

**Российская
академия
наук**

**Институт
проблем
информатики**

А.В. Колесников, И.А. Кириков

**МЕТОДОЛОГИЯ И ТЕХНОЛОГИЯ РЕШЕНИЯ СЛОЖНЫХ
ЗАДАЧ МЕТОДАМИ ФУНКЦИОНАЛЬНЫХ ГИБРИДНЫХ
ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ**



2007

**Колесников А.В., Кириков И.А. Методология и
технология решения сложных задач методами
гибридных интеллектуальных систем**

РОССИЙСКАЯ АКАДЕМИЯ НАУК
ИНСТИТУТ ПРОБЛЕМ ИНФОРМАТИКИ

А.В. Колесников, И. А. Кириков

**МЕТОДОЛОГИЯ И ТЕХНОЛОГИЯ РЕШЕНИЯ
СЛОЖНЫХ ЗАДАЧ МЕТОДАМИ ФУНКЦИОНАЛЬНЫХ
ГИБРИДНЫХ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ**

Москва
ИПИ РАН
2007

УДК 681.3

*Печатается по решению Ученого совета
Института проблем информатики РАН*

Рецензенты:

доктор технических наук И. О. Фамилия

доктор физико-математических наук, профессор И. О. Фамилия

Колесников А.В., Кириков И.А.

Методология и технология решения сложных задач методами функциональных гибридных интеллектуальных систем. – М.: ИПИ РАН, 2007. – 387 с., ил. – ISBN 978-5-902030-55-3

В книге рассматриваются вопросы решения сложных задач обработки информации и управления в системах поддержки принятия решений. Описываются методы моделирования рассуждений экспертов и лиц, принимающих решения в системах поддержки принятия решений. Рассматриваются модели функциональных гибридных интеллектуальных систем. Предлагаются методология и технология их разработки. Приводится пример решения сложных задач.

Издание предназначено для специалистов в области автоматизированного решения задач обработки информации и управления. Книга также будет полезна аспирантам и студентам, изучающим информатику и системы искусственного интеллекта.

© А.В. Колесников, И.А. Кириков, 2007

ISBN 978-5-902030-55-3 © Институт проблем информатики РАН, 2007

Оглавление

ВВЕДЕНИЕ.....	7
1. СЛОЖНЫЕ ЗАДАЧИ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ.....	10
1.1. Условия увеличения сложности задач в современном обществе.....	10
1.2. Классификация задач. Понятие неоднородности задач.....	15
1.3. Источники неоднородности задач.....	27
1.4. Особенности сложных задач.....	32
1.5. Решение задач в автоматизированном режиме.....	40
1.6. Новые подходы к решению сложных задач в автоматизированном режиме.....	43
2. РЕШЕНИЕ СЛОЖНЫХ ЗАДАЧ В СИСТЕМАХ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ.....	60
2.1. Понятие системы поддержки принятия решений.....	60
2.2. Особенности работы экспертов в системах поддержки принятия решений.....	74
2.3. Режимы и организация работы систем поддержки принятия решений.....	77
2.4. Модели процесса принятия коллективных решений.....	82
2.5. Компьютерные системы поддержки принятия решений.....	86
3. МЕТОДЫ МОДЕЛИРОВАНИЯ РАССУЖДЕНИЙ ЭКСПЕРТОВ ПРИ РЕШЕНИИ ЗАДАЧ В СИСТЕМАХ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ.....	103
3.1. Знания и опыт — основа творческого процесса в системах поддержки принятия решений.....	103
3.2. Моделирование теоретических знаний экспертов.....	110
3.3. Моделирование профессиональных знаний экспертов.....	126
3.3.1. Методы рассуждений в экспертных системах.....	126
3.3.2. Методы вычислений в нечетких системах.....	136
3.3.3. Методы вычислений в искусственных нейронных сетях.....	153
3.3.4. Методы эволюционных вычислений.....	164
3.4. Моделирование профессионального опыта экспертов.....	176
3.5. Моделирование невербальных рассуждений экспертов.....	187

4. МОДЕЛИРОВАНИЕ РАССУЖДЕНИЙ ЛИЦА, ПРИНИМАЮЩЕГО РЕШЕНИЯ, В СЛОЖНЫХ ЗАДАЧАХ СИСТЕМ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ.....	191
4.1. Модели интеграции знаний экспертов.....	191
4.2. Методы интеграции знаний в многоагентных системах.....	206
4.3. Методы интеграции знаний в гибридных системах.....	208
4.4. Интегрированные экспертные системы.....	212
4.5. Методы интеграции знаний в условиях расхождения мнений экспертов.....	215
4.6. Методы гибридных интеллектуальных систем.....	217
4.7. Функциональные гибридные интеллектуальные системы.....	229
5. МОДЕЛИРОВАНИЕ ФУНКЦИОНАЛЬНЫХ ГИБРИДНЫХ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ.....	233
5.1. Концептуальные модели — инструмент представления гибридных знаний.....	233
5.2. Концептуальный базис функциональных гибридных интеллектуальных систем.....	236
5.3. Моделирование состояний и ситуаций.....	251
5.4. Модели задач и методов решения однородных задач.....	255
5.5. Элементы функциональных гибридных интеллектуальных систем.....	264
5.6. Крупнозернистые функциональные гибридные интеллектуальные системы.....	270
5.7. Мелкозернистые функциональные гибридные интеллектуальные системы.....	273
5.8. Модели поведения функциональных гибридных интеллектуальных систем.....	275
6. МЕТОДОЛОГИЯ ФУНКЦИОНАЛЬНЫХ ГИБРИДНЫХ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ.....	281
6.1. Понятие проблемно-структурной методологии функциональных гибридных интеллектуальных систем.....	281
6.2. Жизненный цикл методологии.....	283
6.3. Стратификация решения сложных задач в СППР.....	289
6.4. Классификация отношений интеграции знаний.....	294
6.5. Методы выяснения причин возникновения сложных проблем управления и обработки информации.....	303

6.6. Методы редукции сложности задач.....	306
6.7. Выбор методов решения однородных задач.....	312
6.8. Метод проверки неоднородности сложных задач.....	318
6.9. Метод и алгоритмы комбинирования знаний экспертов в функциональных гибридных интеллектуальных системах.....	320
6.10. Свойства проблемно-структурной методологии функциональных гибридных интеллектуальных систем.....	327
6.11. Принципы разработчика функциональных гибридных интеллектуальных систем.....	328
7. ТЕХНОЛОГИЯ РЕШЕНИЯ СЛОЖНЫХ ЗАДАЧ МЕТОДАМИ ФУНКЦИОНАЛЬНЫХ ГИБРИДНЫХ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ.....	333
7.1. Понятие проблемно-структурной технологии функциональных гибридных интеллектуальных систем.....	333
7.2. Информационная среда технологии.....	335
7.3. Операции технологии.....	341
7.4. Последовательность действий разработчика и пользователя функциональных гибридных интеллектуальных систем.....	352
7.5. Инструментальная среда технологии.....	354
7.6. Пример решения сложной задачи планирования урожаев и агротехнических мероприятий в био-производственной системе	361
ЗАКЛЮЧЕНИЕ.....	371
ЛИТЕРАТУРА.....	373

СПИСОК СОКРАЩЕНИЙ

АМ — аналитический метод
БЗ — база знаний
БПС — био-производственная система
БСЭ — большая советская энциклопедия
ГиИС — гибридная интеллектуальная система
ГА — генетический алгоритм
ГМП — гетерогенное модельное поле
ГМ-список — список моделей гетерогенного модельного поля
ЕЯ — естественный язык
ИИ — искусственный интеллект
ИНЖ — инженерная страта
ИНС — искусственная нейронная сеть
ИМ-список — список интерфейсов автономных моделей
ИСМ — имитационное статистическое моделирование
ИЭС — интегрированная экспертная система
КСППР — компьютерная система поддержки принятия решений
ЛПР — лицо, принимающее решения
МИ-список — список пар «модель—интервалы»
НС — нечеткая система
П-список — список входов и выходов моделей из ГМП
ПС — проблемно—структурная
ПС-список — список существенных параметров
ПЭК — планово—экономическая страта
РВ — реальное время
РР-список — список результатов решения задачи
СПП — система поддержки переговоров
СПГР — система поддержки групповых решений
СМО — система массового обслуживания
СН-таблица — таблица связей неоднородной задачи
СППР — система поддержки принятия решений
Т-таблица — таблица гибридных стратегий
ТИ-список — список технологических компонентов
ТПР — теория принятия решений
ФГиИС — функциональная гибридная интеллектуальная система
Э-таблица — таблица опроса экспертов
ЭС — экспертная система
ЯПД — язык профессиональной деятельности

ВВЕДЕНИЕ

Усложняющиеся задачи обработки информации и управления в пространственно-распределенных системах промышленности и энергетики, био-производственных системах сельского хозяйства, в космосе, при противодействии или взаимодействии с партнерами или противниками неизбежно приводят к ошибкам при принятии решений и увеличению сложности и трудоемкости средств автоматизированного решения проблем. Существующие информационные системы всё менее и менее пригодны для этих целей. Человеческий мозг справляется с задачами не потому, что имеет много нейронов, а потому, что эти нейроны взаимодействуют творчески, что наделяет живые существа эмоциями, воображением и способностями к самообучению. Люди решают сложные задачи коллективно. Так можно не только снизить сложность проблемы, разделить ее на части, поручить выработку частных мнений наиболее квалифицированным специалистам, но и, организовав их взаимодействие путем обсуждения, получить общее решение-комбинацию. Коллективы людей научились вырабатывать и применять новые методы к решению сложных задач, в то время как компьютеры по-прежнему используются как инструмент реализации уже известных и ограниченных по своим возможностям инструментариев.

В представляемой книге три понятия — «сложная задача», «система поддержки принятия решений» и «гибридная интеллектуальная система» — рассмотрены вместе. Идея такого изложения материала возникла девять лет назад. Но реализовать ее до настоящего издания в полном объеме не удавалось. Теперь это оказалось возможным и актуальным. И вот почему.

Сложные задачи проявляют свои свойства в полной мере, когда специалисты берутся за решение не «игровых», а практических задач, т.е. таких задач, какие они есть на самом деле, в реальном мире. Сложные, практические задачи оказались «капризнее» к методам и технологиям, чем элегантные научные задачи-образцы.

Системы поддержки принятия решений от заседания правительства до производственной планерки-пятиминутки в большинстве случаев успешно справляются со сложностью задач и вырабатывают качественные решения. Возможно, это происходит благодаря знаниям и опыту экспертов, решающих отдельные части сложной задачи; знаниям и опыту лица, принимающего решения, умеющего объединить частные решения в одно общее, взаимодействуя с экспертами. Может быть, стоит «научить» компьютер моделировать работу СППР над сложной, практической задачей?

