е	*	3,4,5	13
ЕГО	Е	*	Е	ГО	Г	О	3,4,5	2,5
ЕЙ	Е	*	Е	Й	Й	*	3,4,5	9,11
ЕМ	Е	*	Е	М	М	*	3,4,5	7
ЕМУ	Е	*	Е	МУ	М	У	3,4,5	3
ЕЮ	Е	*	Е	Ю	Ю	*	3,4,5	12
ИЕ	И	*	И	Е	Е	*	2,4,5	14,17
И′Е	И′	′	И	Е	Е	*	7	14,17
ИЙ	И	*	И	Й	Й	*	2,4,5	1,4
ИМ	И	*	И	М	М	*	2,4,5	6,16
И′М	И′	′	И	М	М	*	7	6,16
ИХ	И	*	И	Х	Х	*	2,4,5	15,18
И′Х	И′	′	И	Х	Х	*	7	15,18
ИМИ	И	*	И	МИ	М	И	2,4,5	19
И′МИ	И′	′	И	МИ	М	И	7	19
ОЕ	О	*	О	Е	Е	*	1,2	13

Продолжение табл.4.3

О′Е	О′	′	О	Е	Е	*	6,7	13
ОГО	О	*	О	ГО	Г	О	1,2	2,5
О′ГО	О′	′	О	ГО	Г	О	6,7	2,5
ОЙ	О	*	О	Й	Й	*	1,2	9,11
О′Й	О′	′	О	Й	Й	*	6,7	1,4,9,11
ОМ	О	*	О	М	М	*	1,2	7
О′М	О′	′	О	М	М	*	6,7	7
ОМУ	О	*	О	МУ	М	У	1,2	3
О′МУ	О′	′	О	МУ	М	У	6,7	3
ОЮ	О	*	О	Ю	Ю	*	1,2	12
О′Ю	О′	′	О	Ю	Ю	*	6,7	12
УЮ	У	*	У	Ю	Ю	*	1,2,3,4	10
У′Ю	У′	′	У	Ю	Ю	*	6,7	10
ЫЕ	Ы	*	Ы	Е	Е	*	1,3	14,17
Ы′Е	Ы′	′	Ы	Е	Е	*	6	14,17
ЫЙ	Ы	*	Ы	Й	Й	*	1,3	1,4
ЫМ	Ы	*	Ы	М	М	*	1,3	6,16
Ы′М	Ы′	′	Ы	М	М	*	6	6,16
ЫМИ	Ы	*	Ы	МИ	М	И	1,3	19
Ы′МИ	Ы′	′	Ы	МИ	М	И	6	19
ЫХ	Ы	*	Ы	Х	Х	*	1,3	15,18
Ы′Х	Ы′	′	Ы	Х	Х	*	6	15,18
ЮЮ	Ю	*	Ю	Ю	Ю	*	5	10
ЯЯ	Я	*	Я	Я	Я	*	5	8

Поиск информации в памяти с ассоциативным доступом ведется исходя из некоторого признака данных. При выборке входные данные одновременно сравниваются со всеми полями признака во всех ячейках памяти. Для этого было произведено кодирование признаков всех предметных переменных реляционной сети и их возможных сочетаний. Для кодирования алфавитов переменных r, x_2 вершины P_1 данной логической ассоциативной структуры был использован комбинационный код, приведенный в табл. 4.4 – 4.5.

Таблица 4.4

r	Code
1	00000
2	00001
3	00010
4	00011
5	00100
6	00101
7	00110
8	00111
9	01000
10	01001
11	01010
12	01011
13	01100
14	01101
15	01110
16	01111
17	10000
18	10001
19	10010

Таблица 4.5

x_2	Code
Е	0
М	1

Для кодирования алфавитов переменных x_1, x_3, x_4, x_5 вершины P_1 был использован позиционный код. В табл. 4.6 представлено кодирование всех алфавитов ассоциативных векторов вершины P_1.

Таблица 4.6

№	*r*	x_1			x_2	x_3						x_4		x_5	
		М	Ж	С		И	Р	Д	В	Т	П	Н	О	С	А
0	00000	1	0	0	0	1	0	0	0	0	0	1	1	1	1
1	00001	1	0	1	0	0	1	0	0	0	0	1	1	1	1
2	00010	1	0	1	0	0	0	1	0	0	0	1	1	1	1
3	00011	1	0	0	0	0	0	0	1	0	0	1	0	1	1
4	00100	1	0	0	0	0	0	0	1	0	0	0	1	1	1
5	00101	1	0	1	0	0	0	0	0	1	0	1	1	1	1
6	00110	1	0	1	0	0	0	0	0	0	1	1	1	1	1
7	00111	0	1	0	0	1	0	0	0	0	0	1	1	1	1

Продолжение табл. 4.6

8	01000	0	1	0	0	0	1	1	0	0	1	1	1	1	1
9	01001	0	1	0	0	0	0	0	1	0	0	1	1	1	1
10	01010	0	1	0	0	0	0	0	0	1	0	1	1	1	0
11	01011	0	1	0	0	0	0	0	0	1	0	1	1	0	1
12	01100	0	0	1	0	1	0	0	1	0	0	1	1	1	1
13	01101	1	1	1	1	1	0	0	0	0	0	1	1	1	1
14	01110	1	1	1	1	0	1	0	0	0	1	1	1	1	1
15	01111	1	1	1	1	0	0	1	0	0	0	1	1	1	1
16	10000	1	1	1	1	0	0	0	1	0	0	1	0	1	1
17	10001	1	1	1	1	0	0	0	1	0	0	0	1	1	1
18	10010	1	1	1	1	0	0	0	0	1	0	1	1	1	1

Для кодирования алфавитов переменных s, y_1 вершины P_2 логической ассоциативной структуры был использован позиционный код, алфавитов переменных y_2, y_3 – комбинационный код, приведенный в табл. 4.7 – 4.8.

Таблица 4.7

y_2	code
У	0
Б	1

Таблица 4.8

y_3	code
Т	0
М	1

В табл. 4.9 представлено кодирование всех алфавитов ассоциативных векторов вершины P_2.

Таблица 4.9

№	s							y_1																				y_2	y_3
0	1	2	3	4	5	6	7	Б	В	Г	Д	Е	Ж	З	К	Л	М	Н	П	Р	С	Т	Х	Ц	Ч	Ш	Щ		
1	1	0	0	0	0	0	0	1	1	0	1	0	0	1	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1
3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
4	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1
5	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	1
6	0	0	0	0	0	1	0	1	1	0	1	0	0	1	0	1	1	1	1	1	1	1	0	0	0	0	0	1	0
7	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	1	0	1	1	1	1	0

Для вершины P_3 алфавит переменной z был комбинационно закодирован шестью разрядами (табл. 4.10), алфавиты переменных $z_л, z_п$ – четырьмя разрядами (табл. 4.11, 4.14), алфавит переменной z_0 – одним разрядом (табл. 4.12), алфавиты переменных z_1, z_2 – тремя разрядами (табл. 4.13, 4.15), алфавит переменной z_3 – двумя разрядами (табл. 4.16).

Таблица 4.10

z	Code	z	Code
АЯ	000000	ОЙ	010101
А′Я	000001	О′Й	010110
ЕЕ	000010	ОМ	010111
ЕГО	000011	О′М	011000
ЕЙ	000100	ОМУ	011001
ЕМ	000101	О′МУ	011010
ЕМУ	000110	ОЮ	011011
ЕЮ	000111	О′Ю	011100
ИЕ	001000	УЮ	011101
И′Е	001001	У′Ю	011110
ИЙ	001010	ЫЕ	011111
ИМ	001011	Ы′Е	100000
И′М	001100	ЫЙ	100001
ИХ	001101	ЫМ	100010
И′Х	001110	Ы′М	100011
ИМИ	001111	ЫМИ	100100
И′МИ	010000	Ы′МИ	100101
ОЕ	010001	ЫХ	100110
О′Е	010010	Ы′Х	100111
ОГО	010011	ЮЮ	101000
О′ГО	010100	ЯЯ	101001

Таблица 4.11

$z_л$	Code
А	0000
А′	0001
Е	0010
И	0011
И′	0100
О	0101
О′	0110
У	0111
У′	1000
Ы	1001
Ы′	1010
Ю	1011
Я	1100

Таблица 4.12

z_0	Code
*	0
′	1

Таблица 4.13

z_1	Code
А	000
Е	001
И	010
О	011
У	100
Ы	101
Ю	110
Я	111

Таблица 4.14

$z_п$	Code
ГО	0000
Е	0001
Й	0010
М	0011
МИ	0100
МУ	0101
Х	0110
Ю	0111
Я	1000

Таблица 4.15

z_2	Code
Г	000
Е	001
Й	010
М	011
Х	100
Ю	101
Я	110

Таблица 4.16

z_3	Code
И	00
О	01
У	10
*	11

Таким образом, кодирование всех алфавитов вершины P_3 логической ассоциативной структуры представляет собой табл. 4.17.

Таблица 4.17

№	z	$z_л$	z_0	z_1	z_n	z_2	z_3	s							R
								1	2	3	4	5	6	7	
0	000000	0000	0	000	1000	110	11	1	1	1	1	0	0	0	00111
1	000001	0001	1	000	1000	110	11	0	0	0	0	0	1	1	00111
2	000010	0010	0	001	0001	001	11	0	0	1	1	1	0	0	01100
3	000011	0010	0	001	0000	000	01	0	0	1	1	1	0	0	00001
4	000011	0010	0	001	0000	000	01	0	0	1	1	1	0	0	00100
5	000100	0010	0	001	0010	010	11	0	0	1	1	1	0	0	01000
6	000100	0010	0	001	0010	010	11	0	0	1	1	1	0	0	01010
7	000101	0010	0	001	0011	011	11	0	0	1	1	1	0	0	00110
8	000110	0010	0	001	0101	011	10	0	0	1	1	1	0	0	00010
9	000111	0010	0	001	0111	101	11	0	0	1	1	1	0	0	01011
10	001000	0011	0	010	0001	001	11	0	1	0	1	1	0	0	01101
11	001000	0011	0	010	0001	001	11	0	1	0	1	1	0	0	10000
12	001001	0100	1	010	0001	001	11	0	0	0	0	0	0	1	01101
13	001001	0100	1	010	0001	001	11	0	0	0	0	0	0	1	10000
14	001010	0011	0	010	0010	010	11	0	1	0	1	1	0	0	00000
15	001010	0011	0	010	0010	010	11	0	1	0	1	1	0	0	00011
16	001011	0011	0	010	0011	011	11	0	1	0	1	1	0	0	00101
17	001011	0011	0	010	0011	011	11	0	1	0	1	1	0	0	01111
18	001100	0100	1	010	0011	011	11	0	0	0	0	0	0	1	00101
19	001100	0100	1	010	0011	011	11	0	0	0	0	0	0	1	01111
20	001101	0011	0	010	0110	100	11	0	1	0	1	1	0	0	01110
21	001101	0011	0	010	0110	100	11	0	1	0	1	1	0	0	10001
22	001110	0100	1	010	0110	100	11	0	0	0	0	0	0	1	01110
23	001110	0100	1	010	0110	100	11	0	0	0	0	0	0	1	10001
24	001111	0011	0	010	0100	011	00	0	1	0	1	1	0	0	10010
25	010000	0100	1	010	0100	011	00	0	0	0	0	0	0	1	10010
26	010001	0101	0	011	0001	001	11	1	1	0	0	0	0	0	01100
27	010010	0110	1	011	0001	001	11	0	0	0	0	0	1	1	01100
28	010011	0101	0	011	0000	000	01	1	1	0	0	0	0	0	00001
29	010011	0101	0	011	0000	000	01	1	1	0	0	0	0	0	00100
30	010100	0110	1	011	0000	000	01	0	0	0	0	0	1	1	00001
31	010100	0110	1	011	0000	000	01	0	0	0	0	0	1	1	00100
32	010101	0101	0	011	0010	010	11	1	1	0	0	0	0	0	01000
33	010101	0101	0	011	0010	010	11	1	1	0	0	0	0	0	01010
34	010110	0110	1	011	0010	010	11	0	0	0	0	0	0	0	00000

Продолжение табл.4.17

35	010110	0110	1	011	0010	010	11	0	0	0	0	0	0	0	00011
36	010110	0110	1	011	0010	010	11	0	0	0	0	0	0	0	01000
37	010110	0110	1	011	0010	010	11	0	0	0	0	0	0	0	01010
38	010111	0101	0	011	0011	011	11	1	1	0	0	0	0	0	00110
39	011000	0110	1	011	0011	011	11	0	0	0	0	0	1	1	00110
40	011001	0101	0	011	0101	011	10	1	1	0	0	0	0	0	00010
41	011010	0110	1	011	0101	011	10	0	0	0	0	0	1	1	00010
42	011011	0101	0	011	0111	101	11	1	1	0	0	0	0	0	01011
43	011100	0110	1	011	0111	101	11	0	0	0	0	0	1	1	01011
44	011101	0111	0	100	0111	101	11	1	1	1	1	0	0	0	01001
45	011110	1000	1	100	0111	101	11	0	0	0	0	0	1	1	01001
46	011111	1001	0	101	0001	001	11	1	0	1	0	0	0	0	01101
47	011111	1001	0	101	0001	001	11	1	0	1	0	0	0	0	10000
48	100000	1010	1	101	0001	001	11	0	0	0	0	0	1	0	01101
49	100000	1010	1	101	0001	001	11	0	0	0	0	0	1	0	10000
50	100001	1001	0	101	0010	010	11	1	0	1	0	0	0	0	00000
51	100001	1001	0	101	0010	010	11	1	0	1	0	0	0	0	00011
52	100010	1001	0	101	0011	011	11	1	0	1	0	0	0	0	00101
53	100010	1001	0	101	0011	011	11	1	0	1	0	0	0	0	01111
54	100011	1010	1	101	0011	011	11	0	0	0	0	0	1	0	00101
55	100011	1010	1	101	0011	011	11	0	0	0	0	0	1	0	01111
56	100100	1001	0	101	0100	011	00	1	0	1	0	0	0	0	10010
57	100101	1010	1	101	0100	011	00	0	0	0	0	0	1	0	10010
58	100110	1001	0	101	0110	100	11	1	0	1	0	0	0	0	01110
59	100110	1001	0	101	0110	100	11	1	0	1	0	0	0	0	10001
60	100111	1010	1	101	0110	100	11	0	0	0	0	0	1	0	01110
61	100111	1010	1	101	0110	100	11	0	0	0	0	0	1	0	10001
62	101000	1011	0	110	0111	101	11	0	0	0	0	1		0	01001
63	101001	1100	0	111	1000	110	11	0	0	0	0	1	0	0	00111

Итак, логическая ассоциативная структура для реляционной сети склонения полных непритяжательных имен прилагательных имеет три вершины:

$$\Psi = f(P_1, P_2, P_3), O = \{\cup, \cap\},$$

где

$$P_1 = \{r, x_1, x_2, x_3, x_4, x_5\},\ \ \pi_1 = 34,$$

$$\beta_1 = \{\{1,2,3,...,19\}, \{M, Ж, C\}, \{E, M\}, \{И, P, Д, B, T, П\}, \{H, O\}, \{C, A\}\},$$

$$P_2 = \{s, z, y_1, y_2, y_3\},\ \pi_2 = 31,$$

$$\beta_2 = \{\{1,2,...,7\},\{Б,В,Г,Д,Е,Ж,З,К,Л,М,Н,П,Р,С,Т,Х,Ц,Ч,Ш,Щ\},$$
$$\{Б,У\},\{Т,М\}\},$$

$$P_3 = \{z, z_Л, z_0, z_1, z_П, z_2, z_3, r, s\},\ \pi_2 = 111,$$

$$\beta_3 = \{\{АЯ, А'Я, ЕЕ, ЕГО, ЕЙ, ЕМ, ЕМУ, ЕЮ, ИЕ, И'Е, ИЙ, М, И'М,$$
$$ИХ, И'Х, ИМИ, И'МИ, ОЕ, О'Е, ОГО, О'ГО, ОЙ, О'Й, ОМ, О'М, ОМУ,$$
$$О'МУ, ОЮ, О'Ю, УЮ, У'Ю, ЫЕ, Ы'Е, ЫЙ, ЫМ, Ы'М, ЫМИ, Ы'МИ,$$
$$ЫХ, Ы'Х, ЮЮ, ЯЯ\},\{А, А', Е, И, И', О, О', У, У', Ы, Ы', Ю, Я\},$$
$$\{*,'\},\{А, Е, И, О, У, Ы, Ю, Я\},\{ГО, Е, Й, М, МИ, МУ, Х, Ю, Я\},$$
$$\{Г, Е, Й, М, Х, Ю, Я\},\{И, О, У, *\},\{1,2,...,19\},\{1,2,...,7\}\}.$$

Полученная логическая ассоциативная структура предназначена для автоматического решения класса задач, относящихся к склонению полных непритяжательных имен прилагательных. Примером задачи этого класса может служить задача синтеза словоформы. Она заключается в следующем: заданы слово и окружающий его контекст, требуется определить соответствующую им словоформу. Например, пусть задано словосочетание (*слабый*) *сигналом*. Требуется согласовать имя прилагательное *слабый*, приведенные в скобках, с контекстом. Определяем исходные данные: последняя буква основы словоформы – *б*; род – *мужской*; число – *единственное;* падеж – *творительный*. Для каждой вершины логической ассоциативной структуры задаем вектор–маску входных и выходных переменных, а также входное воздействие. В табл. 4.18 представлены маски входа M_1 и выхода Y_1, а также входное воздействие X_1 для вершины P_1 логической ассоциативной структуры. Маски входа M_2 и выхода Y_2, а также входное воздействие X_2 для вершины P_2 логической ассоциативной структуры представлены в табл. 4.19.

Таблица 4.18

	r	x_1			x_2	x_3						x_4		x_5	
		М	Ж	С		И	Р	Д	В	Т	П	Н	О	С	А
M_1	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0
X_1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
Y_1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Таблица 4.19

	s							y_1																				y_2	y_3
	1	2	3	4	5	6	7	Б	В	Г	Д	Е	Ж	З	К	Л	М	Н	П	Р	С	Т	Х	Ц	Ч	Ш	Ш		
М	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

В результате обработки таблицы ассоциативных векторов вершины P_1 (табл.4.6) получаем значение переменной $r = 00101$, вершины P_2 (табл. 4.9) – значение переменной $s = \{1\}$.

В табл. 4.20 представлены маски входа M_3 и выхода Y_3, а также входное воздействие X_3 для вершины P_3 логической ассоциативной структуры.

Таблица 4.20

	z	$z_л$	z_0	z_1	z_n	z_2	z_3	s							r
								1	2	3	4	5	6	7	
M_3	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
X_3	0	0	0	0	0	0	0	1	0	0	0	0	0	0	00101
Y_3	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

В результате обработки таблицы ассоциативных векторов вершины P_3 (табл.4.17) получаем значения переменных $z = \{100010\}$, $z_Л = \{1001\}$, $z_0 = \{0\}$, $z_1 = \{101\}$, $z_П = \{0011\}$, $z_2 = \{011\}$, $z_3 = \{11\}$. После декодирования получаем безударную словоформу *ым*. В результате решения этой задачи формируем словосочетание *слабым сигналом*.

Другим примером может служить задача анализа словоформы: заданы форма слова и окружающий ее контекст, требуется определить грамматические признаки, соответствующие этой словоформе.

Например, пусть задана форма слова *слабым*. Требуется указать род, число и падеж данной словоформы. Определяем исходные данные: основа словоформы – *слаб*; окончание – *ым.* Маски входа и выхода, а также входное воздействие для вершины P_2 логической ассоциативной структуры для задачи анализа аналогичны маскам входа и выхода, входным воздействиям для задачи синтеза (табл. 4.18 – 4.19). В табл. 4.21 представлены маски входа M_3 и выхода Y_3, а также входное воздействие X_3 для вершины P_3 логической ассоциативной структуры.

Таблица 4.21

	z	$z_л$	z_0	z_1	z_n	z_2	z_3	s							r
								1	2	3	4	5	6	7	
M_3	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
X_3	100010	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Y_3	0	1	1	1	1	1	1	0	0	0	0	0	0	0	1

В результате обработки таблицы ассоциативных векторов вершины P_3 получаем два значения переменной $r = \{00101, 01111\}$. В табл.4.22 представлены маски входа M_1 и выхода Y_1, а также входное воздействие X_1 для вершины P_1 логической ассоциативной структуры.

Таблица 4.22

	r	x_1			x_2	x_3						x_4		x_5	
		М	Ж	С		И	Р	Д	В	Т	П	Н	О	С	А
M_1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X_{11}	00101	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X_{12}	01111	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Y_1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1

В результате обработки таблицы ассоциативных векторов вершины P_1 (табл.4.17) получаем два значения переменной $r = \{00101, 01111\}$. После декодирования получаем два решения этой задачи – словоформа *слабым* имеет: 1) мужской или средний род, единственное число, творительный падеж; 2) множественное число, дательный падеж. Например, такие грамматические признаки имеет данная словоформа в словосочетаниях: *слабым сигналом, слабым полем, слабым сигналам*.

Перспективой дальнейших исследований является выбор стратегии анализа преобразованных таблиц, а также синтез графического представления таблиц в виде графов.

4.2. Словосочетания естественного языка как реляционные сети

Рассмотрим способ формульной записи структуры словосочетаний естественного языка и метод синтеза реляционных сетей для операций соединения слов в словосочетаниях. Формульная запись синтаксической структуры словосочетаний – смысловых и грамматических объединений двух или более знаменательных слов основана на подчиненной синтаксической связи. Различают три вида операций соединения слов: согласование, управление и примыкание. При согласовании изменение формы главного слова влечет изменение формы зависимого слова, при управлении изменение формы главного слова не влечет изменения формы зависимого слова, при примыкании зависимое слово, будучи неизменным, связывается с главным только по смыслу.

Рассмотрим конкретные способы образования словосочетаний из отдельных слов. Согласование имени прилагательного A с именем существительным N осуществляется по схеме $C(x) = A(x) \wedge N(x)$. В результате получаем словосочетание C. Например, толстая книга(x)=толстая(x)$\wedge$книга(x). Формула, стоящая справа от знака равенства в вышеприведенном выражении, означает: “Предмет x – толстый, и этот предмет есть книга”. Управление одного имени существительного N_2 другим именем существительным N_1 часто осуществляется по схеме $C(x) = N_1(x) \wedge \exists y(N_2(y) \wedge \text{деталь}\,(x, y))$.

В результате получаем словосочетание C. Предикат $\text{деталь}(x, y)$ имеет следующее содержание: “Предмет x является деталью (составной частью) предмета y”. К примеру, по этой схеме получаем страница книги(x)=страница(x)$\wedge\exists y$(книга(y)$\wedge$деталь(x, y)).

Смысл правой части данного равенства выражается высказыванием: “Предмет x есть страница, и существует предмет y, являющийся книгой, такой что предмет x служит его деталью”. Управление имени существительного N количественным числительным Num может осуществляться по схеме $C(x) = Num(x) \wedge \forall y(y \in x \supset N(y))$.

Например, две книги(x)=два(x)$\wedge\forall y$($y\in x\supset$ книга(y)). Содержание правой части данного равенства можно передать следующим высказыванием: “Множество x состоит из двух предметов, и каждый из предметов y, если он принадлежит множеству x, есть книга”.

Рассмотрим методы синтеза реляционных сетей [3, 4, 5] для словосочетаний естественного языка. Например, возьмем словосочетание *толковый словарь*. В ячейках парадигматических табл. 4.23 – 4.24 приведены все словоформы имени прилагательного *толковый* и имени существительного *словарь*, соответственно, в зависимости от падежа и числа.

Таблица 4.23

Падеж	Ед.ч.	Мн.ч.
И.	толковый	толковые
Р.	толкового	толковых
Д.	толковому	толковым
В.	толковый	толковые
Т.	толковым	толковыми
П.	(о) толковом	(о) толковых

Таблица 4.24

Падеж	Ед.ч.	Мн.ч.
И.	словарь	словари
Р.	словаря	словарей
Д.	словарю	словарям
В.	словарь	словари
Т.	словарём	словарями
П.	(о) словаре	(о) словарях

Для согласования имени прилагательного *толковый* и имени существительного *словарь* составим парадигматическую табл. 4.25.

Таблица 4.25

Падеж	Ед.ч.	Мн.ч.
И.	толковый словарь	толковые словари
Р.	толкового словаря	толковых словарей
Д.	толковому словарю	толковым словарям
В.	толковый словарь	толковые словари
Т.	толковым словарём	толковыми словарями
П.	(о) толковом словаре	(о) толковых словарях

Приступим к формальному описанию процесса согласования имени прилагательного *толковый* и имени существительного *словарь*. С этой целью введём необходимые предметные переменные: x – падеж формы слова со значениями: именительный, родительный, дательный, винительный, творительный, предложный ($x \in A = \{$И, Р, Д, В, Т, П$\}$); y – число формы слова со значениями: единственное, множественное ($y \in B = \{$Е, М$\}$); z_n – окончание формы слова имени прилагательного со значениями: ого, ом, ому, ые, ый, ым, ыми, ых ($z_n \in C = \{$ОГО, ОМ, ОМУ, ЫЕ, ЫЙ, ЫМ, ЫМИ, ЫХ$\}$);

z_c – окончание формы слова имени существительного со значениями: е, ей, ём, и, ь, ю, я, ям, ями, ях ($z_c \in D = \{$Е, ЕЙ, ЁМ, И, Ь, Ю, Я, ЯМ, ЯМИ, ЯХ$\}$). Области изменения введенных переменных A, B, C, D формально зададим следующими уравнениями:

$$x^{\text{И}} \vee x^{\text{Р}} \vee x^{\text{Д}} \vee x^{\text{В}} \vee x^{\text{Т}} \vee x^{\text{П}} = 1;\ y^{\text{Е}} \vee y^{\text{М}} = 1;$$

$$z_c{}^{\text{Е}} \vee z_c{}^{\text{ЕЙ}} \vee z_c{}^{\text{ЁМ}} \vee z_c{}^{\text{И}} \vee z_c{}^{\text{Ь}} \vee z_c{}^{\text{Ю}} \vee z_c{}^{\text{Я}} \vee z_c{}^{\text{ЯМ}} \vee z_c{}^{\text{ЯМИ}} \vee z_c{}^{\text{ЯХ}} = 1;$$

$$z_n{}^{\text{ОГО}} \vee z_n{}^{\text{ОМ}} \vee z_n{}^{\text{ОМУ}} \vee z_n{}^{\text{ЫЕ}} \vee z_n{}^{\text{ЫЙ}} \vee z_n{}^{\text{ЫМ}} \vee z_n{}^{\text{ЫМИ}} \vee z_n{}^{\text{ЫХ}} = 1.$$

В соответствии с введенными предметными переменными составим таблицу отношения P, соответствующего предикату $P(x, y, z_c, z_n)$ (табл.4.26).

Таблица 4.26 - Отношение P

Падеж, x	Число, y			
	Е		М	
И	ЫЙ	Ь	ЫЕ	И
Р	ОГО	Я	ЫХ	ЕЙ
Д	ОМУ	Ю	ЫМ	ЯМ
В	ЫЙ	Ь	ЫЕ	И
Т	ЫМ	ЁМ	ЫМИ	ЯМИ
П	ОМ	Е	ЫХ	ЯХ

Запишем СДНФ данного предиката:

$$\begin{aligned} P(x, y, z_n, z_c) = {} & x^{И} y^{E} z_n{}^{\text{ЫЙ}} z_c{}^{Ь} \vee x^{P} y^{E} z_n{}^{\text{ОГО}} z_c{}^{Я} \vee x^{Д} y^{E} z_n{}^{\text{ОМУ}} z_c{}^{Ю} \vee \\ & \vee x^{B} y^{E} z_n{}^{\text{ЫЙ}} z_c{}^{Ь} \vee x^{T} y^{E} z_n{}^{\text{ЫМ}} z_c{}^{ЁМ} \vee x^{П} y^{E} z_n{}^{\text{ОМ}} z_c{}^{E} \vee \\ & \vee x^{И} y^{M} z_n{}^{\text{ЫЕ}} z_c{}^{И} \vee x^{P} y^{M} z_n{}^{\text{ЫХ}} z_c{}^{ЕЙ} \vee x^{Д} y^{M} z_n{}^{\text{ЫМ}} z_c{}^{ЯМ} \vee \\ & \vee x^{B} y^{M} z_n{}^{\text{ЫЕ}} z_c{}^{И} \vee x^{T} y^{M} z_n{}^{\text{ЫМИ}} z_c{}^{ЯМИ} \vee x^{П} y^{M} z_n{}^{\text{ЫХ}} z_c{}^{ЯХ}. \end{aligned}$$

Составим функцию $f(x, y, z_n, z_c) = u$, $u \in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$ (табл. 4.27).