Гибридные интеллектуальные системы — инструмент синергетического искусственного интеллекта, предназначенный для моделирования эффектов взаимодействия, самоорганизации, адаптации, наблюдаемых в системах, где тесно переплетаются природа, человек и техника. Возможно, модели и методы ГиИС окажутся релевантными сложности задач, решаемых в системах поддержки принятия решений.

Совместное рассмотрение трех сложных понятий не открывает путь к инженерной практике автоматизированных систем обработки информации и управления. В этой связи предлагается методологическое и технологическое знание для разработки одного из классов гибридов — функциональных гибридных интеллектуальных систем. Благодаря настоящей публикации эти знания становятся открытыми не только для практического использования, но и для анализа, критики и выработки новых, более качественных знаний, переводящих решение сложных задач из области научных дискуссий в практику.

Методология и технология разработки функциональных гибридных интеллектуальных систем — это результат работы над проблемами синергетического искусственного интеллекта, проведенной в ФГОУ ВПО «Калининградский государственный технический университет», Санкт-Петербургском государственном политехническом университете и Калининградском филиале Института проблем информатики Российской Академии наук.

Настоящая книга содержит результаты, полученные в разное время, в том числе и при выполнении НИР «Методология и технология разработки гибридных интеллектуальных систем для СППР в сложных, практических задачах» в Калининградском филиале Института проблем информатики РАН в период 2004 — 2006 гг.

Книга имеет следующую структуру.

Глава 1 посвящена объекту-оригиналу методологии функциональных ГиИС — сложным задачам из практики систем управления. Сложные задачи представляются неоднородными системами. Исследуется происхождение гетерогенности, рассматриваются свойства задач. Дается обзор подходов к решению задач в автоматизированном режиме.

Глава 2 рассматривает системы поддержки принятия решений как инструмент решения сложных задач. Исследуются состав, структура и особенности работы экспертов над решением подзадач — частей сложной задачи, а также ЛПР в СППР. Основное внимание смещается на явления и процессы выработки коллективных решений в СППР. Приводится обзор классификаций и рассматривается состояние компьютерных систем поддержки принятия решений.

Глава 3 вводит читателя в мир методов моделирования рассуждений экспертов в КСППР. Методы сгруппированы по их ориентации на теоретические знания экспертов (аналитические и статистические представления), профессиональные знания (экспертные, нечеткие системы, искусственные нейронные сети, эволюционные представления) и опыт. Рассмотрено моделирование образного, невербального стиля рассуждений экспертов. При изложении методов рассуждений экспертов в главе 3, в основном, сохранены общепринятые обозначения.

Глава 4 посвящена моделированию деятельности ЛПР в системах поддержки принятия решений. Основное внимание уделено имитации способности ЛПР интегрировать знания экспертов в едином, окончательном решении сложной задачи. Дается обзор моделей интеграции знаний, и рассматриваются понятия систем: интегрированных экспертных, многоагентных, гибридных интеллектуальных и функциональных ГиИС.

Глава 5 включает результаты моделирования функциональных ГиИС. Предлагается категориальный базис для описания состава, структуры и поведения функциональных ГиИС. Рассмотрены теория схем ролевых концептуальных моделей и построение в ней схем описания объектов-оригиналов, объектов-прототипов и объектов-результатов методологии функциональных ГиИС. В этой главе даны обозначения, используемые в книге для моделирования процессов и явлений в СППР. Для более глубокого знакомства с предлагаемыми моделями необходимо прочитать разд. 5.1 и 5.2.

Глава 6 дает представление о методологическом знании, направляющем и сопровождающем решение сложных задач методами функциональных ГиИС. Здесь даны ответы на вопросы: «Что делать?» и «Как делать?», если предстоит проект решения сложной задачи в СППР. Рассмотрен «жизненный цикл» ПС-методологии, а также методы выяснения причин возникновения сложных проблем, решения сложных задач в СППР, системного анализа задач, выбора инструментария для решения подзадач сложной задачи, а также методы и алгоритмы синтеза функциональных ГиИС. Для практического руководства предложены принципы разработчика функциональных ГиИС.

Глава 7 детализирует этапы действий разработчика функциональных ГиИС. Рассмотрена технология решения сложных задач методами функциональных ГиИС. Даны операции, их последовательность, информационный язык, система управления деятельностью разработчика. Рассмотрен пример решения сложной задачи.

1. СЛОЖНЫЕ ЗАДАЧИ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ

1.1. Условия увеличения сложности задач в современном обществе

Развитие общества непрерывно усложняет совершенствование человеческой деятельности в сферах производства, управления, обучения, науки, творчества и др. Изменения, происходящие в управлении, — это не разовые преобразования с целью достижения «наилучшего» состояния управления, а непрерывающийся во времени процесс [1]. Эти изменения видны из табл. 1.1, где дан анализ тенденций мирового рынка в XX в. по нескольким характеристикам: привычности событий, с которыми сталкиваются управленцы; темпам изменений; предсказуемости будущего и методам управления [2].

Таблица 1.1 — Основные тенденции мирового рынка в XX в.

Характеристики	Годы				
	1900	1930	1950	1970	1990
Привычность событий	Привычные	В пределах экстраполяции	Неожиданные, но имеющие аналоги в прошлом	Неожиданные и совершенно новые	
Темп изменения	Медленнее, чем реакция фирмы	Сравним с реакцией фирмы		Быстрее, чем реакция фирмы	
Предсказуемость будущего	По аналогии с прошлым	Путем экстраполяции	Предсказуемые новые проблемы и возможности	Частичная предсказуемость по слабым сигналам	
Методы управления	На основе контроля	На основе экстраполяции	На основе предвидения изменений	На основе гибких экстренных решений	

Изменения в управлении отображаются и в терминологии. В середине XIX в. И.А. Вышнеградский и Дж. Максвелл впервые применили к автоматике методы математического анализа, заложив основы теории автоматического управления (контроля). До 40-х годов XX в. теория не отличалась разнообразием подходов. В этот период понятие «оптимального управления» стало применяться к техническим и социально-экономическим системам.

В начале 70-х годов советские ученые Ю.И. Клыков и Д.А. Поспелов ввели понятие «ситуационного управления» [3 — 5], способствовавшее разработке и применению в практике управления логико-лингвистических моделей.

В 1976 г. впервые в работах Вербоса появился термин «нейроуправление» [6]. Это означало начало эры использования математической модели нейрона Мак-Каллоха и Питса [7] и идей обучения Хебба [8] в теории управления. Системы управления разрабатывались с использованием не формальных математических, а качественных, эвристических моделей.

В 1975 г. советский психолог В.Ф. Венда [9], чтобы подчеркнуть непреходящее эволюционное значение взаимодействия естественных интеллектов в природе, обществе и технике, ввел понятие «гибридный интеллект» как контрпонятие искусственному интеллекту. В начале 80-х годов была выдвинута идея гибридного интеллектуального управления [10] на многомодельных системах.

Академик РАН И.В. Прангишвили, оценивая в 1999 г. состояние теории управления, указал на необходимость решения задач рационального управления слабо формализованными техническими, социальными, экономическими и другими объектами. По своей природе они размыты, многофункциональны, обладают сложной иерархической структурой, а их строгое описание практически невозможно [11]. Развитие теории управления в последние годы породило новые научные направления: 1) физическую теорию управления, включающую учет ограничений, накладываемых физическими законами на процессы управления; 2) микромеханику на нано-уровне; 3) геометрическую теорию управления с математическим аппаратом теории расслоения; 4) нейроуправление; 5) применение системного подхода и системного мышления при принятии управленческих решений, широкое использование генетических алгоритмов. Все эти направления немислимы без применения средств вычислительной техники.

Изменения в терминах — индикатор изменений в управлении, обусловленных динамическими процессами в производительных силах общества. В этой связи выделяют управление: традиционное, промышленной и постиндустриальной (информационной) стадии [12].

Традиционное управление, когда машинные технологии еще не были развиты, базировалось на физических и умственных способностях человека, его умениях и опыте.

Для управления на промышленной стадии характерно изменившееся положение человека в производстве. Техника и технология определяли место человека, от которого требовалось умело, рационально и эффективно использовать ресурсы, и выдвигали требования к организацион-

ным структурам. Важнейшие тенденции развития управления в промышленном (индустриальном) обществе отражены, в частности, в теории организаций Ф.В. Тейлора [13], базирующейся на четырех принципах:

- 1) жесткой организационно-управляющей иерархии, верхний уровень которой имеет всю полноту власти, в то время как исполнительские уровни не могут принимать решения самостоятельно; отношения по вертикали фиксированы; задачи передаются вниз, а результаты — вверх; руководитель распределяет задачи между исполнителями и контролирует их выполнение; нет перераспределения функций и спонтанно возникающих новых структур;
- 2) специализации с разбиением исходной задачи на более простые операции;
- 3) единоначалия, когда исполнитель имеет одного непосредственного руководителя;
- 4) исключительного вмешательства, когда рутинные задачи решаются исполнителями, а руководитель вмешивается в исключительных случаях.

Тейлоровская организация (табл. 1.2), как считает В.Б. Тарасов, оправдана в условиях стабильной внешней среды, массового производства, хорошей «разложимости» задачи на составные части — подзадачи и исполнения подзадачи без обмена информацией между экспертами.

Концепция «постиндустриального, посттейлоровского общества» (табл. 1.2) была выдвинута американским социологом и политологом Д. Беллом в 1965 г. [14], а модель «информационное общество» предложена в конце 60-х — начале 70-х годов XX столетия как вариант экономического развития Японии.

В качестве основных черт этого общества выделяют, прежде всего, переход от производства вещей к производству услуг в сферах здравоохранения, образования, исследований и управления. При этом перераспределяются занятия: растет численность интеллигенции, профессионалов и «технического класса».

Индустриальное общество требует рациональности и эффективного использования ресурсов, а центральное звено постиндустриального общества — теоретическое знание. Для превращения знаний в источник инноваций возникают наукоемкие отрасли: химическая промышленность, вычислительная техника, электроника, оптика и др. Соединение науки, техники и экономики выражается в феномене научных исследований и разработок, которые, должны играть более значимую роль в обществе, ориентированном на будущее. Существенное место в борьбе с «организованной сложностью» займет возникшая интеллектуальная

технология: системный анализ и теория принятия управленческих решений. Роль «мастера» в интеллектуальной технологии играет теория принятия решений, а роль «инструмента» выполняют компьютеры, сокращающие время на выполнение большого объема вычислений и рассуждений.

Таблица 1.2 — Сравнительный анализ тейлоровских и посттейлоровских организаций В.Б. Тарасова

Характеристика	Тейлоровские организации	Посттейлоровские организации
Адаптация	Телогенез (адаптация к заданному состоянию)	Арогенез (расширение набора сред существования)
Взаимодействия	Вертикальные (субординация)	Горизонтальные (координация)
Вид организации	Монолитная, замкнутая	Открытая, распределенная, сетевая
Внешняя среда	Статическая, стабильная	Динамическая, быстроменяющаяся
Задачи	Простые (специализация)	Реинтегрированные (деспециализация)
Организационная единица	Однородное функциональное подразделение	Автономная междисциплинарная рабочая группа
Развитие	Жесткое планирование	Самореорганизация
Связи	Постоянные	Гибкие, переменные
Структура	Иерархия типа «дерево»	Гетерархия (неоднородная сеть)
Управление	Централизованное	Децентрализованное

Модель «информационное общество» исследована, например, в [15—17]. Авторы выделяют следующие условия решения задач управления и обработки информации: 1) источник роста производительности — знания, распространяемые через обработку информации; 2) научная обработка данных и поддержка знаний позволят принимать более продуманные и обоснованные решения; 3) коллективный разум — опора человечества; 4) система, использующая новые информационные технологии, имеет сетевую логику, необходимую для структурирования неструктурированного, сохранения гибкости; 5) информационно-технологическая парадигма, основанная на гибкости всех систем, процессов и отношений; 6) растущая конвергенция конкретных технологий в высокоинтегрированной системе; 7) по мере все большей специализации, детализации концепций современной науки, наиболее важными результатами ее связей с технологией становятся интеграция различных областей в единую теоретическую систему и синергия в открытиях и

разработках новых продуктов и теорий; 8) телекоммуникации и обработка информации сливаются в единую модель, получившую название «компьюникация».

Таковы модели анализа реального положения дел с развитием производительных сил конкретного общества. Такова реальность.

Столкновение с реальностью методов и программ искусственного интеллекта С. Рассел и П. Норвиг [18] относят к периоду с 1966 по 1973 г. Решение задач системами управления выполнялось в условиях изменяющегося мирового рынка (табл. 1.1) и их эволюции от тейлоровских к посттейлоровским организациям (табл. 1.2). При этом управленцы, решая задачи реальные, а не искусственно упрощенные наукой, все чаще сталкивались с неожиданными и новыми событиями во внешней среде, с заметным отставанием реакции фирм от темпов изменения рынка, со сложностями планирования из-за частичной предсказуемости будущего по слабым, нечетким сигналам. Реакция системы управления на события, действия и изменения во внешней среде могла быть найдена на основе гибких, экстренных, коллективных решений.

В таких условиях перестали работать методы искусственного интеллекта, использовавшие принцип проверки различных комбинаций возможных шагов, выполняемой для поиска решения [18]. Такая стратегия применима для искусственно упрощенных микромиров с ограниченными ресурсами, действиями, событиями и ситуациями, когда последовательность шагов решения задачи была короткая. Однако в условиях «комбинаторного взрыва» — стремительного увеличения сложности задач — оказалось, что если программа и может найти решение в принципе, то этот факт еще не означает, что программа содержит все механизмы, позволяющие найти данное решение на практике [18].

Искусственный интеллект «отреагировал» на увеличение сложности задач, прежде всего, появлением экспертных систем Э. Фейгенбаума, Б. Бьюкенена, Дж. Леденберга (1969 г.) для решения задач рассуждениями на профессиональных знаниях. Были разработаны нечеткие множества Л. Заде (1965 г.) для решения задач в условиях лингвистической неопределенности. Анонсированы репродукционные планы Дж. Холланда (1975 г.) для поиска решений в областях экстремумов функций. Была развита теория искусственных нейросетей Т. Кохоненом, С. Гроссбергом, Дж. Андерсеном (70 — 80-е годы XX в.). Однако успехи с преодолением барьера «комбинаторного взрыва» оказались скромнее, чем можно было ожидать.

Реальность задач управления и обработки информации в современном обществе оказалась, как представляется авторам, еще «сложнее», чем думали в середине 70 — начале 80-х годов XX столетия. А парадиг-

мы искусственного интеллекта еще «слабее», не релевантнее способностям людей решать задачи в своей профессиональной деятельности.

1.2. Классификация задач. Понятие неоднородности задач

Без проблем, больших или малых, не функционирует ни одна система управления, они есть в любой сфере деятельности коллектива.

Проблема (греч. *pro'blēma* — задача): 1) в широком смысле сложный теоретический или практический вопрос, требующий изучения, разрешения [19]; 2) сложный вопрос, задача, требующие разрешения, исследования [20]. Таким образом, в русском языке понятия «проблема» и «задача» по содержанию могут использоваться как синонимы. Такого же мнения придерживается О.И. Ларичев в работе [21] и Ж.-Л. Лорьер (см. табл. 1.6).

Однако в повседневной жизни эти термины применяются людьми в различных ситуациях. Ситуация использования понятия «проблема» характеризуется обнаружением в системе управления признаков, событий и отношений, указывающих на отклонение ее состояния от обычного, желаемого, целевого. В подобных случаях говорят: «У нас проблема» или «Существует проблема».

Ситуация, когда чаще используется понятие «задача», возникает несколько позже, уже после установления факта наличия проблемы, и характеризуется не только осознанием существования проблемы, но и желанием ее преодолеть. В подобных случаях говорят: «Перед нами стоит задача» или «Необходимо решить задачу».

Таким образом, термин «проблема» по своему содержанию ближе к категориям обусловленности, причины, а термин «задача» — к категориям результативности, следствия, действия.

В указанном смысле понятие проблемы исследовано, например, в работе М.В. Самсоновой, В.В. Ефимова [22], и представлено на рис. 1.1 трехслойной иерархией относительно шкалы «известное — неизвестное». Эту иерархию можно сравнить с айсбергом, доставлявшим и по-прежнему, несмотря на технический прогресс, доставляющим человечеству беды и излишние хлопоты в океане. Здесь уместно образное выражение писателя-фантаста П. Андерсона, что проблема, сколь бы сложной она не была, становится еще сложнее, если на нее правильно посмотреть [23].

Чаще всего возникают управленческие и организационные проблемы. Но не исключены проблемы и в конструкторско-технологической деятельности служб предприятий.

В основе любого решения стоит человек, а человеку свойственно ошибаться в силу различных причин. И следствием такой ошибки мо-

жет явиться проблема. Чаще всего реагируют на возникновение проблемы только тогда, когда она становится очевидной и нет возможности отложить ее решение.

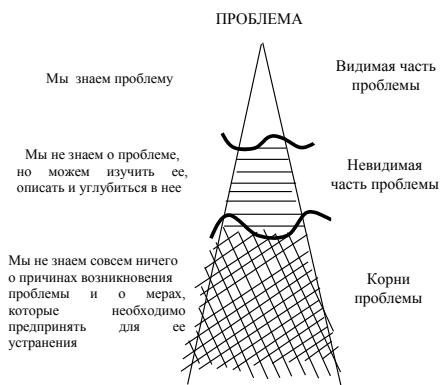


Рис. 1.1 – Трехслойная иерархия проблемы

При этом следует иметь в виду, что, по мнению Г. Саймона (H. Simon), «возможность человеческого ума формулировать и решать сложные проблемы очень мала по сравнению с размером проблем, решение которых необходимо для объективно рационального поведения в реальном мире или даже для разумного приближения к такой объективной реальности» (Цит. по кн. [21, с. 148]). Однако здесь мы

уже перешли к исследованию содержания понятия «задача». Решению задач уделяется основное внимание в настоящей книге.

Раскрывая природу любой задачи как субъективной сущности имеет смысл начать с рассмотрения условий возникновения задач. Принято считать [24], что задача есть тогда, когда: 1) существует субъект, перед которым возникает задача; 2) в распоряжении субъекта имеется минимум две альтернативы, т.е. есть выбор; 3) существует цель, которую он стремится достичь; 4) существуют оценки, по которым можно сравнить альтернативы и судить о достижении цели. Однако для ЛПР задача становится предметом решения только тогда, когда оно не знает, какая альтернатива наилучшая, и именно это ему и требуется определить.

Такое понимание термина «задача» введено специалистами по исследованию операций Р. Акофом и М. Сасиени в конце 60-х — начале 70-х годов, в период бурного расцвета идей индустриального общества, становления методов оптимизации в кибернетике. Похожая трактовка, правда, уже математических задач, предложена математиком и педагогом Дж. Пойя и опубликована им в начале 60-х годов [25].

Определения Р. Акофа, М. Сасиени и Дж. Пойя дошли до наших дней без существенных изменений. Это первый вывод, который можно сделать, если рассмотреть и проанализировать табл. 1.3 — 1.6 (со списком цитируемой литературы можно познакомиться в [1]), в которые сведены типичные определения задач, начиная с 1962 г. и до настоящего времени. Они систематизированы в четыре группы научных дисциплин: 1) лингвистику (табл. 1.3); 2) математику, теорию принятия реше-

ний, исследование операций, теорию управления (табл. 1.4); 3) теорию систем и системный анализ (табл. 1.5); 4) информатику, искусственный интеллект (табл. 1.6), взявших на себя «ответственность» за моделирование решения задач человеком.

Таблица 1.3 — Трактовка понятия «задача» в лингвистике

Авторы	Трактовка понятия «задача»
Ожегов, 1972	<i>Задача</i> — это: 1) то, что требует исполнения, размышления; 2) упражнение, которое выполняется посредством умозаключения, вычисления и т.п.
БСЭ, 1972	<i>Задача</i> — это: 1) поставленная цель, которую стремятся достигнуть; 2) полученное задание; 3) вопрос, требующий решения на основании определенных знаний и размышлений, проблема; 4) один из методов обучения и проверки знаний

Приведенные определения сформировались почти за 40-летний период развития этих научных дисциплин. Второй вывод можно сделать, обратившись, например, к определению Э.В. Попова и других специалистов по экспертным системам.

Это мнение особенно ценно, потому что именно экспертные системы — один из первых классов систем, ориентированных на решение не искусственно упрощенных задач исследования операций, теории принятия решений и управления, а задач, возникающих на практике. Заметим, что при переходе от искусственных к практическим задачам, как объектам анализа и решения в мире методов моделирования (рис. 7.3), по существу, изменилось определение задачи и расширился спектр свойств, которые целесообразно учесть, специфицируя проблему. Однако самое главное — это то, что в определение введены подзадачи, выделенные внутри исходной проблемы. Таким образом, сделан очевидный шаг к тому, что практика заставляет модельера отказаться от рассмотрения задачи как неделимой на составные части, единой сущности.