Таблица 4.27. $f(x, y, z_n, z_c) = u$

Падеж, x	Число, y					
	Е			М		
И	ЫЙ	Ь	1	ЫЕ	И	7
Р	ОГО	Я	2	ЫХ	ЕЙ	8
Д	ОМУ	Ю	3	ЫМ	ЯМ	9
В	ЫЙ	Ь	4	ЫЕ	И	10
Т	ЫМ	ЁМ	5	ЫМИ	ЯМИ	11
П	ОМ	Е	6	ЫХ	ЯХ	12

Запишем формульный вид предиката, соответствующего данному отношению:

$$
\begin{gathered}
R(x, y, z_n, z_c, u) = \\
= x^{И} y^{Е} z_n{}^{\text{ЫЙ}} z_c{}^{Ь} u^1 \vee x^{Р} y^{Е} z_n{}^{\text{ОГО}} z_c{}^{Я} u^2 \vee x^{Д} y^{Е} z_n{}^{\text{ОМУ}} z_c{}^{Ю} u^3 \vee \\
\vee x^{В} y^{Е} z_n{}^{\text{ЫЙ}} z_c{}^{Ь} u^4 \vee x^{Т} y^{Е} z_n{}^{\text{ЫМ}} z_c{}^{ЁМ} u^5 \vee x^{П} y^{Е} z_n{}^{\text{ОМ}} z_c{}^{Е} u^6 \vee \\
\vee x^{И} y^{М} z_n{}^{\text{ЫЕ}} z_c{}^{И} u^7 \vee x^{Р} y^{М} z_n{}^{\text{ЫХ}} z_c{}^{ЕЙ} u^8 \vee x^{Д} y^{М} z_n{}^{\text{ЫМ}} z_c{}^{ЯМ} u^9 \vee \\
\vee x^{В} y^{М} z_n{}^{\text{ЫЕ}} z_c{}^{И} u^{10} \vee x^{Т} y^{М} z_n{}^{\text{ЫМИ}} z_c{}^{ЯМИ} u^{11} \vee x^{П} y^{М} z_n{}^{\text{ЫХ}} z_c{}^{ЯХ} u^{12}.
\end{gathered}
\tag{4.1}
$$

Используя квантор существования по переменной u, получим исходный предикат $P(x, y, z)$:

$$
\begin{aligned}
&\exists u \in U\ R(x, y, z, u) = \exists u \in \{1,2,3,4,5,6,7,8,9,10,11,12\}\ (x^{И} y^{Е} z_n{}^{\text{ЫЙ}} z_c{}^{Ь} u^1 \vee \\
&\vee x^{Р} y^{Е} z_n{}^{\text{ОГО}} z_c{}^{Я} u^2 \vee x^{Д} y^{Е} z_n{}^{\text{ОМУ}} z_c{}^{Ю} u^3 \vee x^{В} y^{Е} z_n{}^{\text{ЫЙ}} z_c{}^{Ь} u^4 \vee \\
&\vee x^{Т} y^{Е} z_n{}^{\text{ЫМ}} z_c{}^{ЁМ} u^5 \vee x^{П} y^{Е} z_n{}^{\text{ОМ}} z_c{}^{Е} u^6 \vee x^{И} y^{М} z_n{}^{\text{ЫЕ}} z_c{}^{И} u^7 \vee \\
&\vee x^{Р} y^{М} z_n{}^{\text{ЫХ}} z_c{}^{ЕЙ} u^8 \vee x^{Д} y^{М} z_n{}^{\text{ЫМ}} z_c{}^{ЯМ} u^9 \vee x^{В} y^{М} z_n{}^{\text{ЫЕ}} z_c{}^{И} u^{10} \vee \\
&\vee x^{Т} y^{М} z_n{}^{\text{ЫМИ}} z_c{}^{ЯМИ} u^{11} \vee x^{П} y^{М} z_n{}^{\text{ЫХ}} z_c{}^{ЯХ} u^{12}) =
\end{aligned}
$$

$$= x^{И} y^{Е} z_n^{\text{ЫЙ}} z_c^{Ь} \vee x^{Р} y^{Е} z_n^{\text{ОГО}} z_c^{Я} \vee x^{Д} y^{Е} z_n^{\text{ОМУ}} z_c^{Ю} \vee$$
$$\vee x^{В} y^{Е} z_n^{\text{ЫЙ}} z_c^{Ь} \vee x^{Т} y^{Е} z_n^{\text{ЫМ}} z_c^{ЁМ} \vee x^{П} y^{Е} z_n^{\text{ОМ}} z_c^{Е} \vee$$
$$\vee x^{И} y^{М} z_n^{\text{ЫЕ}} z_c^{И} \vee x^{Р} y^{М} z_n^{\text{ЫХ}} z_c^{ЕЙ} \vee x^{Д} y^{М} z_n^{\text{ЫМ}} z_c^{ЯМ} \vee$$
$$\vee x^{В} y^{М} z_n^{\text{ЫЕ}} z_c^{И} \vee x^{Т} y^{М} z_n^{\text{ЫМИ}} z_c^{ЯМИ} \vee x^{П} y^{М} z_n^{\text{ЫХ}} z_c^{ЯХ}.$$

Исключая с помощью кванторов существования по три переменные из (4.1), построим бинарные предикаты:

$$\begin{gathered}
P_1(x,u) = \exists y \in B \exists z_n \in C \exists z_c \in D \;\; R(x,y,z_n,z_c,u) = \\
= x^{И}u^1 \vee x^{Р}u^2 \vee x^{Д}u^3 \vee x^{В}u^4 \vee x^{Т}u^5 \vee x^{П}u^6 \vee \\
\vee x^{И}u^7 \vee x^{Р}u^8 \vee x^{Д}u^9 \vee x^{В}u^{10} \vee x^{Т}u^{11} \vee x^{П}u^{12}; \\
P_2(y,u) == \exists x \in A \exists z_n \in C \exists z_c \in D \;\; R(x,y,z_n,z_c,u) = \\
= y^{Е}u^1 \vee y^{Е}u^2 \vee y^{Е}u^3 \vee y^{Е}u^4 \vee y^{Е}u^5 \vee y^{Е}u^6 \vee \\
\vee y^{М}u^7 \vee y^{М}u^8 \vee y^{М}u^9 \vee y^{М}u^{10} \vee y^{М}u^{11} \vee y^{М}u^{12}; \\
P_3(z_n,u) = \exists x \in A \exists y \in B \; \exists z_n \in C \;\; R(x,y,z_n,z_c,u) = \\
= z_n^{\text{ЫЙ}}u^1 \vee z_n^{\text{ОГО}}u^2 \vee z_n^{\text{ОМУ}}u^3 \vee z_n^{\text{ЫЙ}}u^4 \vee z_n^{\text{ЫМ}}u^5 \vee z_n^{\text{ОМ}}u^6 \vee \\
\vee z_n^{\text{ЫЕ}}u^7 \vee z_n^{\text{ЫХ}}u^8 \vee z_n^{\text{ЫМ}}u^9 \vee z_n^{\text{ЫЕ}}u^{10} \vee z_n^{\text{ЫМИ}}u^{11} \vee z_n^{\text{ЫХ}}u^{12}; \\
P_4(z_c,u) = \exists x \in A \exists y \in B \; \exists z_n \in C \;\; R(x,y,z_n,z_c,u) = \\
= z_c^{Ь}u^1 \vee z_c^{Я}u^2 \vee z_c^{Ю}u^3 \vee z_c^{Ь}u^4 \vee z_c^{ЁМ}u^5 \vee z_c^{Е}u^6 \vee \\
\vee z_c^{И}u^7 \vee z_c^{ЕЙ}u^8 \vee z_c^{ЯМ}u^9 \vee z_c^{И}u^{10} \vee z_c^{ЯМИ}u^{11} \vee z_c^{ЯХ}u^{12};
\end{gathered} \tag{4.2}$$

Образуя конъюнкцию этих предикатов, получаем предикат модели реляционной сети для согласования имени прилагательного *толковый* и имени существительного *словарь*:

$$R(x,y,z_n,z_c,u) = P_1(x,u) \wedge P_2(y,u) \wedge P_3(z_n,u) \wedge P_4(z_c,u).$$

На рис. 4.4 – 4.5 приведены графические представления отношений P_1, P_2 и P_3, соответствующих предикатам $P_1(x,u)$, $P_2(y,u)$ и $P_3(z_n,u)$, в виде двудольных графов.

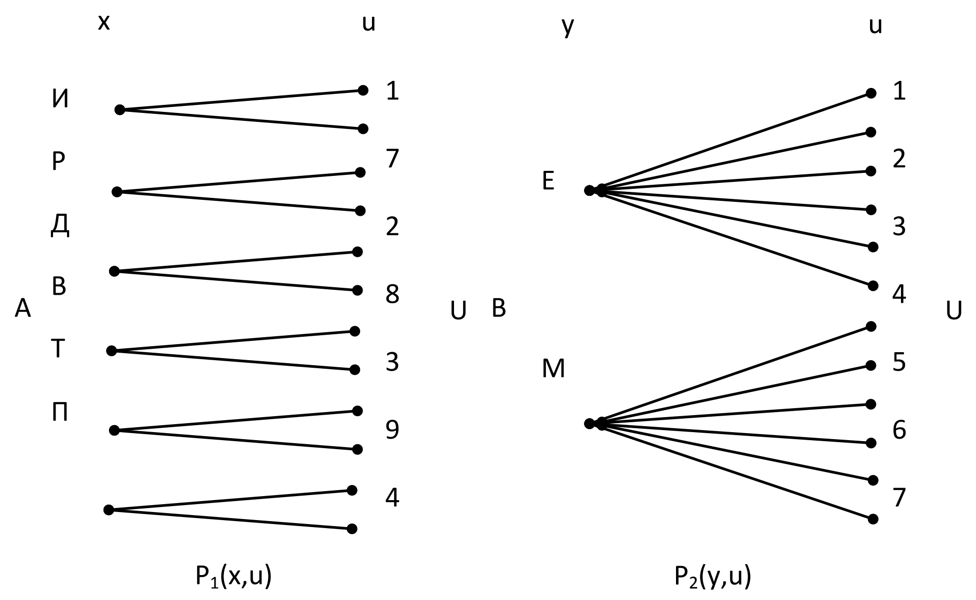

Рис.4.4. Отношения P_1 и P_2

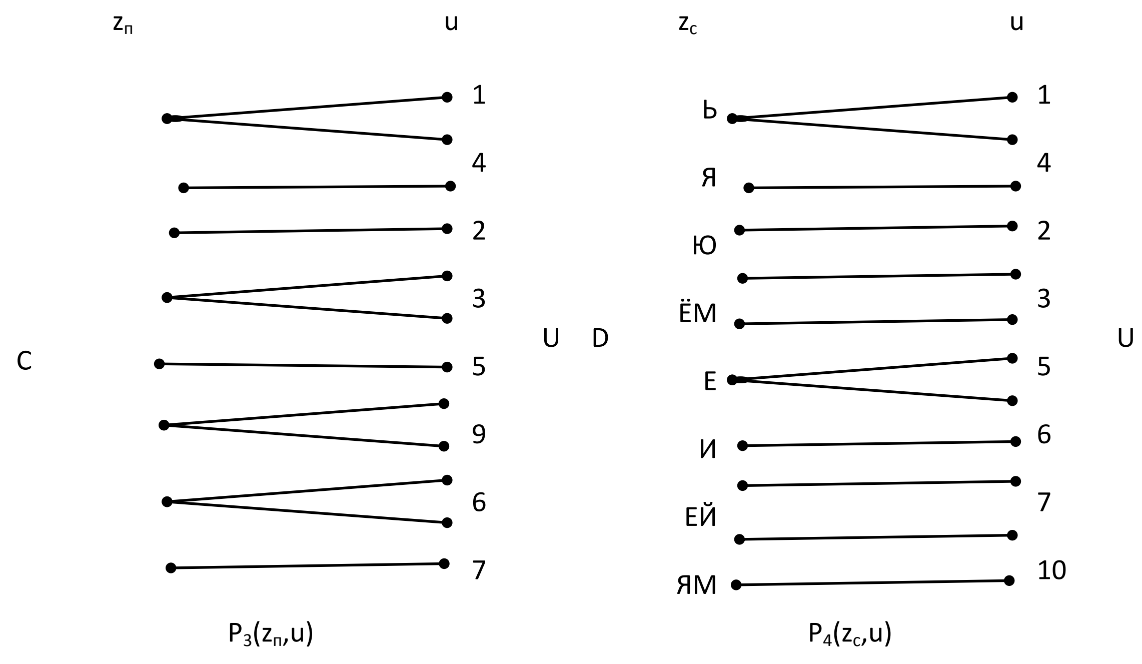

Рис. 4.5. Отношения P_3 P_4

Реляционная сеть для согласования имени прилагательного *толковый* и имени существительного *словарь* приведена на рис. 4.6.

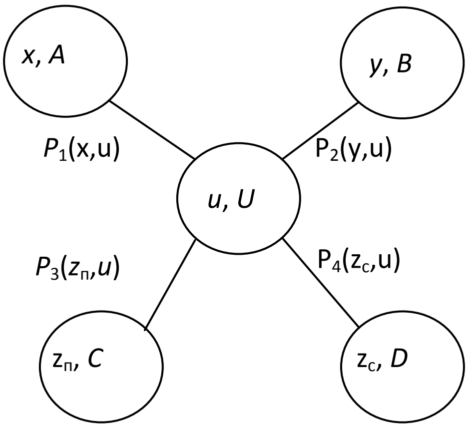

Рис.4.6. Структура реляционной сети

Выполненный в работе анализ естественного языка как лингвистической алгебры свидетельствует о том, что такой подход предоставляет исследователю определенные возможности для формального описания механизма интеллекта, а также его представления в виде реляционных сетей для дальнейшей аппаратной реализации [89].

4.3. Построение направленной схемы реляционной сети отношения эквивалентности

Пусть задано простейшее булево уравнение, выражающее отношение эквивалентности,

$$x \sim y = z, \tag{4.5}$$

где x, y, z – логические переменные, $x, y, z \in \Sigma = \{0,1\}$. Уравнение (4.5) можно представить в виде отношения P с помощью табл. 4.28, которая называется таблицей истинности.

Таблица 4.28. Таблица истинности отношения эквивалентности

x	y	z
0	0	1
0	1	0
1	0	0
1	1	1

Предикат $P(x, y, z)$, соответствующий данному отношению $P = \{(0,0,1),(0,1,0),(1,0,0),(1,1,1)\}$, на языке алгебры конечных предикатов выражается в следующем формульном виде:

$$P(x, y, z) = x^0 y^0 z^1 \vee x^0 y^1 z^0 \vee x^1 y^0 z^0 \vee x^1 y^1 z^1. \qquad (4.6)$$

Далее необходимо провести бинарную декомпозицию формульного описания объекта (4.6). Используя промежуточную переменную $u \in \{0,1,2,3\}$, получим полную (развернутую) бинаризацию:

$$R(x, y, z, u) = x^0 y^0 z^1 u^0 \vee x^0 y^1 z^0 u^1 \vee x^1 y^0 z^0 u^2 \vee x^1 y^1 z^1 u^3.$$

Используя квантор существования по переменной u, получим исходный предикат $P(x, y, z)$:

$$\exists u R(x, y, z, u) = x^0 y^0 z^1 \vee x^0 y^1 z^0 \vee x^1 y^0 z^0 \vee x^1 y^1 z^1. \qquad (4.7)$$

Покажем это:

$$\exists u \in \{0,1,2,3\} R(x, y, z, u) = (x^0 y^0 z^1 0^0 \vee \vee x^0 y^1 z^0 0^1 \vee x^1 y^0 z^0 0^2 \vee x^1 y^1 z^1 0^3) \vee$$
$$\vee (x^0 y^0 z^1 1^0 \vee x^0 y^1 z^0 1^1 \vee x^1 y^0 z^0 1^2 \vee \vee x^1 y^1 z^1 1^3) \vee (x^0 y^0 z^1 2^0 \vee x^0 y^1 z^0 2^1 \vee$$
$$\vee x^1 y^0 z^0 2^2 \vee x^1 y^1 z^1 2^3) \vee (x^0 y^0 z^1 3^0 \vee \vee x^0 y^1 z^0 3^1 \vee x^1 y^0 z^0 3^2 \vee x^1 y^1 z^1 3^3) =$$
$$= (x^0 y^0 z^1 \cdot 1 \vee x^0 y^1 z^0 \cdot 0 \vee x^1 y^0 z^0 \cdot 0 \vee \vee x^1 y^1 z^1 \cdot 0) \vee (x^0 y^0 z^1 \cdot 0 \vee x^0 y^1 z^0 \cdot 1 \vee$$

$$\vee x^1y^0z^0\cdot 0\vee x^1y^1z^1\cdot 0)\vee(x^0y^0z^1\cdot 0\vee\vee x^0y^1z^0\cdot 0\vee x^1y^0z^0\cdot 1\vee x^1y^1z^1\cdot 0)\vee$$
$$\vee(x^0y^0z^1\cdot 0\vee x^0y^1z^0\cdot 0\vee x^1y^0z^0 0\vee\vee x^1y^1z^1\cdot 1)=x^0y^0z^1\vee x^0y^1z^0\vee$$
$$\vee x^1y^0z^0\vee x^1y^1z^1.$$

Исключая с помощью кванторов существования по одной переменной из отношения (4.7), построим бинарные предикаты:

$$P_1(x,u)=x^0u^0\vee x^0u^1\vee x^1u^2\vee x^1u^3=x^0(u^0\vee u^1)\vee x^1(u^2\vee u^3);$$
$$P_2(y,u)=y^0u^0\vee y^1u^1\vee y^0u^2\vee y^1u^3=y^0(u^0\vee u^2)\vee y^1(u^1\vee u^3);\qquad(4.8)$$
$$P_3(z,u)=z^1u^0\vee z^0u^1\vee z^0u^2\vee z^1u^3=z^0(u^1\vee u^2)\vee z^1(u^0\vee u^3).$$

Проводя аналогию со способом построения связей между таблицами в базах данных, можно сказать, что при бинаризации отношения эквивалентности действует полный простой ключ. Т.е. введенная выше дополнительная переменная u представляет собой полный простой ключ и означает порядковый номер набора значений (x,y,z) и вводится с помощью табл. 4.29.