Практически все ученые, так или иначе определявшие понятие «задача», систематизировали проблемы в классификации, которые затем выполняли функцию инструмента сокращения многообразия в мире задач, выделения типовых классов задач, что оправдывало появление, существование и эволюцию методов решения задачи в мире моделирования.

При этом следует иметь в виду, что речь идет не о методе решения задачи дроблением на составные части, о котором говорил еще Декарт, или редукции методом Данцига-Вулфа [26], количественно сформулированных сложных задач линейного программирования, а об априорной гипотезе о том, что задача, возникающая в практике управленческой деятельности, должна рассматриваться во взаимосвязи ее частей, т.е.

как система. К системе должны применяться существующие или выработанные заново принципы системного анализа.

Таблица 1.4 — Трактовка понятия «задача» в математике, теории принятия решений, исследовании операций и теории управления

Авторы	Трактовка понятия «задача»
Пойя, 1970	В любой задаче содержится неизвестное — если все известно, то нечего искать, нечего делать. В задаче что-то должно быть известно или дано (данные) — если ничего не дано, то нет никакой возможности узнать требуемый объект. Наконец, задача должна содержать условие, конкретизирующее связь между неизвестными и данными
Акоф, Сасиени, 1970	Перед субъектом возникает задача, если ему нужно достичь какой-либо цели и существуют различные пути ее достижения, каждый из которых характеризуется различной эффективностью, причем возможны сомнения относительно того, какую стратегию ему выбрать
Бенерджи, 1972	Задача сводится к отысканию последовательности таких элементарных управлений, что независимо от сопряженных с каждым из них элементарных возмущений эта последовательность, в конце концов, приводит к выигрышной ситуации
Растргин, 1980	Для постановки задачи необходимо иметь модель объекта $F(X, U)$, информацию о состоянии среды X и цель управления Z^*
Борисов и др., 1986	Задача принятия решения — это семерка $\langle t, X, R, A, F, G, D \rangle$, где t — постановка задачи; X — множество допустимых альтернатив; R — множество критериев оценки степени достижения поставленных целей; A — множество шкал измерения по критериям; F — отображение множества допустимых альтернатив на множество критериальных оценок; G — система предпочтений решающего элемента
Науман, 1987	Задача принятия решения определяется ситуацией: кто должен или обязан (или хочет) решать? Где, т.е. на (в) каком месте, в каком окружении (среде), при каких обстоятельствах и граничных условиях предстоит принимать решение? Когда (до какого срока, иногда — как часто) требуется принимать решения? Как (каким образом или в какой форме, иногда — чем) должно быть выражено решение? Что обуславливает решение? В чем его замысел, цель? Для чего оно служит? Зачем его нужно принимать?
Голубков, 2006	Задача принятия решений для индивидуального ЛПР: $\langle C, T, P \mid C_0, \Pi, \Pi, O, A, K, f, A^* \rangle$, где слева от \mid — известное, а справа — неизвестное; C — исходная проблемная ситуация; T — время для принятия решения; P — ресурсы для принятия решения; C_0 — доопределенная проблемная ситуация; $\Pi = (\Pi_1, \dots, \Pi_n)$ — множество гипотез о развитии ситуации в будущем; $\Pi = (\Pi_1, \dots, \Pi_k)$ — множество целей решения; $O = (O_1, \dots, O_l)$ — множество ограничений; $A = (A_1, \dots, A_m)$ — множество альтернатив; $K = (K_1, \dots, K_p)$ — множество критериев выбора; f — функция предпочтения ЛПР, включающая объективные критерии из K и субъективные предпочтения ЛПР; A^* — оптимальное решение. Задача принятия решений для группового (коллективного) ЛПР: $\langle C, T, P \mid C_0, \Pi, O, A, K, F(f), L, A^* \rangle$, где $C, T, P, C_0, \Pi, O, A, K, A^*$ — см. выше; $F(f)$ — функция группового предпочтения, зависящая от вектора индивидуальных предпочтений членов группы $f = (f_1, \dots, f_d)$, здесь d — количество членов в группе; L — принцип согласования личных предпочтений в групповом мнении

Тем самым ограничивался перечень задач, которые могли быть решены с использованием того или иного метода. Как подчеркивал Д. Пойя, при решении задач может оказаться полезной их классификация.

установление различия между задачами по их типам. Хорошая классификация предполагает такое разбиение задач, что тип задачи предполагает метод ее решения [25].

Таблица 1.5 — Трактовка понятия «задача» в теории систем и системном анализе

Авторы	Трактовка понятия «задача»
Поспелов, Ириков, 1976	<i>Задача</i> — желаемый результат деятельности, достижимый на намеченный (заданный) интервал времени $[t_0, t_1]$ и характеризующийся набором количественных данных или параметров этого результата, т.е. цель становится задачей, если указан срок ее достижения и конкретизированы количественные характеристики желаемого результата
Волкова, Денисов, 1999	<i>Задача</i> — это цель, возможные средства ее достижения, критерии (несколько критериев), отражающие требования к достижению цели. Для решения задачи нужно определить выражения, связывающие цель со средствами
Нечаев, 1990	Концептуальная модель <i>задачи</i> может быть представлена тройкой $\pi = \langle G, D, C \rangle$, где G — конечная цель (результат) решения задачи; D — исходные данные, удовлетворяющие требованиям необходимости и достаточности (полноты) для получения результата; C — условия (известные или предполагаемые), конкретизирующие отношения между D и G , отражающие, как D преобразуется в G ; $C = \langle m, a, p \rangle$, где m — метод решения (указания и описания); a — алгоритм и p — программа решения задачи

На рис. 1.2 показана онтология (греч. *ontos* — сущее и *logos* — понятие, учение мира задач (рис. 7.3), релевантная современному уровню понимания задач и зафиксированная в научных дисциплинах из табл. 1.3 — 1.6).

Онтология построена следующим образом. Центральное место в ней занимает понятие «задача», которое уточняется через свойства, записанные внутри пятиугольников. Эти же свойства — одновременно основания классификаций задач. В прямоугольниках даны названия классов, указаны авторы (со списком цитируемой литературы можно ознакомиться в [1]) и в левом верхнем углу — номер классификации.

Подчеркнем, что данная онтология охватывает далеко не все, что сочетается со словом задача. Так, например, не учтены типовые задачи исследования операций: назначения, размещения, распределения и т.п., класс тестовых задач и другие, не влияющие на дальнейшие рассуждения. Некоторые пятиугольники содержат более одного основания, и тогда, в порядке «сверху вниз», им соответствуют и разделенные черточками классификации.

Классификации **1, 6, 9 и 10** [1] традиционны для науки. Рассмотрим классификации **2 — 4 и 7**.

Классификация по знаниям, необходимым для решения задач
(2). Различают два типа задач: хорошо- и плохоопределенные задачи. Задача называется хорошоопределенной, если решающий ее эксперт располагает каким-то способом узнать или имеет хотя бы принципиальную возможность узнать, когда он завершил решение задачи, т.е. существует некий алгоритмический метод ее решения.

Таблица 1.6 — Трактовка понятия «задача» в искусственном интеллекте

Авторы	Трактовка понятия «задача»
Нильсон, 1973	Любая <i>задача</i> поиска в пространстве состояний может быть представлена как совокупность трех составляющих $\langle S, F, G \rangle$, где S — множество начальных состояний; F — множество операторов, отображающих описание состояний в описание состояний; G — множество целевых состояний
Ефимов, 1982	<i>Задача:</i> заданы исходная и конечная ситуации. Необходимо построить из конкретных операций программу, которая, будучи примененной к исходной ситуации, позволила бы достичь конечной ситуации
Гладун, 1987	Текущая ситуация и цель в совокупности образуют <i>задачу</i> , которая управляет деятельностью
Поспелов, 1986	Для постановки <i>задачи</i> управления объектом ЛПР должно знать описание объекта (его структуру и функционирование), цель существования объекта и критерий управления
Лорьер, 1991	Смысл слова «задача» как синонима слову «проблема» происходит от значения греческого слова «баллейн» — бросать. Задача — это объект, брошенный вперед
Попов и др., 1986	Для идентификации неформализованной (плохоструктурированной) <i>задачи</i> необходимо указать: общую характеристику задачи; подзадачи, выделенные внутри данной задачи; ключевые понятия, характеристики и отношения; входные и выходные данные; предположительный вид решения; знания релевантных задач; примеры (тесты) решения задачи
Башлыков, Еремеев, 1994	<i>Задача</i> поиска удовлетворительного решения определяется набором $\langle P, G, X, \tau \rangle$ и интерпретируется как поиск такого $\tilde{x} \in \tilde{X}$, что выполняется условие допустимости $(\forall \omega \in \Omega)(g(\tilde{x}, \omega)R\tau(\omega))$, где X — множество исходных данных (решений, входов, альтернатив, водных воздействий и т.п.); $\tilde{X} \subseteq X$ — множество допустимых решений; $\tilde{x} \in \tilde{X}$ — удовлетворительное решение; Y — множество результатов (выходов, следствий, выходных параметров); Ω — множество неопределенностей, элементы которого представляют факторы, влияющие на получаемый результат; $P: X \times \Omega \rightarrow Y$ — выходная функция; $G: X \times \Omega \times Y \rightarrow V$ — оценочная функция (функция качества); V — множество платежей; $R \subseteq V \times V$ — отношение (например, \leq); $g(x, \omega) = G(x, P(x, \omega))$ — целевая функция, $\tau(\omega)$ — функция допустимости.

Большинство же задач в повседневной практике управления плохо определены. Их еще иногда называют интеллектуальными. Решая такие задачи, ЛПР или эксперт выбирает последовательность действий, не

будучи уверенным, что они окажутся эффективными в данных обстоятельствах.