Таблица 4.29

x	y	z	u
0	0	1	0
0	1	0	1
1	0	0	2
1	1	1	3

Поставив в соответствие введенную дополнительную переменную u предметным переменным x, y и z, мы получаем бинарные отношения P_1, P_2 и P_3 (табл. 4.30 – 4.32), соответствующие полученным бинарным предикатам $P_1(x,u)$, $P_2(y,u)$ и $P_3(z,u)$.

Таблица 4.30 Отношение P_1

x	u
0	0
0	1
1	2
1	3

Таблица 4.31 Отношение P_2

y	u
0	0
1	1
0	2
1	3

Таблица 4.32 Отношение P_3

z	u
1	0
0	1
0	2
1	3

Проверим, возможна ли полноценная бинаризация предиката $P(x,y,z)$ без введения промежуточной переменной. Используя кванторы существования по каждой переменной, введем бинарные предикаты $L_1(y,z), L_2(x,z), L_3(x,y)$:

$$L_1(y,z) = \exists x \in \{0,1\} P(x,y,z) = (0^0 y^0 z^1 \vee 0^0 y^1 z^0 \vee$$
$$\vee 0^1 y^0 z^0 \vee 0^1 y^1 z^1) \vee (1^0 y^0 z^1 \vee 1^0 y^1 z^0 \vee 1^1 y^0 z^0 \vee$$
$$\vee 1^1 y^1 z^1) = y^0 z^1 \vee y^1 z^0 \vee y^0 z^0 \vee y^1 z^1 =$$
$$= (y^0 \vee y^1) z^0 \vee (y^0 \vee y^1) z^1 = (y^0 \vee y^1)(z^0 \vee z^1);$$

$$L_2(x,z) = \exists y \in \{0,1\} P(x,y,z) = (x^0 0^0 z^1 \vee x^0 0^1 z^0 \vee$$
$$\vee x^1 0^0 z^0 \vee x^1 0^1 z^1) \vee (x^0 1^0 z^1 \vee x^0 1^1 z^0 \vee x^1 1^0 z^0 \vee$$
$$\vee x^1 1^1 z^1) = x^0 z^1 \vee x^1 z^0 \vee x^0 z^0 \vee x^1 z^1 =$$
$$= (x^0 \vee x^1) z^0 \vee (x^0 \vee x^1) z^1 = (x^0 \vee x^1)(z^0 \vee z^1);$$

$$L_3(x,y) = \exists z \in \{0,1\} P(x,y,z) = (x^0 y^0 0^1 \vee x^0 y^1 0^0 \vee$$
$$\vee x^1 y^0 0^0 \vee x^1 y^1 0^1) \vee (x^0 y^0 1^1 \vee x^0 y^1 1^0 \vee x^1 y^0 1^0 \vee$$
$$\vee x^1 y^1 1^1) = x^0 y^1 \vee x^1 y^0 \vee x^0 y^0 \vee x^1 y^1 =$$
$$= (x^0 \vee x^1) y^0 \vee (x^0 \vee x^1) y^1 = (x^0 \vee x^1)(y^0 \vee y^1).$$

Возьмем произведение бинарных предикатов L_1, L_2, L_3. Получаем:

$$P'(x,y,z) = L_1(y,z) L_2(x,z) L_3(x,y) = ((y^0 \vee y^1) \wedge$$
$$\wedge (z^0 \vee z^1))((x^0 \vee x^1)(z^0 \vee z^1))((x^0 \vee x^1) \wedge$$
$$\wedge (y^0 \vee y^1)) = (x^0 \vee x^1) \wedge (y^0 \vee y^1)(z^0 \vee z^1) =$$

$$= (x^0y^0 \vee x^0y^1 \vee x^1y^0 \vee x^1y^1)(z^0 \vee z^1) = = (x^0y^0z^0 \vee x^0y^1z^0 \vee x^1y^0z^0 \vee x^1y^1z^0) \vee$$
$$\vee (x^0y^0z^1 \vee x^0y^1z^1 \vee x^1y^0z^1 \vee x^1y^1z^1) = = x^0y^0z^0 \vee x^0y^0z^1 \vee x^0y^1z^0 \vee x^0y^1z^1 \vee$$
$$\vee x^1y^0z^0 \vee x^1y^0z^1 \vee x^1y^1z^0 \vee x^1y^1z^1.$$

Результат непосредственной бинаризации $P'(x,y,z)$ не совпадает с исходной дизъюнкцией $P(x,y,z)$. Этим доказано, что бинаризация неполна без введения промежуточной переменной.

Итак, мы построили математическую модель для решения исходного булевого уравнения (4.5). Она характеризуется системой бинарных отношений P_1, P_2 и P_3, задаваемых двудольными графами (рис. 4.7) и формулами соответствующих предикатов $P_1(x,u)$, $P_2(y,u)$ и $P_3(z,u)$ (2).

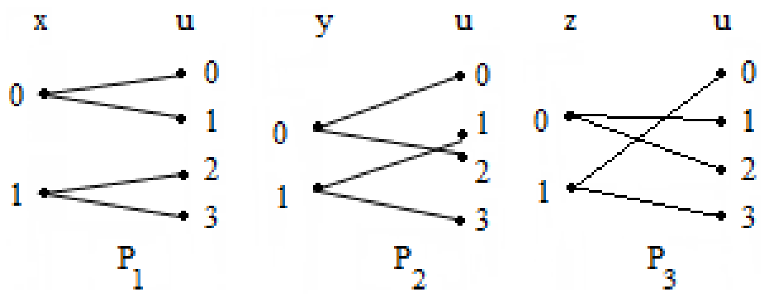

Рис. 4.7. Двудольные графы связей переменных реляционной сети

Образуя конъюнкцию этих предикатов, получаем предикат модели:

$$R(x,y,z,u) = P_1(x,u) \wedge P_2(y,u) \wedge P_3(z,u).$$

Графическое изображение результата бинарной декомпозиции предиката $P(x,y,z)$ представляет собой следующую реляционную сеть (рис. 4.8).

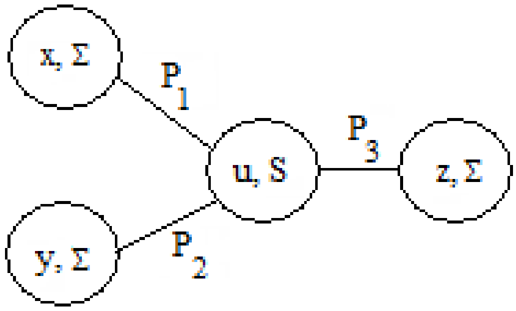

Рис. 4.8. Реляционная сеть для вычисления операции эквивалентности

На схеме реляционной сети показаны все перечисленные ранее переменные x, y, u, z ($x,y,z \in \Sigma = \{0,1\}$, $u \in S = \{0,1,2,3\}$) и связывающие их отношения P_1, P_2, P_3. Наборы (x,Σ), (y,Σ), (u,S), (z,Σ), составленные из переменной и области ее задания, характеризуют полюсы сети, а отношения P_1, P_2, P_3 – ее ветви, которые соединяют полюсы.

Метод потактовой работы реляционных сетей разработан в работе [65]. Рассмотрим примеры потактовой работы построенной реляционной сети. Функция $x \sim y = z$ вычисляется однозначно по набору значений (x, y). Однако в обратную сторону – от значений переменной z к значениям переменных x, y однозначности нет. Чтобы по значению переменной z найти значения переменных x и y, необходимо уточнить значение дополнительной переменной u. Промоделируем потактовую работу реляционной сети для всех наборов значений переменных x, y:

1) $x = 0, y = 0$. Аналитически связь всех переменных логической сети запишется так: $x^0 y^0 z^1 = u^0$.

Рассмотрим оба полутакта работы сети. На первом полутакте заданные значения каждой переменной x, y порождают множества соответствующих им значений переменной u – соответственно $\{0,1\}$ и $\{0,2\}$. На втором полутакте вычисляется пересечение этих множеств, в результате чего получаем однозначное значение вспомогательной переменной u=0. Из отношения P_3 получаем однозначное значение переменной z=1. На рис. 4.9 представлен результат работы реляционной сети на наборе $x=0, y=0$.

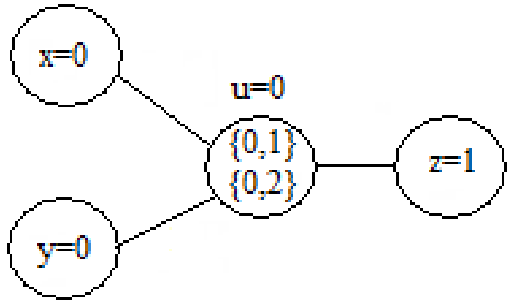

Рис. 4.9. Результат работы реляционной сети на наборе $x=0, y=0$

2) $x=0, y=1$, аналитическая связь – $x^0 y^1 z^0 = u^1$. На рис. 4.10 представлен результат работы реляционной сети на наборе $x=0, y=1$.

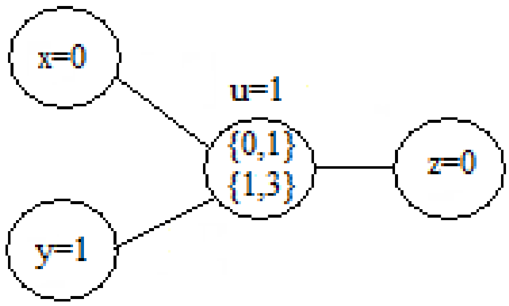

Рис. 4.10. Результат работы реляционной сети на наборе $x=0, y=1$

3) $x=1, y=0$, аналитическая связь – $x^1 y^0 z^0 = u^2$. На рис. 4.11 представлен результат работы реляционной сети на наборе. $x=1, y=0$

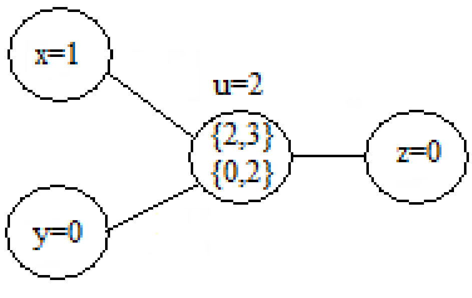

Рис. 4.11. Результат работы реляционной сети на наборе $x=1, y=0$

4) $x=1, y=1$, аналитическая связь – $x^1 y^1 z^1 = u^3$. На рис. 4.12 представлен результат работы реляционной сети на наборе $x=1, y=1$.

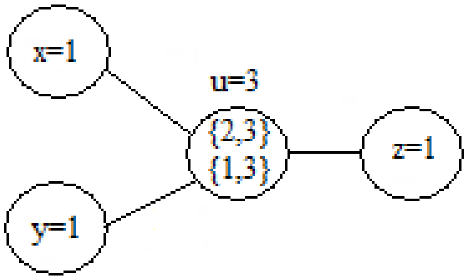

Рис. 4.12. Результат работы реляционной сети на наборе $x=1, y=1$

Т.о., построенная модель реляционной сети аналитически протестирована для всех наборов значений предметных переменных x, y.

В процессе работы реляционной сети по всем ее ветвям происходит двустороннее движение информации, сопровождаемое ее преобразованием. Обработка знаний в ветвях реляционной сети осуществляется линейными логическими операторами. Каждая ветвь реляционной сети представляет собой двунаправленную дугу, которая описывается парой линейных логических операторов. Т.о., функционирующая реляционная сеть представляет собой систему взаимодействующих линейных логических операторов.

Разработке теории линейных логических операторов посвящены работы [25, 59, 88]. Линейными логическими операторами $L(A)=B$ и $L(B)=A'$ с ядром $F(x,y)$ называются преобразования

$$B(y) = \exists x \in M\,(F(x,y) \wedge A(x)), \tag{4.8}$$

$$A'(x) = \exists y \in N(F(x,y) \wedge B(y)). \tag{4.9}$$

Преобразование вида (4.8) представляет собой образ множества $A \subseteq M$ относительно некоторого отображения $f(x) = y$, т.е. множество $B \subseteq N$, образованное из всех образов предметов, принадлежащих множеству A. Преобразование вида (4.9) представляет собой прообраз множества $B \subseteq N$ относительно некоторого отображения $f(x) = y$, т.е. множество $A' \subseteq M$, образованное из всех прообразов предметов, принадлежащих множеству B.

Для того чтобы построенная выше модель реляционной сети функционировала, т.е. чтобы из нее можно было бы извлечь некоторые знания, необходимо решать систему логических уравнений:

$$\begin{cases} P_1(x,u) = 1, \\ P_2(y,u) = 1, \\ P_3(z,u) = 1. \end{cases}$$

Приведем пример функционирования построенной выше модели реляционной сети для дуги, которая описывается предикатом:

$$P_1(x,u) = x^0 u^0 \vee x^0 u^1 \vee x^1 u^2 \vee x^1 u^3.$$

Поставим задачу решения логического уравнения

$$P_1(x,u) = 1. \tag{4.10}$$

Решение логического уравнения (4.10) сводится к нахождению образа или прообраза некоторого множества относительно отображения $f(x) = u$ (рис. 4.13).

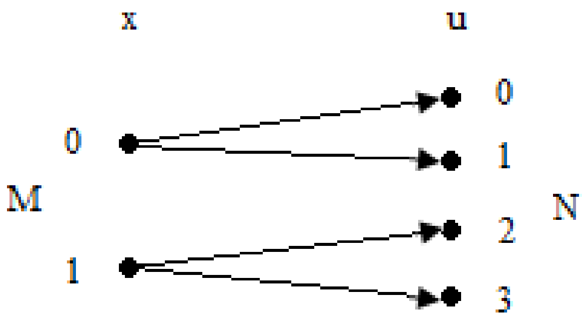

Рис. 4.13. Отображение $f(x) = u$

Рассмотрим примеры решения логического уравнения (4.10):

1) Дано: $x = 1$. Необходимо отыскать всевозможные значения аргумента u, т.е. найти образ одноэлементного множества $A = \{1\}$ относительно отображения $f(x) = u$: $A = \ 1 \xrightarrow{f} B = \ 2,3$ (рис. 4.14).