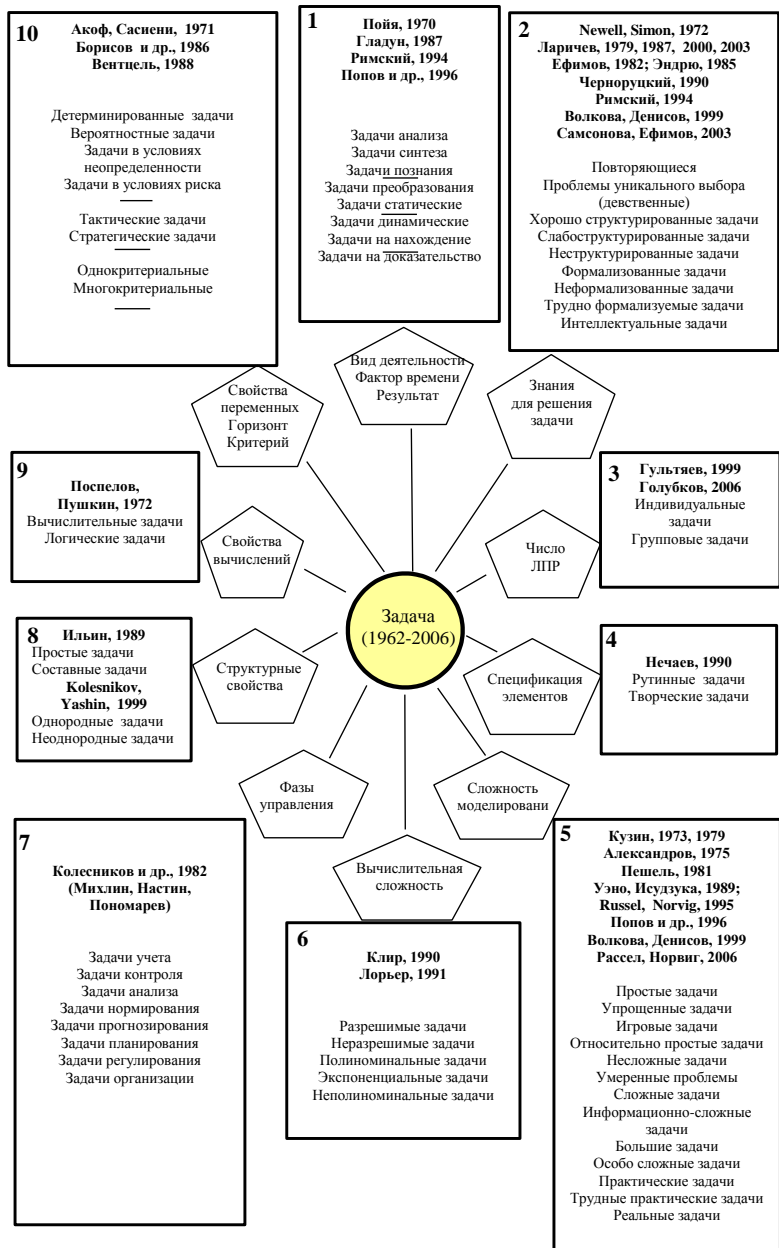


Рис. 1.2 — Онтология мира задач

Задачи разделяются также на формализованные (хорошо структурированные (англ. well-structured)) и неформализованные (трудноформализуемые, плохо структурированные (англ. ill-structured)). Неформализованные задачи обладают одним или несколькими из следующих свойств: задача не может быть решена в численной форме; цели не могут быть выражены в терминах точно определенной целевой функции; не существует алгоритмического решения задачи; алгоритм есть, но его нельзя использовать из-за ограниченности времени, памяти компьютера и других ресурсов.

Можно сказать, что типичные задачи исследования операций хорошо структурированы, а многокритериальные задачи слабоструктурированы. Для последних отсутствие объективной информации неустранимо на момент принятия решений. Более того, существуют проблемы, в которых можно определить перечень основных параметров, но количественные, аналитические связи между ними сформулировать нельзя. В таких случаях структура, понимаемая как совокупность связей между параметрами, не определена, и проблема считается неструктурированной.

Среди множества задач в системах управления выделяются задачи уникального выбора. Для них характерны: уникальность и неповторимость ситуации выбора; сложный для оценки характер рассматриваемых проблем; недостаточная определенность последствий принимаемых решений; наличие совокупности разнородных факторов, которую следует принимать во внимание; наличие лица или группы, ответственных за принятие решений.

Классификация задач по числу ЛПР (3). По признаку числа ЛПР различают задачи индивидуального и группового принятия решений. При групповом выборе решений определяющую роль играет проблема согласования индивидуальных предпочтений членов группы.

Классификация задач по спецификации элементов (4). Следует различать рутинные и творческие задачи. Первые предполагают, что цели G , условия C и данные D известны (см. определение В.В. Нечаева в табл. 1.5). Творческая задача имеет место, когда априори неизвестны программа p , или алгоритм a , или метод m . Таким образом, прежде чем решать такую задачу, необходимо определить (подобрать, синтезировать, обосновать) метод m , алгоритм a и программу p .

Вопросы сложности моделирования и структурных свойств задач отражены в классификациях 5 и 8.

Классификация задач по сложности моделирования (5). Эта классификация предполагает существование некоторой шкалы, на которой задачи могут быть упорядочены с точки зрения результатов измерения их сложности. Однако, судя по публикациям, такая шкала на сегодня не сформировалась в двух отношениях: 1) не ясны ее левая и правая границы; 2) не установлена численная мера сложности задачи. Туманность предельных свойств задачи видна из разд. 5 онтологии, который, скорее, не классификация в общепринятом смысле этого слова, а терминологическое поле.

Попробуем такие предельные свойства сформулировать. Ближе к левой границе лежат понятия «простой» и «игровой» задачи (англ. *toy problem*). Простая задача характеризуется, как определяют П. Рассел и С. Норвиг, следующими значениями параметров: малым (не слишком большим) размером пространства поиска; точностью и полнотой данных; статичностью области; возможностью адекватного описания предметной области с помощью модели. На практике встречается мало приложений, удовлетворяющих перечисленным требованиям. По природе игровые задачи могут быть заданы краткими, сжатыми, четкими, точными, строгими описаниями [27]. К простым относят такие задачи, как «игра в пять», «задача восьми ферзей», «задача о ханойских башнях», «задача о миссионерах и людоедах» и др. «Игра — удобный объект для подобных исследований (машинного обучения — *прим. авторов*), ибо лишена многих сложностей, свойственных задачам, взятым из реальной жизни» [28, с. 73]. «Упрощенная задача предназначена для иллюстрации или проверки различных методов решения задач. Ей может быть дано краткое, точное описание. Это означает, что такая задача может легко использоваться разными исследователями для сравнения производительности алгоритмов» [18, с. 116].

Ближе к правой границе находятся понятия «сложной», «практической» задачи. В книге [29, с. 140] переход к сложным задачам определен так: «Сложные задачи характеризуются тем, что значение хотя бы одного из перечисленных параметров оказывается в них более сложным». В той же работе дано еще одно определение: сложные задачи — это задачи с нефиксированными предметной и проблемной областями. Поясним второе определение. Оно подчеркивает полиязыковый характер принятия решений и то, что состав, а значит, и структура сложной задачи — переменные величины. В книге [27] подчеркнуто, что практические задачи, в противоположность игровым, имеют тенденцию не иметь единственного общепринятого описания. Во втором издании своей книги эти же авторы пишут: «Реальной задачей называется такая задача, решение которой действительно требуется людям. Как правило, такие задачи не имеют единого приемлемого для всех описания» [18, с. 116].

Между левой и правой границами располагается широкий спектр терминов — классов задач с промежуточными значениями меры сложности, выраженной качественно, а не количественно.

Классификация по фазам управления (7). Исследования содержания управления, как специфического вида трудовой деятельности, позволили выделить восемь основных фаз (стадий): учета, контроля, анализа, нормирования, прогнозирования, регулирования, организации и планирования со специфическими свойствами решаемых на этих фазах задач. Задачи учета решаются, чтобы зафиксировать данные о состоянии объекта управления и выполнить их первичную обработку: классификацию, группировку и сортировку для выработки учетной информации. Задачи контроля решаются, чтобы обработать учетную информацию, выявить и отобрать существенные данные о состоянии объекта управления, сравнить их с нормативными или плановыми значениями. В результате решения вырабатывается контрольная информация об отклонениях параметров объекта управления. Задачи анализа предполагают регулярную обработку учетной информации для выявления аналитических, статистических, логических и других видов связей между параметрами состояний, характеризующими поведение объекта управления. В результате решения вырабатываются аналитические, статистические, логические и частично формализованные знания о поведении объекта управления. Задачи нормирования предполагают систематическую обработку накапливаемой учетной информации и знаний для определения допустимых значений существенных параметров. В результате формируется нормативная информация для нормального функционирования объекта управления. Задачи прогнозирования предполагают регулярную обработку учетной, нормативной информации и использование всех видов знаний, чтобы предвидеть возможное поведение объекта управления в будущем. Задачи регулирования предполагают оперативное принятие решений для ликвидации отклонений от нормального режима функционирования объекта управления. Решения принимаются на основе контрольной и нормативной информации и оформляются как управляющая информация. Задачи организации решаются для эффективного функционирования СППР (или организационной структуры системы управления), и их результаты содержатся в инструкциях, положениях и других документах. Задачи планирования решаются в ходе периодического принятия решений для обеспечения эффективности функционирования объекта управления на заданный плановый период на основании целей, стоящих перед СППР, и предполагают обработку учетной, нормативной, прогнозной информации с использованием всех видов накопленных знаний. В результате вырабатывается управляющая

информация — планы на заданный интервал времени: смену, сутки, месяц, год и более.

В целом, анализ онтологии на рис. 1.2 указывает на следующие обстоятельства: 1) несовершенство наших знаний о свойствах задач, которые могли бы послужить основой их классификации и сравнительного анализа; 2) необходимость перехода к определению задачи с позиций теории систем и системного анализа, т.е. как к объекту, имеющему системные свойства состава, структуры, упорядоченности, эмерджентности, взаимодействия с другими объектами, т.е. внешней средой; 3) актуальность построения концептуальных моделей задач, которые могли бы дать количественные оценки меры сложности задачи.

Классификация задач по структурным свойствам (8). Анализ задач по структурным свойствам дан в классификации В.Д. Ильина, примененной к автоматизированному составлению программ учета и планирования. Определенное им понятие «простая задача» не имеет внутренней структуры, т.е. не подлежит редукции. «Составная задача» понимается автором в узком смысле слова как задача, решаемая операционной системой ЭВМ и представляющая собой конструируемый из простых задач объект (память, выделяемая под задачу — объединение памяти, выделяемой под входы и выходы всех простых задач).

В классификации М. В. Самсоновой, В. В. Ефимова [22] обращено внимание на принципиальную разницу в решении повторяющихся и «девственных» задач. Эффективное решение первых требует тщательного анализа с целью установления причины проблемы, которая позволит исправить или отрегулировать ситуацию. Необходимо «сужать» и детализировать проблемное пространство для выяснения источника проблемы и реализации наиболее эффективного решения (рис. 1.3,а).