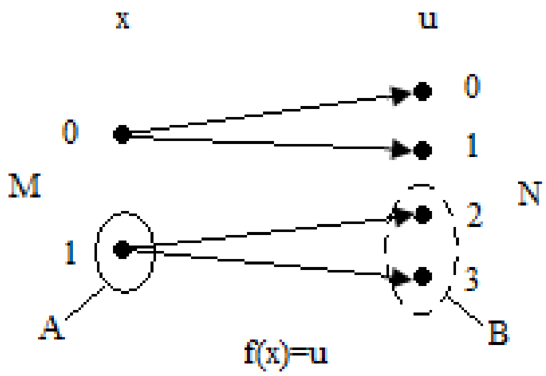

Рис. 4.14. Образ множества $A = \ 1 \xrightarrow{f} B = \ 2,3$

По формуле (4.8) имеем:

$$B(u) = \exists x \in \{1\}\ (F(x,u) \wedge A(x)) = = F(1,u)A(1) = F(1,u) \cdot 1 =$$
$$= 1^0 u^0 \vee 1^0 u^1 \vee 1^1 u^2 \vee 1^1 u^3 == u^2 \vee u^3.$$

2) Дано: $u = 3$. Необходимо отыскать всевозможные значения аргумента x, т.е. найти прообраз одноэлементного множества $B = \{3\}$ относительно

отображения $f(x)=u$: $B = \ 3 \xrightarrow{f} A' = \ 1$ (рис. 4.15).

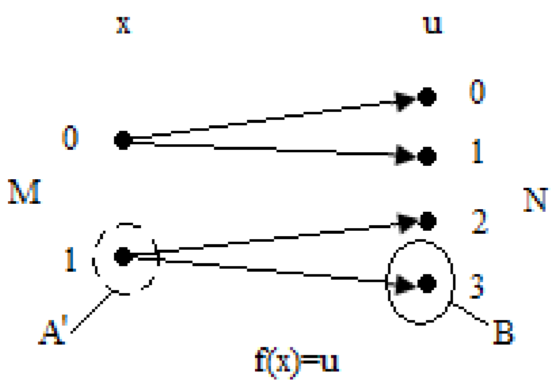

Рис. 4.15. Прообраз множества $B = \ 3 \xrightarrow{f} A' = \ 1$

По формуле (4.9) имеем:

$$A'(x) = \exists u \in \{3\}\,(F(x,u) \wedge B(u)) = F(x,3)B(3) == F(x,3)\cdot 1 = x^1.$$

3) Дано: $x \in \{0,1\}$. Необходимо отыскать всевозможные значения аргумента u, т.е. найти образ множества $A = \{0,1\}$ относительно отображения $f(x)=u$: $A = \ 0,1 \xrightarrow{f} B = \ 0,1,2,3$ (рис. 4.16).

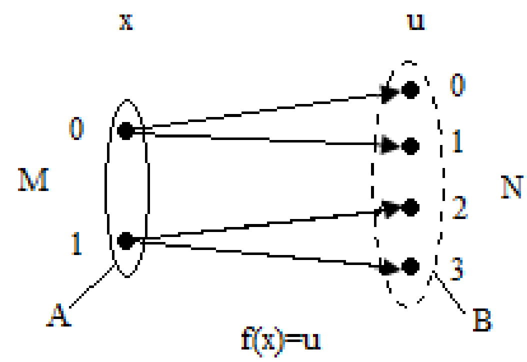

Рис. 4.16. Образ множества $A = \ 0,1 \xrightarrow{f} B = \ 0,1,2,3$

По формуле (4.8) имеем:

$$B(u) = \exists x \in \{0,1\}\,(F(x,u) \wedge A(x)) =$$
$$= F(0,u)A(0) \vee F(1,u)A(1) = F(0,u)\cdot 1 \vee F(1,u)\cdot 1 =$$
$$= (0^0u^0 \vee 0^0u^1 \vee 0^1u^2 \vee 0^1u^3) \vee \vee(1^0u^0 \vee 1^0u^1 \vee 1^1u^2 \vee 1^1u^3) = u^0 \vee u^1 \vee u^2 \vee u^3.$$

4) Дано: $u \in \{1,2\}$. Необходимо отыскать всевозможные значения аргумента x, т.е. найти прообраз множества $B = \{1,2\}$ относительно отображения $f(x) = u$: $B = \ 1,2 \ \xrightarrow{f} A' = \ 0,1$ (рис. 4.17).

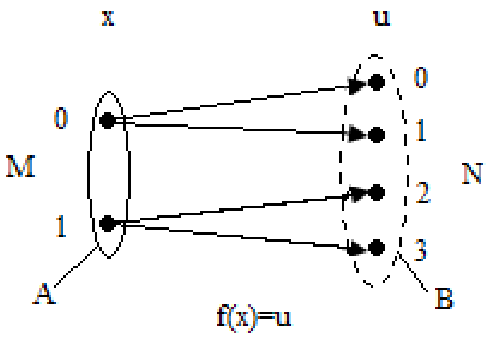

Рис. 4.17. Прообраз множества $B = \ 1,2 \ \xrightarrow{f} A' = \ 0,1$

По формуле (4.9) имеем:

$$A'(x) = \exists u \in \{1,2\}\,(F(x,u) \wedge B(u)) =$$
$$= F(x,1)B(1) \vee F(x,2)B(2) = F(x,1)\cdot 1 \vee F(x,2)\cdot 1 = = (x^0 1^0 \vee x^0 1^1 \vee x^1 1^2 \vee x^1 1^3) \vee$$
$$\vee(x^0 2^0 \vee x^0 2^1 \vee x^1 2^2 \vee x^1 2^3) = x^0 \vee x^1.$$

Направленная схемная реализация связи между предметными переменными x, y и u промоделированной реляционной сети представлена на рис. 4.18.

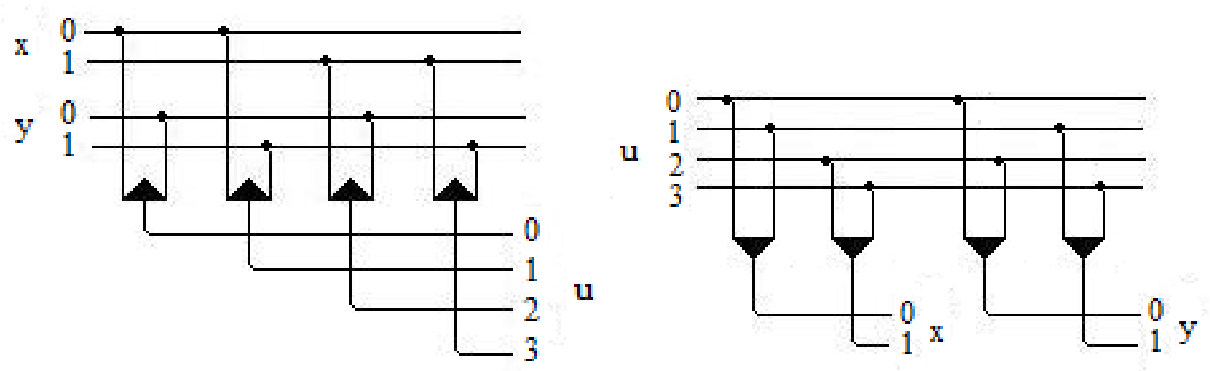

Рис. 4.18. Направленные схемы связи переменных x, y и u, прямая и обратная

Направленная схемная реализация связи между предметной переменой z и дополнительной переменной u реляционной сети представлена на рис. 4.19.

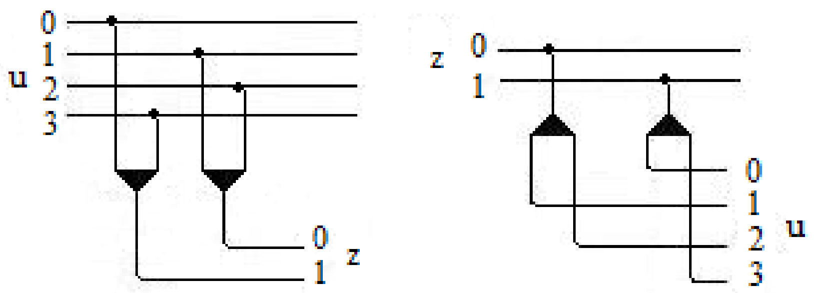

Рис. 4.19. Направленные схемы связи переменных z и u, прямая и обратная

Группы горизонтальных линий на рис. 4.18 – 4.19 представляют собой шины переменных x, y, z, u. Каждая линия шины предназначена для передачи своего сигнала и имеет два состояния – сигнал или его отсутствие. Черные треугольники изображают, в зависимости от направления, операцию дизъюнкции или конъюнкции:

$$x^0 y^0 = u^0,\ x^0 y^1 = u^1,\ x^1 y^0 = u^2,\ x^1 y^1 = u^3;$$

$$u^0 \vee u^1 = x^0,\ u^2 \vee u^3 = x^1,\ u^0 \vee u^2 = y^0,\ u^1 \vee u^3 = y^1;$$

$$u^0 \vee u^3 = z^1,\ u^1 \vee u^2 = z^0;$$

$$z^1 = u^0,\ z^0 = u^1,\ z^0 = u^2,\ z^1 = u^3.$$

Направленная схемная реализация реляционной сети представлена на рис. 4.20 и рис. 4.21.

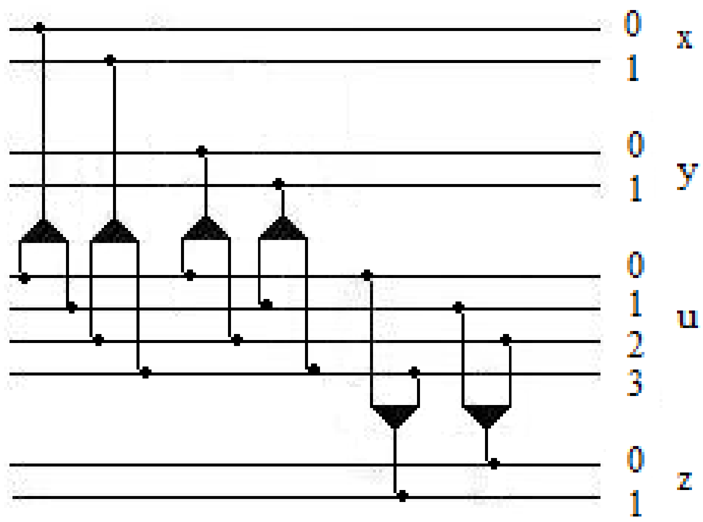

Рис. 4.20. Направленная схема реляционной сети (прямая)

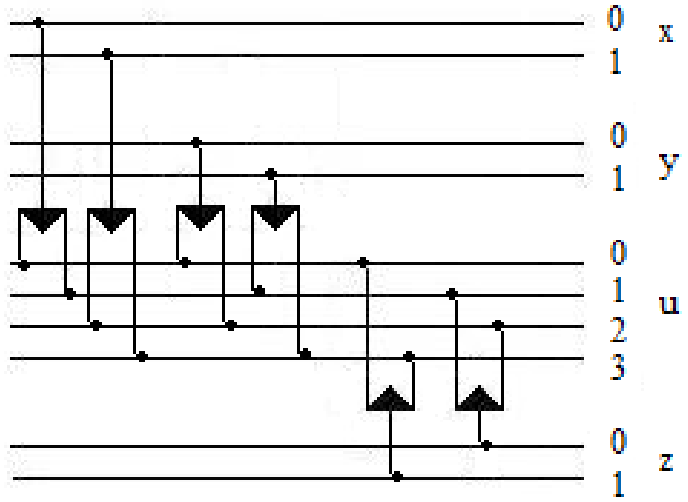

Рис. 4.21. Направленная схема реляционной сети (обратная)

По аналогии с построенной реляционной сетью для отношения эквивалентности можно построить сети для любых логических операций и использовать их в качестве элементарных строительных блоков для моделирования и параллельной обработки любых логических отношений.

4.4 Программная реализация и практические применения реляционных сетей

Для тестирования работы реляционных сетей была разработана программная реализация, которая состоит из двух частей. Первая часть представляет собой программную реализацию реляционной сети и реализована на платформе .Net, а ее графическая часть с использованием технологии Microsoft Silverlight. Интерфейс редактора реляционных сетей представлен на рис. 4.22.

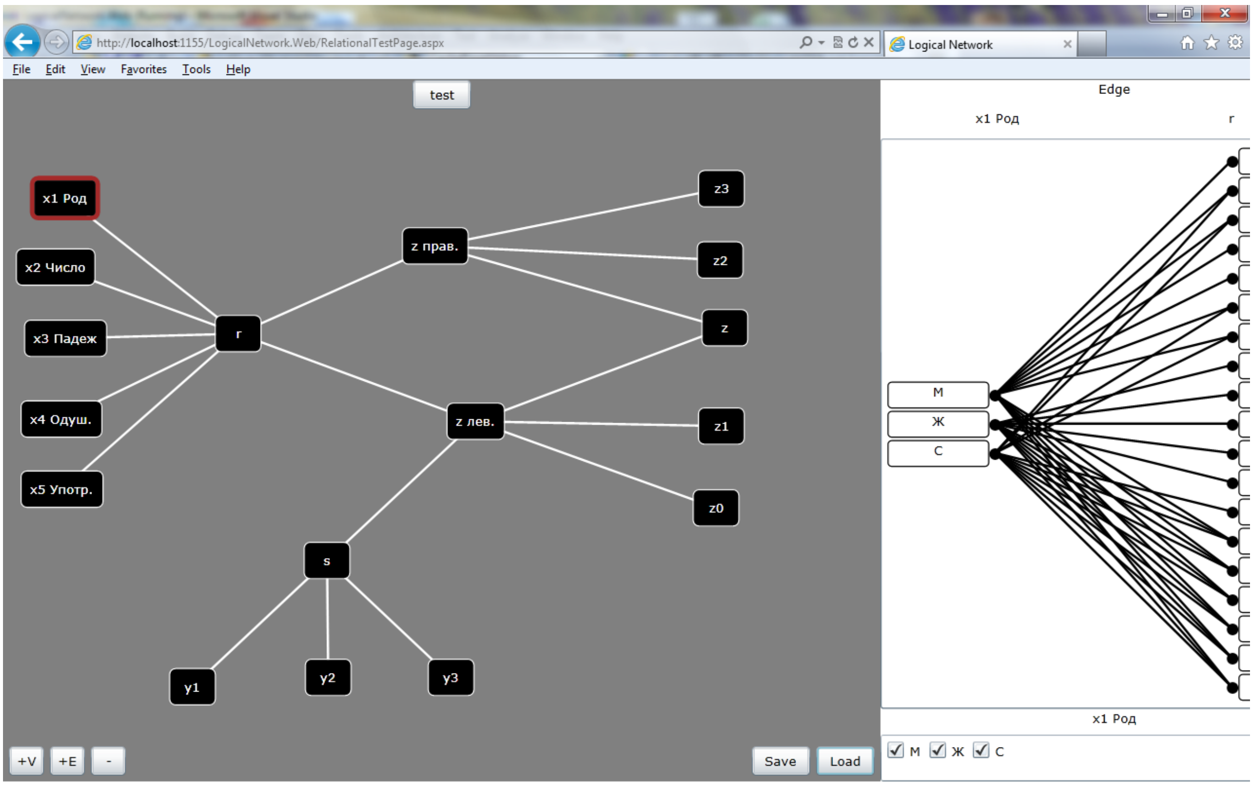

Рис. 4.22. Интерфейс редактора реляционных сетей

В редакторе реляционных сетей можем строить и тестировать реляционные сети для различных частей речи естественных языков. Отношения между предметными переменными реляционной сети настраиваются с помощью графического интерфейса в веб-браузере. Алгоритм работы реляционных сетей реализован как отдельный сервис, который будет доступен в сети Интернет. Для удобства работы с приложением существует возможность задавать различные формы работы сети. После их создания пользователь может сохранить каждую форму в виде отдельного XML-файла, который будет храниться на жестком диске сервера. Для использования сети через веб-сервис необходимо будет передать на сервер данные о том, какую из созданных форм использовать и задать для выбранной сети параметры начального состояния. Кроме того, в редакторе реляционных сетей разработан модуль для идентификации частей речи в текстах естественного языка. На рис. 4.23 приведен пример идентификации имени прилагательного и определения всех его признаков.

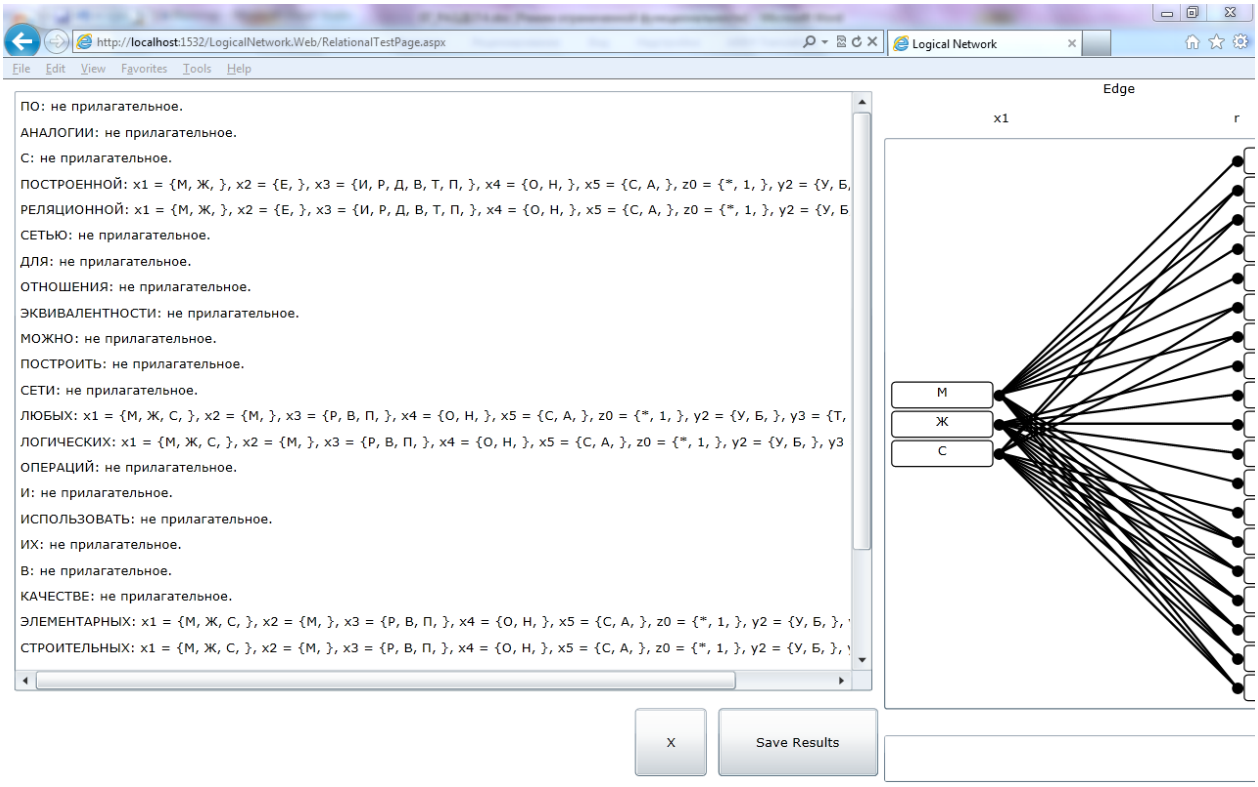

Рис. 4.23. Идентификация имени прилагательного в редакторе реляционных сетей

Вторая часть программной реализации представляет собой веб-приложение. На рис. 4.24 приведен интерфейс модуля статистики поисковой выборки. У пользователя есть возможность задать множество тематических словоформ, которое после подтверждения будет передано через веб-сервис в реляционную сеть, которая обрабатывает исходные данные и выдает результат.

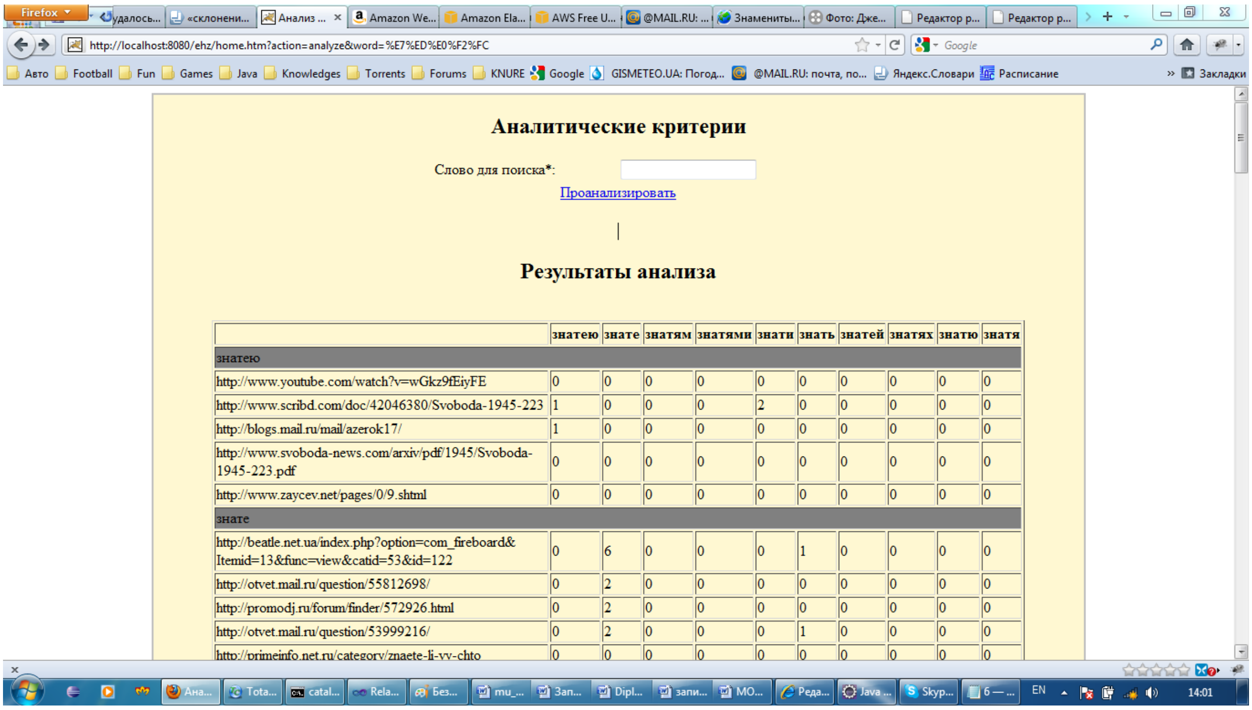

Рис.4.24. Интерфейс модуля статистики поисковой выборки

На основании этого результата будет сформирован ряд запросов в поисковую систему. Из полученного в результате поиска списка сайтов система отберет первые *N* страниц и загрузит их содержимое, которое будет проанализировано. Пользователь сможет задавать различные критерии для анализа, такие как количество повторений семантического ядра в тексте страницы, анализ использования тегов, анализ изображений. Данная часть системы реализована на языке Java с использованием Spring Framework. Интерфейс представляет собой веб-страницу, которая создана на основе технологии JSP и обработана с помощью Spring MVC. Для удобства работы реализована возможность сохранять полученную статистику в различных БД. Отсутствие привязки к конкретной СУБД обеспечивает технология Hibernate.

Разработанное программное обеспечение имеет практическое значение при составлении семантического ядра для оптимизации тематических сайтов и повышения их рейтинга при ранжировании в выдаче поисковых систем. Принцип работы системы заключается в том, что она получает на вход множество тематических слов, ассоциированных с тематикой ресурса. Система находит множество различных словоформ с помощью решения системы линейных логических уравнений реляционных сетей и на основании полученных результатов производит запросы в поисковые системы. Поисковые результаты анализируются по указанным статистическим критериям и отображаются пользователю в графическом виде. На основании этого специалист по оптимизации и продвижению сайтов корректирует семантическое ядро сайта и формирует задание для копирайтеров и веб-разработчиков.

Результаты исследований также были использованы для поддержки конструкторских решений системы "СМ РЭС", которая предназначена для мониторинга основных технических параметров нормального и аварийного режимов распределительных электрический сетей 10 (6) кВ (наличие напряжения, появление тока межфазного короткого замыкания), а также метеорологических условий (температуры и относительной влажности воздуха) в местах эксплуатации и передачи данных на диспетчерский пункт через GSM-сети действующих операторов. Система СМ РЭС применяется для оптимизации поиска и локализации места повреждения воздушных линий электропередачи напряжением 10(6) кВ, а также для предупреждения аварийных состояний в результате влияния климатических факторов.

В состав системы СМ РЭС входит блок диспетчерский, который располагается на диспетчерском пункте районных электрических сетей и блоки БВ (от 1 до 98), которые располагаются на опорах воздушной линии электропередачи 10 (6) кВ. Между блоком диспетчерским и блоками БВ обеспечивается беспроводная сотовая связь для передачи информации из блоков БВ.

Система СМ РЭС обеспечивает выполнение следующих функций:

– фиксацию и передачу на блок диспетчерский аварийной информации о нарушении целостности или отключении отдельных участков контролируемой линии электропередачи, межфазных замыканиях, превышении заданных критических значений метеопараметров;
– фиксацию и передачу на блок диспетчерский информации о состоянии контролируемой линии электропередачи с заданной периодичностью;
– измерение температуры и влажности в местах установки блоков БВ;
– прием и обработку информации с распознаванием адреса отправителя;
– программную установку периода сбора и передачи информации;
– усреднение результата за период сбора данных.

Принцип действия системы поясняет структурная схема, приведенная на рис. 4.25.

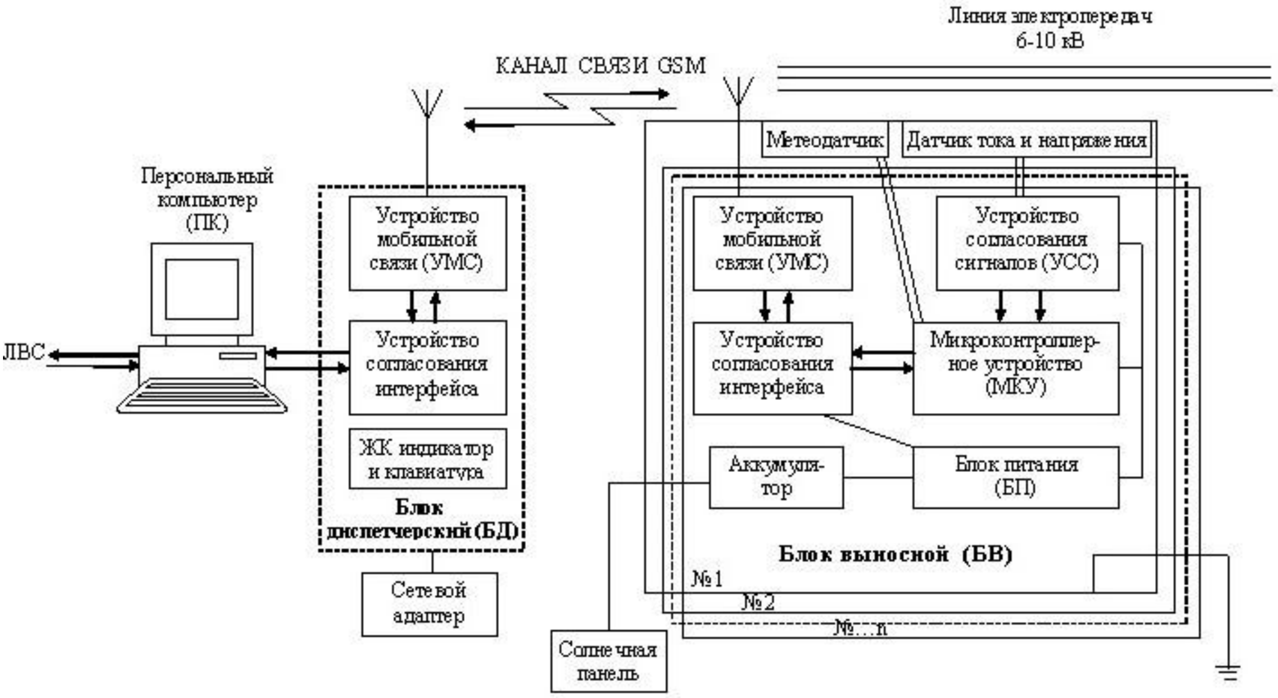

Рисунок 4.25. Структурная схема системы СМ РЭС

Для системы СМ РЭС было разработано программное обеспечение с использованием визуализации в виде реляционных сетей, интерфейс которого приведен на рис. 4.26.