«Девственными» проблемами считаются те, которые являются новыми для системы управления, и зачастую они связаны с высокой степенью неопределенности по методам и результатам. Таким проблемам обычно предшествуют внутренние или внешние изменения. Для работы с «девственными» проблемами создаются группы (коллективы), так как эти проблемы требуют согласованности и

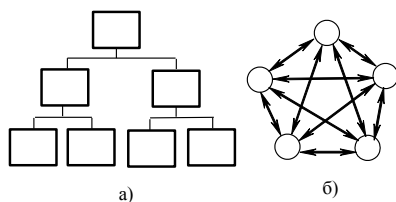


Рис. 1.3 — Структура решения проблем

видения новых перспектив (рис. 1.3,б).

В статьях [30 — 32] предложена иная классификация, рассматривающая задачи с позиций системного анализа с использованием хорошо

изученных в химии, термодинамике и физике понятий «гомогенной» и «гетерогенной» систем.

Гомогенная система (от греч. *homogenēs* — однородный) — физическая система, не содержащая частей, отличающихся по составу или свойствам и отделенных друг от друга поверхностями раздела [33].

Гетерогенная система (от греч. *heterogenēs* — разнородный) — неоднородная физико-химическая система, состоящая из различных по физическим свойствам или химическому составу частей (различных фаз). Одна фаза отделена от другой поверхностью раздела, на которой скачком изменяется одно или несколько свойств системы: состав, поле, структура и др. [34].

Применение этих двух терминов к понятию «задача» и приводит к двум моделям: 1) модели «однородная (гомогенная) задача» и 2) модели «неоднородная (гетерогенная)» задача. Такой подход сокращает многообразие степеней сложности задач в классификации **5** до двух, заменяя левую границу, т.е. «простые», «игровые», «упрощенные» задачи, на однородные, а правую границу, т.е. «сложные», «практические», «реальные», — на неоднородные.

Использование свойств однородности и неоднородности, широко наблюдаемых в деятельности человека, применительно к задачам уже достаточно давно присуще многим ученым. В частности, в книге [35] подчеркнуто, что особо сложные задачи в условиях неполной, недостаточной текущей информации возникают в экономике, исследованиях космоса и других областях человеческого разума — везде, где имеют дело с функционированием систем, зависящих от многих разнородных переменных. В работах [36, 37] отмечается, что среди множества проблем выделяются проблемы уникального выбора. Для них характерны: уникальность и неповторимость ситуации выбора; сложный для оценки характер рассматриваемых проблем; недостаточная определенность последствий принимаемых решений; наличие совокупности разнородных факторов, которую следует принимать во внимание; наличие лица или группы, ответственных за принятие решений. В статье [38] подчеркиваются характерные особенности ставшего актуальным в последнее время нового класса задач управления и поддержки принятия решений: обработка огромных объемов накапливаемой информации; принятие решений, в том числе, в режиме реального времени; манипулирование, как правило, разнородными по своему характеру данными и др. Еще в книге [39] отмечалось, что «знание» как цель исследования — слишком обширное и неоднородное понятие, и попытки решать основанные на знаниях задачи в общем виде — преждевременны. В статье [40] подчеркнуто, что знания разных типов объединены в цельную иерархиче-

скую, неоднородную структуру — сеть понятий. В [41] разрабатываются гетерогенные подсистемы в составе гибридной системы для решения практических задач. В [42, 43] подчеркнуто, что многие задачи в современном обществе неоднородны и распределены: а) в пространстве; б) в функциональном плане, поскольку ни один человек не может создать сложную систему в одиночку.

Неоднородность данных, переменных, которыми манипулирует человек, используя неоднородные знания и другие факторы, рассмотренные в разд. 1.3, естественно, приводят к свойствам однородности и неоднородности практических задач, которые следует рассматривать как еще один из НЕ-факторов, как обычных [44]: неопределенность, неточность, неполнота, ..., так и синергетических [13]: нелинейность, неустойчивость, неравновесность, незамкнутость,...

1.3. Источники неоднородности задач

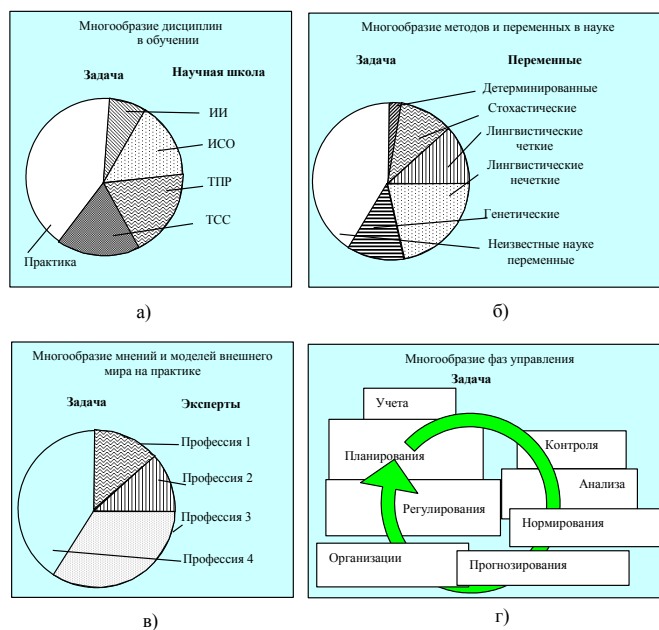
Главная причина неоднородности задач (или, иными словами, того, что сложные задачи релевантны модели «неоднородная задача») — существование диалектического противоречия между тем, что Вселенная обладает целостностью, а совокупность знаний о ней такой целостности не обладает ни в масштабах человека, ни в масштабах общества. Отсутствие целостности знаний в обществе — следствие эволюционного развития науки, которое проанализировано в книге [45], где на обширном фактическом материале показана закономерность смены ведущих парадигм. Под «парадигмой» понимается принятая модель (образец) построения знаний, т.е. внутренне непротиворечивая система понятий, концепций, теорий, практических исследований и их результатов. При этом свойством целостности обладает каждая отдельная парадигма, но не их совокупность. Например, в физике существуют две противоречивые парадигмы о природе света: волновая и корпускулярная.

Детализируя тезис о глобальном противоречии целостности мироздания и отсутствии целостности знаний о нем, можно выделить несколько источников-причин неоднородности задач, тесно связанных со знаниями людей, решающих задачи в системах управления [1].

Обратимся к рис. 1.4, на котором представлены главные из источников. Задача здесь изображена окружностью, разделенной на сектора, что графически обозначает внутренне присущее ей многообразие — неоднородность.

Многообразие дисциплин в обучении (рис. 1.4,*a*). Этот источник объясняется отсутствием системности образования на всех его ступенях от яслей до докторантуры. Достаточно вспомнить вузовский учебный

план, содержащий десятки дисциплин, и то, что каждый из педагогов старается сформировать в мировоззрении студента свою «полочку» со знаниями. Естественность дисциплинарной редукции науки следует из ее все большей специализации и приводит к тому, что профессиональная деятельность в системах управления — это полиязыковая деятельность, например, математика, кибернетика, автоматика, механика, что, естественно, отражается на свойствах решаемых субъектами задач. Каждый сектор на рис. 1.4,а и обозначает только ту часть сложной задачи, которую субъект (эксперт или ЛПР) представляет, находясь в той или иной ипостаси.



Обозначения: ИИ — искусственный интеллект; ИСО — исследование операций; ТПР — теория принятия решений; ТСС — теория систем и системный анализ

Рис. 1.4 — Источники неоднородности

Многообразие методов и переменных в науке (рис. 1.4,б). Автономное развитие научных школ в прикладной науке, их изолированность, борьба за заказы от практиков привели к огромному разнообразию методов и технологий решения задач. Это — с одной стороны. С другой — метод всегда представляет ограниченное описание объекта, процесса, явления и процедуру нахождения решения задачи, что в свою

очередь тесно увязано с ограниченным набором используемых при решении переменных (см. разд. 3.1). Таким образом, научная школа, моделируя одну и ту же задачу, «видит» ее исключительно через призму своих собственных методов, отделяя от нее ту часть, которую способен «осилить» ограниченный по определению метод моделирования. В подобных случаях также не может быть создано целостное, системное представление о задаче. Модель предметной области неоднородна, подчеркивают Т.С. Миронова и Г.С. Плесневич, и не только потому, что неоднородна моделируемая предметная область, но также из-за того, что модель включает фрагменты, отвечающие различным взглядам (точкам зрения) на предметную область различных экспертов-разработчиков модели или различных (по типу) пользователей [46].

Многообразие мнений и моделей внешнего мира на практике (рис. 1.4,в). Отсутствие целостности знаний во внутреннем мире экспертов и ЛПР, где отдельные элементы знаний находятся в отдельных структурных единицах личного профессионального опыта, объясняется тем, что связи между знаниями специфицируются отнюдь не требованиями системности, а более энергонасыщенными элементами опыта: ощущениями, эмоциями, оценками, что во многом определяет личность. Следует помнить, что эксперт — это специалист, располагающий знаниями в достаточно узкой профессиональной области.

Такие ограниченные знания формируют у него и узкое представление задачи, по существу, только часть от общей, целой задачи, возникающей в практике управления и обработки информации. Разработчик же видит моделируемую систему глазами эксперта-управленца и, естественно, переносит эти представления в мир моделирования. Он, конечно, может получить знания о задаче у нескольких экспертов, однако если их даже и «сложить», то утверждать, что сумма составит целое нельзя.

Многообразие фаз управления (рис. 1.4,г). Следуя определениям фаз (см. разд. 7 онтологии на рис. 1.2), можно разделить соответствующие им задачи на простые, однородные, например, задачи учета, и задачи сложные, неоднородные, которые априори невозможно решить, если не иметь учетную, контрольную, прогнозную и другую информацию. В этом случае задачи планирования включают в себя более простые, однородные задачи учета, нормирования и прогнозирования. Это показано на рис. 1.4,г наложением друг на друга прямоугольников, обозначающих фазы управления. Не решив хотя бы одну из «вложенных» задач, как однородных частей более сложной неоднородной задачи, невозможно понять и решить всю задачу в целом. Однако решение всех вложенных задач еще не означает, что решена исходная задача и могут потребоваться новые и новые итерации решения однородных задач.

Кроме изображенных на рис. 1.4, есть и другие источники.

Многообразие целей решения задач. Многообразие целей в среде участников решения задачи в группах также приводит к ее многокритериальному, неоднородному характеру.

Многообразие отношений в знаниях. Знания играют огромную роль при решении задач в системах управления. Однако знания тем и отличаются от данных, что для их представления используется все многообразие отношений, встречающихся в языке профессиональной деятельности [4]. Отношения формируют пространственную, временную, причинно-следственную, включения, принадлежности и другие структуры из однородных задач, входящих в состав неоднородной. Именно это свойство и приводит к многообразию возможных декомпозиций сложной задачи, и нет уверенности в том, что какая-то из таких декомпозиций окажется лучше, а какая-то хуже, хотя можно предположить, что чем больше таких декомпозиций будет построено и исследовано, тем релевантнее станет модель процессам и явлениям реального мира [4].

Многообразие фокусов взаимодействия в мозге человека. Еще один источник неоднородности задач, как считает член-корреспондент РАН А. Иваницкий, скрыт в физической форме ее отражения в сознании человека.

Человеческая мысль рождается за счет интеграции активированных структур мозга. При этом в процессе мышления рисунок внутрикоровых связей между активированными структурами мозга изменяется таким образом, что они начинают собираться к определенным центрам-фокусам взаимодействия. Изучение мышления человека с применением нового метода картирования корковых связей позволило описать рисунок связей, типичный для разных видов мышления. Симметричный в состоянии покоя рисунок связей при решении задач изменяется: связи начинают сходиться к фокусам взаимодействия. Их топография различается в зависимости от вида мыслительной деятельности. При образном мышлении, например, определении эмоций изображенных на фотографиях лиц, фокусы локализовывались в теменно-височной коре. При абстрактно-вербальном мышлении, решении анаграмм или задач по категоризации слов они располагались в лобной коре. При пространственном мышлении, включающем элементы как образного, так и абстрактного мышления, связи сходились к теменной и лобной коре. Было также установлено, что информация поступает к фокусам из различных отделов коры, имеющих свою специализацию. В фокусе, нейронные группы которого соединены жесткими связями, происходит синтез поступающих сведений, вследствие чего, как предполагают ученые, и принимается решение [47].

Рассмотренные выше источники требуют нового взгляда на задачу с позиций системного анализа для учета возникающих при ее анализе (понимании) многообразия и нелинейности в реальном мире и нового взгляда на групповое решение задач с моделированием синергетических эффектов.

На рис. 1.5 субъективная сущность «задача» изображена как система. Для этого выделены ее элементы — подзадачи. Эти элементы обладают свойствами, например, зашумленности, нечеткости исходных данных, а также уникальными наименованиями. Для существования задачи как системы в ее составе должно быть минимум две подзадачи, обладающие разнородными свойствами.

Исходную задачу с подзадачами связывают отношения включения «целое-часть», обозначенные на рис. 1.5 пунктиром. Эти отношения задают состав задачи-системы, изменяющийся в определенных пределах, не затрагивая качества системы, т.е. ее эмерджентных свойств. Другая часть связей, уже между подзадачами, ограничивает степень свободы между элементами и не позволяет решать одни задачи без других, задавая порядок на причинно-следственной, временной и других шкалах.

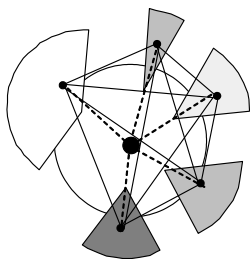


Рис. 1.5 — Задача как система

В настоящее время вопрос о полноте связей для того, чтобы систему можно было считать системой, не решен [48], хотя, следуя работе [49], можно предположить, что для того чтобы система-задача не распалась на части, необходимо обеспечить превышение суммарной силы (мощности) связей между элементами системы, т.е. внутренних связей над суммарной мощностью связей между элементами системы и элементами внешней среды. Под внешней средой можно понимать другие задачи-системы и «задачи-несистемы». К сожалению, измерения силы связей очень трудно реализовать и можно судить о них лишь косвенно.

Особое значение в определении задачи как системы имеет свойство эмерджентности, когда свойства системы не могут быть получены простым суммированием свойств ее отдельных элементов. Этот эффект можно наблюдать на примере классификации задач по фазам управления (разд. 7 онтологии на рис. 1.2): сколько бы задач учета, прогнозирования, нормирования не «суммировались», т.е. учитывались результаты их решения, решение задачи-системы на фазе «планирование» не будет получено до тех пор, пока не будет установлен порядок на множестве ее подзадач и сила этого порядка не превысит воздействие других задач.

Таким образом, смысл слова «планирование» и выражает в языке профессиональной деятельности эмерджентность задачи-системы.

Рассмотрим еще один вопрос: может ли элемент, в свою очередь, быть задачей-подсистемой и какое количество подзадач может быть включено в состав задачи-системы? На первый вопрос ответим утвердительно. Ответ на второй вопрос может зависеть от масштабности решения задачи. На примере трех исследованных предметных областей можно сказать, что задача-система обычно включает до семи — восьми задач-элементов. При этом можно высказать предположение о том, что это количество подзадач должно совпадать с числом Миллера (7 ± 2) [21], поскольку человек, решая целую задачу, должен удерживать в своей оперативной памяти ее части.

Будем называть задачу-систему — неоднородной, а задачу-элемент — однородной задачей. В гетерогенной задаче-системе, при переходе через «границу разделения» однородных элементов, происходят мгновенные скачки имен, параметров и характеристик, выражающих суть однородных задач, что, безусловно, должно учитываться при моделировании ее решения на ЭВМ.

1.4. Особенности сложных задач

В разд. 1.2 даны определения понятий «упрощенной» и «реальной задач», однако, на наш взгляд, имеет смысл не только специфицировать эти термины, но и пояснить эти различия.

Для этого рассмотрим, прежде всего, процесс возникновения в памяти человека упрощенных, игровых задач, характерный для рационализма в информатике, искусственном интеллекте, теории управления, теории принятия решений, структурной лингвистики, когнитивной психологии. До появления квантовых и релятивистских теорий, неравновесной термодинамики и фрактальных представлений, как отмечает В.Б. Тарасов в [13], научные школы исследовали простые системы с детерминированным или (и) периодическим поведением. Методология классической науки основывалась на следующих положениях: 1) предмет науки — общие (повторяющиеся), воспроизводимые и обратимые явления; 2) наука позволяет преодолеть неопределенность и случайность, с которыми следует бороться при описании явлений и открытии законов; 3) научное объяснение — это, прежде всего, объяснение свойств целого из свойств частей; 4) общенаучные положения должны выражаться как точное, непротиворечивое знание и описываться строгими логико-математическими моделями.

К данным положениям примыкал принцип механистичности и лапласовского детерминизма, которому подчиняется поведение простых систем: выход системы вычисляется по входным сигналам и состоянию системы. Кроме этого, рационализм обычно сочетается и принципом редукционизма, т.е. сведения реального сложного явления к сильно упрощенной модели.

На рис. 1.6 показано преобразование человеком информации «слева—направо» о реальной задаче, традиционное для парадигмы рационализма, детерминизма и редукционизма. В итоге такого преобразования, суть которого сводилась к введению некоторой совокупности ограничений типа «предположим, что в задаче использованы только детерминированные переменные, принимающие значения на множестве $\{0,1\}$ », оказывалось возможным получить игровую задачу как сильно упрощенную, ограниченную модель реальности. Далее предпринималась попытка встроить такую модель в автоматизированную систему обработки информации и управления. Однако в большинстве случаев оказывалось, что человек-управленец не принимает такую модель. Все дело в том, что научная школа, стоящая на позициях рационализма, «вырезает» отдельные части из неоднородной субъективной сущности, которые доступны пониманию в рамках того или иного метода моделирования, исповедуемого участниками данной школы.

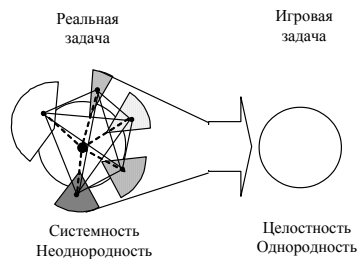


Рис. 1.6 — Преобразование задачи в методологии рационализма, детерминизма и редукционизма

При этом «... одни части системы моделируются преувеличенно точно, некоторые методики развиваются чрезмерно, а другие части системы из-за соображений математического удобства недопустимо упрощаются. Факторам, которым нельзя придать математическую форму, уделяется незначительное внимание» [50, с. 20].

«Внедрение» таких научных результатов ничего не меняет на практике. ЛПР по-прежнему решает общую (целую) задачу, и компьютерная технология «не вписывается» в его творческий процесс. Как цитирует в своей работе Р. Шеннон, самый большой недостаток моделей управления в том, что они никогда не используются на практике [51].

Однако положительными результатами рационализма является то, что он позволил наработать в разных дисциплинах множество методов моделирования решения игровых задач и установить соответствие «игровая задача — методы решения».

Методологический вред рационализма состоит в том, что он выработал в сознании разработчиков автоматизированных систем обработки информации неверную стереотипную схему последовательности действий: «сведение реальной задачи к одной из игровых — использование соответствия «игровая задача — методы» для выбора метода — разработка модели — моделирование решения задачи на компьютере — использование программы в составе автоматизированной системы обработки информации и управления».

Еще один вред состоит в том, что сознание человека, обученного на методологии рационализма и стереотипе «игровая задача — методы» перестает вообще воспринимать смысл понятия «реальная задача». Чаще всего это проявляется в вопросе: «А какая задача называется реальной?» (можно подумать, что есть какие-то другие задачи, которые ставят и решают в системах управления). А все дело в том, что ни одна реальная задача не укладывается в парадигму опыта рационализма, основанного на прецедентах «задача — метод».

И, тем не менее, попытаемся ответить на подобный вопрос, идя от стереотипа научного мышления к реальной жизни. Для этого выберем хорошо известную игровую задачу и попытаемся «вернуть» ей свойства, утраченные в ходе рационалистического преобразования «реальная задача → игровая задача».

Одна из таких популярных игровых задач — задача о коммивояжере (англ. *traveling salesman problem* — TSP) из теории графов. В области оптимизации дискретных задач задача коммивояжера служит своеобразным полигоном (эталоном), на котором испытываются новые методы. Кроме этого, она имеет и практическое значение в обширном классе приложений, например: распознавании траекторий, образов, построении оптимальных схем движения и т.д.

Коммивояжер (бродячий торговец) должен выйти из первого города, посетить по разу в неизвестном порядке города v_1, v_2, \dots, v_n и вернуться в первый город. Расстояния между всеми городами известны $d_{ij} = d(v_i, v_j) \mid i, j \in \overline{1; n}$ и являются целыми положительными числами. В каком порядке следует обходить города, чтобы замкнутый путь коммивояжера был кратчайшим? В терминах теории графов требуется найти гамильтонов цикл в графе минимальной длины.

Задача коммивояжера — *NP*-полная задача (см. разд. 6 онтологии на рис. 1.2) и для ее решения нет эффективного алгоритма.