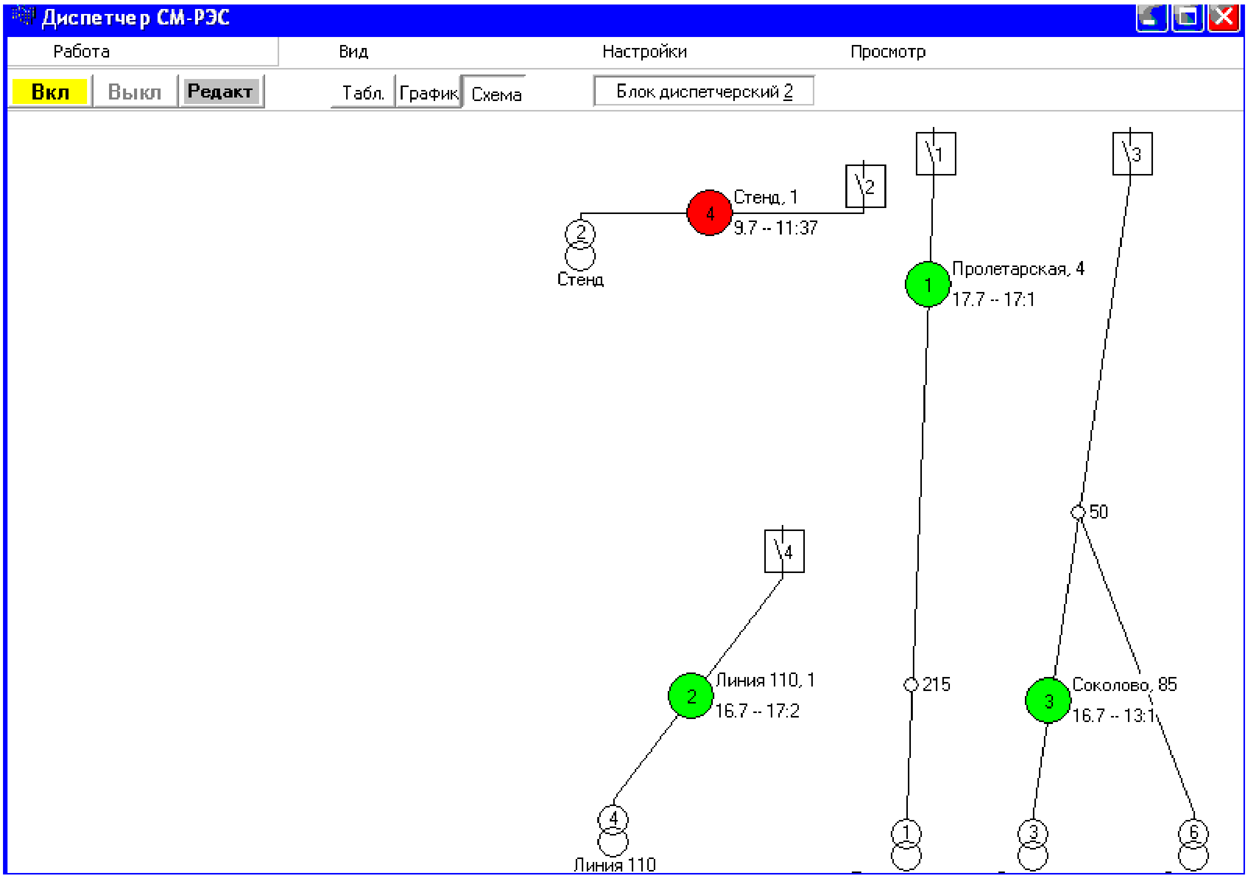

Рисунок 4.26. Интерфейс «Диспетчер_РЭС»

ВЫВОДЫ

В исследовании решена актуальная научно-практическая задача построения комплекса методов и моделей для формализации различных лингвистических структур естественного языка с помощью аппарата алгебры конечных предикатов и реляционных сетей для расширения возможностей естественно-языкового интеллектуального пользовательского интерфейса. Получены следующие основные научные результаты:

1. Модели линейных логических операторов первого и второго рода, описывающие функционирование реляционных сетей. Полученные модели базируются на теории линейных логических операторов. Введение дуальных логических операторов упрощает разработку и тестирование реляционных сетей.

2. Метод построения логической ассоциативной структуры реляционной сети, который позволяет повысить быстродействие параллельного ассоциативного поиска решений по минимальному числу таблиц при составлении семантического ядра для оптимизации тематических сайтов и повышения их рейтинга при ранжировании в выдаче поисковых систем.

3. Метод построения предикатной модели предложений естественного языка путем введения дополнительных и несущественных предметных переменных. Метод обеспечивает однозначность при решении задач анализа и синтеза предложений, т.е. их понимание и формализацию.

4. Получил дальнейшее развитие математический аппарат для анализа структуры естественного языка – лингвистическая алгебра. Показано, что естественный язык имеет двухъярусную структуру. Первый ярус представлен алгеброй предикатов, второй - алгеброй предикатных операций. Исследованы возможности лингвистической алгебры как булевой алгебры.

ПЕРЕЧЕНЬ ЛИТЕРАТУРЫ

1. Алисейко Е.В. Модели семантической эквивалентности данных в экспертных системах (на примере АСУ природоохранного комплекса): Дисс. на соиск. уч. степ. канд. тех. наук. (05.25.05). – Харьков. ХИРЭ, 1990. – 131 с.

2. Анисимов А.В. Компьютерная лингвистика для всех: Мифы. Алгоритмы. Язык / А.В. Анисимова – К.: Наукова думка, 1991. – 208 с.

3. Апресян Ю.Д. Избранные труды: В 2 т. – Т. 1: Лексическая семантика (синонимические средства языка) / Ю.Д. Апресян – М.: Наука, 1995. – 472 с.

4. Апресян Ю.Д. Избранные труды: В 2 т. – Т. 2: Интегральное описание языка и системная лексикография. / Ю.Д. Апресян –М.: Наука, 1995.–767 с.

5. Белоногов Г.Г. Языковые средства автоматизированных информационных систем / Г.Г. Белоногов, Б. А. Кузнецов – М.: Наука. Главная редакция физико–математической литературы, 1983. – 288 с.

6. Библиотеки и ассоциации в меняющемся мире: новые технологии и новые формы сотрудничества // Материалы 3–й международной конференции “Крым – 96”. – Крым, 1996. – Т. 2. – 254 с.

7. Бондарев В. М. Разработка и исследование методов построения и решения на ЭВМ лингвистических уравнений (на примере морфологии имен существительных русского языка): Дисс. ... канд. техн. наук: 05.13.06. – Харьков, 1982. – 195 с.

8. Бондаренко М.Ф. Автоматическая обработка информации на естественном языке: Учеб. пособие. / М.Ф. Бондаренко, А.Ф. Осыка – К.: УМК ВО, 1994. – 144 с.

9. Бондаренко М.Ф. Математические модели морфологических отношений и их применение для автоматизации обработки речевых сообщений. – Дисс. ... докт. техн. наук: 05.13.01. – Харьков, 1984.– 349 с.

10. Бондаренко М.Ф. Модели языка / М.Ф. Бондаренко, В.А. Чикина,

Ю.П. Шабанов–Кушнаренко // Бионика интеллекта. – 2004. – №1 (61). – С.27–37.

11. Бондаренко М.Ф. О булевых реляционных сетях / М.Ф. Бондаренко, И.В. Каменева, Н.Е. Русакова, Ю.П. Шабанов–Кушнаренко, И.Ю. Шубин // Бионика интеллекта. – 2011. – № 1. – С. 3–7.

12. Бондаренко М.Ф. О мозгоподобных структурах / М.Ф. Бондаренко, Н.Е. Русакова, Ю.П. Шабанов–Кушнаренко // Бионика интеллекта. – 2010. – № 2. – С.68–73.

13. Бондаренко М.Ф. О мозгоподобных ЭВМ / М.Ф. Бондаренко, З.В. Дударь, И.А. Ефимова, В.А. Лещинский, С.Ю. Шабанов–Кушнаренко // Радиоэлектроника и информатика. – 2004. – № 2. – С. 89–105.

14. Бондаренко М.Ф., Шабанов–Кушнаренко Ю.П. Мозгоподобные структуры / Справочное пособие, Т. 1. Под ред. акад. НАН Украины И.В. Сергиенко – К.: Наукова думка, 2011. – 460 с.

15. Бондаренко М.Ф., Шабанов–Кушнаренко Ю.П. Теория интеллекта. Учебник – Харьков: изд–во СМИТ, 2007 – 576 с.

16. Бондаренко М.Ф. Булева структура текста / М.Ф. Бондаренко, Ю.П. Шабанов–Кушнаренко, Н.В. Шаронова // Бионика интеллекта. – 2010. – № 3. – С. 14 – 19.

17. Бондаренко М.Ф. Ситуационно–текстовый предикат / М.Ф. Бондаренко, Ю.П. Шабанов–Кушнаренко, Н.В. Шаронова // Бионика интеллекта. – 2010. – № 3. – С. 20 – 25.

18. Бондаренко М.Ф. Инфраструктура анализа логических ассоциативных отношений / М.Ф. Бондаренко, В.И. Хаханов, И.А. Лещинская, Н.Е. Русакова// Радиоэлектроника и информатика. – 2010. – № 2. – С. 43 – 54.

19. Бондаренко М.Ф. О реляционных сетях / М.Ф. Бондаренко, Н.П. Кругликова, И.А. Лещинская, Н.Е. Русакова, Ю.П. Шабанов–Кушнаренко // Бионика интеллекта. – 2010. №3 (74). – С. 8 – 13.

20. Бондаренко М.Ф. Об алгебре одноместных предикатов /

М.Ф. Бондаренко, Н.П. Кругликова, И.А. Лещинская, Н.Е. Русакова, Ю.П. Шабанов–Кушнаренко // Бионика интеллекта. – 2010. №2 (73). – С. 62 – 67.

21. Будущее искусственного интеллекта / Под ред. Поспелова Д.А., Левина К.Е. – М.: Наука, 1991. – 302 с.

22. Бургин М.С. Кванторы в теории свойств // Нестандартная семантическая неклассическая логика. – М., 1986. – С. 99–107.

23. Вейценбаум Дж. Возможности вычислительных машин и человеческий разум. От суждений к вычислениям: Пер. с англ. / под ред. Горелика А.Л.– М.: Радио и связь, 1982. – 368 с.

24. Вейшедл Р.М. Представление знаний и обработка естественных языков // ТИИЭР. 1986. – т. 74. № 7. Обработка естественных языков. – С. 11–30.

25. Вечирская И.Д. Линейные логические преобразования и их применение в искусственном интеллекте. Дисс. … канд. техн. наук: 05.13.23; – Защищена 27.03.2008; Утв. 21.05.2008. – Харьков, 2008 – 153 с.

26. Виноград Т. Программа, понимающая естественный язык. – М.: Мир, 1976. – 294 с.

27. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. - СПб: Питер, 2000. – 384 с.

28. Гаврилова Т.А., Червинская К.Р. Извлечение и структурирование знаний для экспертных систем. - М.: Радио и связь, 1992. 200 с.

29. Гарбузов Б.Ю. Приближенные методы решения лингвистических уравнений и их применение в информационных системах. – Дисс. ... канд. техн. наук: 05.25.05. – Харьков,1989. – 230 с.

30. Гвоздинская Н.А., Дударь З.В, Шабанов–Кушнаренко Ю.П. О математическом описании смысла текстов естественного языка // Проблемы бионики. – 1998. – Вып. 48. – С. 52–60.

31. Гиляревский Р.С., Нюкша Ю.П. и др. / Под редакцией Попова В.В. Библиотеки и библиотечное дело США: комплексный подход. – М.:

Издательская фирма “Логос”, 1992. – 296 с.

32. Гладкий А.В. Элементы математической лингвистики. – М.: Наука, 1980. – 320 с.

33. Дударь З.В. Математические модели флективной обработки словоформ и их использование в системах автоматической обработки текстов русского языка. – Дисс. ... канд. техн. наук: 05.13.01. – Харьков, 1984. – 212 с.

34. Дударь З.В., Кравец Н.С., Шабанов–Кушнаренко Ю.П. О прикладной алгебре предикатных операций // Проблемы бионики. Х.: Изд–во ХТУРЭ - 1998. №49. – С. 78–87.

35. Дударь З.В., Кравец Н.С., Шабанов–Кушнаренко Ю.П. О фундаментальной алгебре предикатных операций // Проблемы бионики. Х.: Изд–во ХТУРЭ. - 1998. №49. – С. 68–77.

36. Дударь З.В., Мельникова Р.В., Шабанов–Кушнаренко Ю.П. Отношения как предмет формульного описания // Проблемы бионики. – 1997. – Вып. 48. – С. 94–103.

37. Дударь З.В., Пославский С.А., Ситникова А.В., Шабанов–Кушнаренко С.Ю., Шуклин Д.Е. О физико–математическом описании смысла текстов естественного языка // Радиоэлектроника и информатика. – 2002. – Вып. № 2. – С. 105–116.

38. Ефимова И.А. Моделирование механизмов естественного языка с помощью бинарных логических сетей И.А. Ефимова, В.А. Лещинский // Вестник НТУ «ХПИ»: Сб. науч. трудов. Тематич. вып.: Новые решения в современных технологиях. – Харьков: НТУ «ХПИ», 2005. – № 57. – С. 3–10.

39. Ефимова И.А. О кванторной алгебре предикатных операций / И.А. Ефимова, В.А. Лещинский // Искусственный интеллект. – Донецк, 2006.– № 4. – С. 603–612.

40. Ефимова И.А. Синтез бинарных логических сетей и особенности их функционирования / И.А. Ефимова, В.А. Лещинский, В.В. Токарев, Г.Г. Четвериков // Бионика интеллекта. –2006. – №2 (65). –С.14–18.

41. Ефимова И.А. О методе построения моделей бинарных логических

сетей / И.А. Ефимова, В.А. Лещинский // Восточно–Европейский журнал передовых технологий. – 2005.– №4.– С. 121–124.

42. Ефимова И.А. О проблемах и перспективах формализации естественного языка средствами алгебры предикатов и предикатных операций // 10–й Междунар. молодежный форум «Радиоэлектроника и молодежь в XXI веке»: Сб. материалов. – Харьков: ХНУРЭ, 2006.– С.197.

43. Ефимова И.А. Об алгебро–логических моделях механизмов естественного языка / И.А. Ефимова // 9–й Междунар. молодежный форум «Радиоэлектроника и молодежь в XXI веке»: Сб. материалов. – Харьков: ХНУРЭ, 2005.– С.377.

44. Ефимова И.А. Построение формальных моделей структур естественного языка с помощью алгебры предикатов и предикатных операций и их реализация в виде бинарных логических сетей / И.А. Ефимова, В.А. Лещинский // Междунар. науч. конф. «MegaLing’2006 – Горизонты прикладной лингвистики и лингвистических технологий»: Сб. докладов. – Симферополь: Изд–во «ДиАйПи», 2006.– С. 207.

45. Єфімова І.О. Моделювання логічних мереж для булевих рівнянь / І.О. Єфімова, В.О. Лещинський // VIII Всеукраїнська (III Міжнар.) студентська наук. конф. з прикладної математики та інформатики: Зб.матеріалів. – Львів: ЛНУ, 2005.– С.128–129.

46. Жолковский А.К. Модель “Смысл – текст” // Энциклопедия кибернетики, 1974. – Т.2. – С. 46.

47. Закревский А.Д. Логические уравнения. – Минск: Наука и техника, 1975. – 92 с.

48. Закревский А.Д. Метод «отражения волн» решения логических уравнений // Прикладные аспекты теории автоматов. – Варна: изд–во БАН, 1975.– Т.2. – С. 81–84.

49. Захарченко П.В. Математические модели корневых структур и их технические приложения. – Дисс. ... канд. техн. наук: 05.13.01. – Харьков, 1983. – 271с.

50. Звегинцев В.А. Мысли о лингвистике. – М.: Изд–во Моск. ун–та. 1996. – 335 с.

51. Иванилов А.А. Метод бинарной декомпозиции функциональных предикатов / А.А. Иванилов // Системи обробки інформації. – 2006. – №6. – С. 86–97.

52. Иванилов А.А. Реляционные алгебры и алгебры предикатов / А.А. Иванилов, Ю.П. Шабанов–Кушнаренко // Восточно–Европейский журнал передовых технологий. – 2007. – № 4/2. – С. 43–48.

53. Искусственный интеллект. – В 3 кн. Кн. 2. Модели и методы: Справочник / Под ред. Д.А. Поспелова – М.: Радио и связь, 1990. – 304 с.

54. Искусственный интеллект: применение в интегрированных производственных системах / Под ред. Э. Кьюсиака: пер. с англ. – М., 1991. – 544 с.

55. Калиниченко О.В. Алгебра идей как аппарат формализации семантики естественного языка в системах искусственного интеллекта. – Дисс. ... канд. техн. наук: 05.13.23. – Харьков, 2004. – 156 с.

56. Калянов Г.Н. Структурный системный анализ. - М.: Лори, 1996. – 226 с.

57. Кандрашина Е.Ю., Литвинцева Л.В., Поспелов Д.А. Представление знаний о времени и пространстве в интеллектуальных системах / Под ред. Д.А. Поспелова. – М., 1989. – 328 с.

58. Ковальски Р. Логика в решении проблем: Пер. с англ. – М., 1990. – 280 с.

59. Козяев Л.Л. Методы формализации и модели морфологических структур и их применение в системах искусственного интеллекта: Дисс. … канд. техн. наук: 05.13.23; – Защищена 12.04.2006; Утв. 8.06.2006. – Харьков, 2006. – 153 с.

60. Кокорева Л.В., Перевозчикова О.Л., Ющенко Е.Л. Диалоговые системы и представление знаний. – Киев: Наукова думка, 1992. – 448 с.

61. Кравец О.А. Математические модели словосочетаний с

инструментальным значением и их использование в информационных системах. Дис... канд. техн. наук: – Х. ХИРЭ. 1992. – 156 с.

62. Кубрякова Е.С. Типы языковых значений. Семантика производного слова. – М.: Наука, 1981. – 199 с.

63. Куратовский К., Мостовский А. Теория множеств. М.: “Мир”, 1970. 413 с.

64. Ландау Э. Основы анализа. – М.: ИЛ, 1947. – 182 с.

65. Леонтьева Н. Н. О компонентах системы понимания текста. – В кн.: Уровни текста и методы его лингвистического анализа. – М.: Наука, 1982.– С. 124–140.

66. Леонтьева Н.Н. Семантика связного текста и единицы информационного анализа. – НТИ, Сер. 2, № 1, 1981. – С. 21–29.

67. Лещинская И.А. Аппаратная реализация модели реляционной сети памятью с ассоциативным доступом / И.А. Лещинская, Н.Е. Русакова,// Друга факультетська наук.–практ. молодіжна школа–семінар «Інформаційні інтелектуальні системи – 2009»: Сб. матеріалів. – Харків, 2009. – С.182–187.

68. Лещинская И.А. Лингвистическая алгебра как аппарат формализации смысла предложений естественного языка / И.А. Лещинская, В.А. Лещинский, Л.Г. Петрова, С.Ю. Шабанов–Кушнаренко // Системи обробки інформації. –2011. – №2 (92). – С. 34–38.

69. Лещинская И.А. Линейные логические операторы первого и второго рода в реляционных сетях / И.А. Лещинская // Системи управління, навігації та зв'язку. – 2010. №2 (14). – С. 214 – 217.

70. Лещинская И.А. О методе построения направленных схем реляционных сетей на примере отношения эквивалентности / И.А. Лещинская // Системи обробки інформації – 2010. – № 1 (82). – С. 75 – 81.

71. Лещинская И.А. О формальной модели реляционной сети синтаксической структуры предложения естественного языка / И.А. Лещинская // 14–й Междунар. молодежный форум «Радиоэлектроника и

молодежь в XXI веке»: Сб. материалов. – Харьков: ХНУРЭ, 2010.– С. 197.

72. Лещинская И.А. Словосочетания естественного языка как реляционные сети / И.А. Лещинская // 17 Міжнар. конф. з автоматичного управління «Автоматика–2010»: Тези доповідей. Т.2. – Харків: ХНУРЕ, 2010. — С. 123–125.

73. Лещинский В.А. Модели бинарных логических сетей и их применение в искусственном интеллекте: Дисс. … канд. техн. наук: 05.13.23; – Защищена 17.01.2007; Утв. 15.03.2007. – Харьков, 2007. – 160 с.

74. Лингвистическая прагматика и общение с ЭВМ. – М., 1989. – 584 с.

75. Лихачёва О.А. Математические модели семантики суффиксального словообразования и их использование в автоматизированных информационных системах: Дисс. на соиск. уч. степ. канд. тех. наук. (05.25.05). – Харьков: ХИРЭ, 1990 – 256 с.

76. Логический подход к искусственному интеллекту: от классической логики к логическому программированию: Пер. с франц. / Тейз А., Грибомон П., Луи Ж. и др. – М., 1990. – 432 с.

77. Любарский Ю.Я. Интеллектуальные информационные системы. – М.: Наука. Гл. ред. физ. – мат. литературы, 1990. – 232 с.

78. Марка Д.А., МакГоуэн К. Методология структурного анализа и проектирования. М.: МетаТехнология, 1993. – 358 с.

79. Мельникова Р.В. Алгебрологические модели морфологии и их применение в логических сетях. Дисс. ... канд. техн. наук: 05.13.23; – Защищена 12.04.2006; Утв. 08.06.2006. – Харьков, 2006. – 160 с.

80. Мендельсон Э. Введение в математическую логику: Пер. с англ. – М., 1984. – 320 с.

81. Моделирование языковой деятельности в интеллектуальных системах / Под ред. Кибрика А.Е., Нариньяни А.С. – М.: Наука, 1987. – 280 с.

82. Морозов А.А., Ященко В.А. Интеллектуализация ЭВМ на базе нового класса нейроподобных растущих сетей. Нац. АН Украины, ин–т пробл. мат. машин и систем. – К.: Тираж, 1997. – 125 с.

83. Нариньяни А.С. Формальная модель: общая схема и выбор адекватных средств. Препринт № 107/ВЦ СО АН СССР. Новосибирск, 1978. – 19 с.

84. Нильсон Н. Принципы искусственного интеллекта. – М.: Радио и связь, 1984. – 374с.

85. Осипов Г.С. Приобретение знаний интеллектуальными системами. – М.: Наука, 1997. – 247 с.

86. Осуга С. Обработка знаний. – М.: Мир, 1989. 291 с.

87. Перспективы развития вычислительной техники: Справ. пособие / Под ред. Смирнова Ю.М. Кн. 2: Интеллектуализация ЭВМ / Кузин Е.С. и др.– М.: Высш. шк., 1989. – 159 с.

88. Перспективы развития вычислительной техники: Справ. пособие / Под ред. Смирнова Ю.М. Кн.1: Информационные семантические системы / Соломатин Н.М. – М.: Высш. шк., 1989. – 128 с.

89. Пронюк А.В. Метод многослойной декомпозиции предикатов и его применение в системах искусственного интеллекта: Дисс. … канд. техн. наук: 05.13.23; – Защищена 07.07.2004; Утв. 16.09.2004. – Харьков, 2004. – 195 с.: ил. – Библиогр.: с.186–195.

90. Ситников Д.Э. Методы решения уравнений алгебры конечных предикатов с параметрами и их применение в информационных системах. Дисс. … канд. техн. наук: 05.25.05; Защищена 6.03.1992; Утв. 12.06.1992. – Харьков, 1992. – 188 с.: ил. – Библиогр.: с. 176–188.

91. Ситникова П.Е, Коваленко А.И. Метод решения логических уравнений с конечными бинарными отношениями в информационной системе Системы обработки информации – Харьков: ХУПС. – 2011. – № 2(92). – С. 49–52.

92. Совпель И.В. Автоматизированная переработка текста на основе воспроизводящего моделирования лингвистических объектов и процессов: Дисс. на соиск. уч. степ. докт. техн. наук. (05.25.05). – Минск: БГУ, 1991 – 361с.

93. Стороженко А.В. Алгебро–логические модели семантики текстов естественного языка: Дисс. на соиск. уч. степ. канд. тех. наук. (01.05.02). – Харьков: ХТУРЭ, 2000 – 185 с.

94. Терзиян В.Я. Теоретическое и экспериментальное исследование проблемы семантического анализа естественно–языковых высказываний: Дисс. на соиск. уч. степ. канд. тех. наук. (05.13.01): В 2–х томах. – Харьков: ХИРЭ. Т.1: Диссертация. – 1984. – 208 с. Т.2: Приложения. – 1984. – 175 с.

95. Уткин А.А. Решение логических уравнений // Автоматизация логического проектирования. – Минск: БГУ. – 1982. – С. 41–58.

96. Уэно Х. Представление и использование знаний. – М.: Мир. 1989. 220 с.

97. Харламов А.А., Ермаков А.Е., Кузнецов Д.М. Технология обработки текстовой информации с опорой на семантическое представление на основе иерархических структур из динамических нейронных сетей, управляемых механизмом внимания // Информационные технологии. 1998. № 2.

98. Хомский Н. Формальные свойства грамматик // Кибернетический сборник. – М.: Физматгиз, Вып.2, 1966. – С. 121–130.

99. Хомский Н., Миллер Дж. Введение в формальный анализ естественного языка // Кибернетический сборник. Новая серия. – М.: Физматгиз, Вып.1, 1965. – С. 229–290.

100. Цинман Л. Л., Сизов В. Г. Лингвистический процессор ЭТАП: дескрипторное соответствие и обработка метафор // Труды Международного семинара “Диалог 2000 по компьютерной лингвистике и ее приложениям”. Т. 2. Протвино 2000. С. 366–369.

101. Чень Ч., Ли Р. Математическая логика и автоматическое доказательство теорем. – М.: Наука, 1983. – 274 с.

102. Четвериков Г.Г., Федорова Т.М., Токарєв В.В., Вечирська І.Д., Лещинская И.А. Концептуально–методологічний підхід до моделювання природної мови алгебро–логічними засобами // Міжнар. наук. конф. «MegaLing’2009 – Горизонти прикладної лінгвістики та лінгвістичних

технологій»: Сб. матеріалів. – К., 2009.– С.68.

103. Шабанов–Кушнаренко Ю.П. Логическая сеть как технология моделирования естественного языка / Шабанов–Кушнаренко Ю.П., Хаханов В.И., Процай Н.Т. и др. // Информационные технологии – в науку и образование: наук.–практ. конф. Харків, 21–22 березня 2005 г.: тези доп. – Харків, 2005. – С. 30–33.

104. Шабанов–Кушнаренко Ю.П., Шаронова Н.В. Компараторная идентификация лингвистических объектов: Монография. – К.: ИСИО, 1993. – 116 с.

105. Шенк Р. Обработка концептуальной информации. – М.: Энергия, 1980. – 198 с.

106. Шуклин Д.Е. Модели семантических нейронных сетей и их применение в системах искусственного интеллекта: Дисс. на соиск. уч. степ. канд. тех. наук. (05.13.23). – Харьков: ХТУРЭ, 2003 – 196 с.

107. Aho A.V., Hopcroft J.E. The Design and Analysis of Computers Algorithms Reading. – M.A.: Addison – Wesley, 1994.

108. Bondarenko M.F. About ‘Similar–To–Brain’ Computers / Bondarenko M.F., Dudar Z.V. // Proc. East–West Design & Test Workshop.– Yalta, Alushta, Crimea, Ukraine, 2004. – P. 251–256.

109. Bondarenko M.F. Logic networks application for computing process organization / Bondarenko M.F., Hahanova I.V. // Радиоэлектроника и информатика. – 2003. – № 3. – С. 150–156.

110. Charniac E., Wilks Y. Computtational Semantics. North – Holland publishing company. – Amsterdam, New – York, Oxford, 1976. – 409 p.

111. Chetverikov G. Methods of synthesizing reversible spatial multivalued structures of language systems / G. Chetverikov , I. Leschinskaya, I. Vechirskaya // Information science & computing. – 2009. – N.15 – P. 32–39.

112. Feigenbaum E., The art of Artificial Intelligence. // Thomes and Case Studies of Knowledge Engineering. IJCAIS. – 1977. – P. 49–65.

113. Fillmor Ch. T. Subjects, Speakers and Roles. – Synthese, 1970, vol.21.

114. Guarino, N., Poli. R. The role of ontology in the information technology // International Journal of Human–Computer Studies. - 1995. - No. 5/6 - P. 623–665.

115. Hundhausen R. Working with Microsoft® Visual Studio® 2008 Team System. – Washington: Microsoft Press, 2008. – 568 pp.

116. Kaiz J. J., Fodor J. A. The Structure of a Semantic Theory. ‘Language’, 1983, vol. 39. – P. 170–210.

117. Khoroshevsky V.F. Knowledge Based Design of Knowledge Based Systems in PiES WorkBench // Proc. of JCKBSE'94, Japan. - 1994 - P. 256–261.

118. Kifer M., Lausen G., Wu J. Logical Foundations of Object–Oriented and Frame–Based Languages // Journal of the ACM. - 1992. - № 28 - P. 436 - 438.

119. Lauriere J.–L. Intelligence Artificielle. – Eyrolles, Paris. 1987.

120. Lenat D. B. "Cyc: A Large–Scale Investment in Knowledge Infrastructure." Communications of the ACM 38. - 1995. - № 11. - P. 124 - 138.

121. Lloyd J. W. Foundations of Logic Programming. – Springer – Verlag, Berlin, 1984. – 230 p.

122. Newell A., Shaw I.C., Simon H.A. Empirical explorations with the logic theory machine // Proc. Western Joint Computer Conf., 1957.– P. 218–239.

123. Nils J. Nilsson. Logic and Artifitial Intellegence. Computer Science Department, Stanford University, Stanford, CA 94305, USA.Schank R. Conceptual Dependency: a Theory of Natural – Language Understanding Cognitivte. – Psychology, 1982. – v.3. – P. 552–631.

124. Pacak M. Morphological analysis. Washington, 93. (Georgetown University. Occasional papers on machine translation N5).

125. Page–Jones M. The Practical Guide to Structured Systems Design. - Englewood Cliffs, NJ: Jourdon Press, 1988 - 320p.

126. Pastre D. Automatic Theorem Proving in Set Theory // A.I. – 1978. – №10. P. 1–27.

127. Ramamoganarao K., Harland J. An Introduction to Deductive Database Languages and Systems // VLDB Journal. - 1994. - No. 3. - P. 107–122.

128. Rauh–Hinden W. A.I. Grading the Hardware and Software Options // System and Software, August 1985. – P. 38–60.

129. Rojas R. Neural Networks. A Systematic Introduction.–Berlin: Springer–Verlag, 1996. – 502p.

130. Schank R. Conceptual Dependency: a Theory of Natural Language Understanding Cognitive. – Psychlogy, 1972. – vol. 3. – №4. – P.552–631.

131. Shabanov–Kushnarenko Yu. / Brainlike computing // Shabanov–Kushnarenko Yu., Klimushev V., Lukashenko O., Nabatova S., Obrizan V., Protsay N. // Proceedings of the East–West Design and Test Workshop. Odessa, Ukraine, September 15 – 19, 2005, p. 274 – 279.

132. Shank R. Natural Language, Philosophy and Artificial Intelligence / Ed. by M.Ringie. N.Y., 1981. – P.221.

133. Slagle J.R., Gardiner O.A., Kyungsook N. Knowledge Specification on an Expert System // IEEE Expert. - 1990. - P.29–38.

134. Stein J. Object–Oriented Programming and Datebase Design // Dr. Dobb's Journal of Software Tools for the Professional Programmer. - 1988. - Vol. 137 - P.18.

135. Synthesis of arithmetic circuits: FPGA, ASIC and embedded systems / Jean–Pierre Deschamps, Gery Jean Antoine Bioul, Gustavo D.Sutter., 2005. – 556pp.