В табл. 1.7 дана характеристика стандартной (классической) задачи о коммивояжере по шкале «оригинал — модель». Введение существенных упрощений (ограничений) на количество отображаемых объектов (города и дороги), использование одного единственного отношения

«находиться на расстоянии... от...» и только детерминированных переменных позволяют свести представление системы к реберно-взвешенному неориентированному графу, для которого может быть найден гамильтонов цикл наименьшей длины.

В [52] рассматривается одна из разновидностей задачи коммивояжера — задача коммивояжера с временными окнами. Здесь требуется проложить маршрут движения автомобиля, развозящего заказы потребителям. Все заказы изначально загружены в машину на складе — исходную точку маршрута. Для каждого потребителя задан промежуток времени, в течение которого он должен быть посещен машиной.

Таблица 1.7 — Характеристика классической (игровой) задачи коммивояжера

Оригинал	Модель		
	Теория графов	Отношения	Детерминированные переменные
Города	Вершины графа (точки на плоскости)	«Находиться на расстоянии... от...»	Количество вершин
Дороги	Ребра графа (отрезки линий на плоскости)	-	Количество ребер Расстояние между городами

Также заданы матрицы расстояний и времени проезда между потребителями. Если машина прибывает к потребителю до начала интервала его обслуживания, то она ожидает его начала, после чего начинается разгрузка.

Если машина прибывает к потребителю после завершения временно-го интервала, маршрут либо считается недопустимым (задача с жесткими временными окнами), либо подвергается дополнительному штрафу за опоздание (задача с мягкими временными окнами). Требуется найти маршрут посещения потребителей, обладающий, по возможности, наивысшим качеством.

Если сравнивать данный вариант задачи коммивояжера с классической (табл. 1.7), то можно заметить, что в постановку добавлены еще две детерминированные переменные: «время проезда между потребителями» и «интервал обслуживания потребителя». При этом граф модели превратился уже в вершинно- и в реберно-взвешенный.

Очевидно, что могут иметь место и другие постановки задачи коммивояжера. При этом расширяется не только количество детерминированных переменных, существенных для решения задачи, что оставляет задачу в классе однородных задач, но и вводятся новые отношения и

переменные из других классов: стохастических, лингвистических, логических и эволюционных [1].

В табл. 1.8 рассмотрен еще один вариант постановки задачи о коммивояжере. Здесь сняты ограничения на количество моделируемых объектов оригинала, добавлен «человеческий фактор», расширен список классов отношений между объектами и (или) их свойствами. Кроме этого учтены еще два класса переменных — стохастических, позволяющих учесть неопределенность, а также лингвистических для качественного описания поведения системы «города-дороги-люди».

Таблица 1.8 — Характеристика сложной задачи коммивояжера

Оригинал	Отношения	Образы, эмоции, характер	Модель		
			Переменные		
			Детерминированные	Стохастические	Лингвистические
Города Узлы Отели Вокзалы Пробки на улицах	Определения Включения Сравнения Пространственные Временные Причинности	Карта Схема	Число городов Число клиентов Расстояния: «город-вокзал», «клиент-клиент», «вокзал-клиент» Затраты Стоимость	Время задержки и обслуживания Погода	Названия Адреса Положение Престижность гостиницы Возможности агента Погода
Дороги Транспорт Потоки	Определения Сравнения Пространственные Временные Причинности	Карта Путь	Расстояния: «город-город», «город-поселок» Скорость Затраты Стоимость (проезда, сервиса) Время в пути	Вероятность проезда Время задержки и обслуживания Плотность потока Интервал движения Время прибытия Время отправления Погода	Названия (пунктов, дорог) Категория дороги Загрузка магистрали Вид транспорта Расписание Безопасность Опоздания Наличие билетов Расписание Очереди
Люди: ЛПР, агент, клиент, сервис	Семейные Подчиненности Согласования Принадлежность группе Дружеские	Активность Антиципация Надежность Гибкость Настойчивость Страх Эгоизм Жадность	Координаты агента Критерии выбора	Вероятность изменения маршрута Вероятность появления клиента	Субъективные оценки Личные предпочтения Критерии и риск выбора Здоровье, уверенность, Ошибки агента и сервиса Важность клиента

При этом табл. 1.8 не претендует на полный охват всего множества условий, обстоятельств, событий, процессов, переменных, актуальных для коммивояжера и ЛПР, которые стремятся достичь целей своего бизнеса. Задача таблицы в том, чтобы показать возможность постановки некоторого множества задач коммивояжера. Среди задач из этого множества есть и классическая постановка, однако есть и другие.

Например, можно рассмотреть область стохастического поведения системы «города-дороги-люди», добавив вероятность изменения (целенаправленного или по ошибке) ЛПР или самим агентом первоначально избранного маршрута. Этими оценками можно взвесить как города-вершины графа, так и дороги-ребра.

Можно не только расширить область стохастического поведения добавлением новых переменных, но сформировать еще одну область рассуждений. При этом использовать детерминированные количественные и качественные, двоичные (пропозициональные) переменные, учитывая в задаче названия пунктов, дорог, виды транспорта и его расписание, а также наличие билетов в городских кассах.

Можно сформировать области уверенного (неуверенного) поведения агента с нечеткими оценками важности тех или иных клиентов для агента, а также деятельности самого агента лицом, принимающим решения. Можно исследовать влияние на эти области экономических оценок деятельности агента с учетом затрат на командировку. Наконец, можно детализировать города и дороги до карт и схем транспортных узлов или укрупнять графические образы до городов-точек и дорог-отрезков линий на плоском графе.

Что произойдет, если выполнять рассмотренные выше действия по детализации, уточнению условий задачи в том или ином объеме? Очевидно, что следующее: 1) смещение задачи по шкале сложности слева—направо, от игровых, простых, однородных к реальным, сложным, неоднородным задачам, что на понятийном уровне означает уход от классической задачи коммивояжера и переход к понятию реальной (сложной, практической) задачи коммивояжера; 2) формирование множества «задач-систем», состоящих из различных областей однородных подзадач, причем отношения возникают как между задачами внутри области, так между задачами из разных областей; минимальное количество областей, комбинируемых в «задаче-системе» — два; 3) все большее количество разнородных переменных и их формальных и эвристических зависимостей должно быть учтено, чтобы «задача-система» могла считаться решенной по мере интеграции большего количества областей; при этом растет и структурная сложность задачи.

В графическом виде этот процесс, противоположный идеологии рационализма, детерминизма и редукционизма, показан на рис. 1.7. На

нем изображен переход («справа—налево») от понятия «игровая задача» к понятию «сложная задача» методом комбинирования (интеграции) пяти областей подзадач с однородными переменными. Подзадачи показаны точками, а связи между ними внутри областей — линиями.

Как результат комбинирования образуется множество моделей реальных практических задач. Элементы этого множества отличаются друг от друга: 1) количеством однородных областей, интегрируемых в задаче-системе; 2) количеством подзадач в областях; 3) различными по классам и количеству отношениями между подзадачами в различных областях.

Такое многообразие различий может породить множество моделей реальных практических задач достаточно большой мощности. Причем единственный критерий ответа на вопрос, а какая из этих моделей релевантнее сложной задаче — результаты моделирования и их сравнение с практикой.

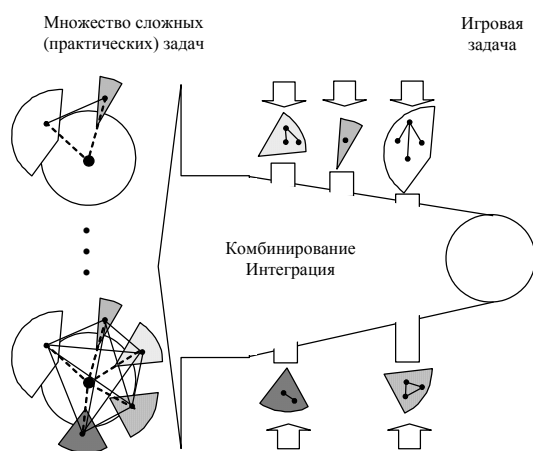


Рис. 1.7 — Преобразование игровой задачи комбинированием областей подзадач с однородными переменными

Можно подойти к определению понятия сложной задачи и с другой стороны, указав на ее специфику, особенности, свойства.

Первое свойство — это системность. Сложные задачи — это задачи-системы, и акцент в ходе их решения должен быть смещен на глубокое понимание задачи. Толкование задачи Р. Акофом и М. Сасиени (специалистами по исследованию

операций в период бурного становления методов оптимизации в кибернетике), хорошо работавшее долгие годы, отстало от эволюции задач обработки информации и управления, требует ревизии и наработки методов системного анализа, способствующих лучшему пониманию смысла сложной задачи.

Второе свойство — неоднородность. Сложные задачи — это неоднородные системы. На множестве подзадач, входящих в их состав, существуют подмножества с различными характеристическими свойствами,

в том числе по целям и критериям, что, в частности, приводит к проблеме многокритериальности. Такое разнообразие — следствие отсутствия целостности знаний в обществе и наличия множества парадигм в науке, что проявляется, например, в существовании биполярных шкал «непрерывное-дискретное», «детерминированное-случайное», «четкое-нечеткое», «отношение-число», «эвристика-зависимость», «лингвистическое-математическое» и т.п. Из этого следуют два важных обстоятельства. Первое — нет и не может быть одного единственного метода для решения сложной задачи. Вместо этого есть набор инструментальных средств экспертов для работы в пределах отдельных частей неоднородной системы, которые можно комбинировать для получения инструмента решения задачи-системы в целом. Второе — проблема эрудиции. Ее суть в том, что для автоматизации решения сложных задач недостаточно владеть одним методом (или одним классом методов). Необходим широкий спектр знаний в области теории и методов принятия решений, исследования операций, системного анализа, искусственного интеллекта, инженерии знаний, т.е. в тех дисциплинах, которые «взяли на себя ответственность» за моделирование механизмов решения задач человеком.

Третье свойство — динамичность. Сложные задачи имеют варьирующиеся во времени нефиксированные предметную и проблемную области, т.е. изменяющиеся данные и знания для решения задачи, а также состав подмножеств задач-элементов. Из этого следует, что нельзя вести речь о методе, один раз и навсегда разработанном и применяемом для решения сложной задачи. Метод, выработанный в данный момент времени, в другой момент времени уже может быть не актуальным. Значит, существует множество методов, нарабатываемых в системе управления за некоторый период времени.

Четвертое свойство — это неопределенность в ситуации решения задачи и необходимость измерения и анализа риска последствий. Неопределенность ситуации может быть вызвана неполнотой исходных данных и знаний экспертов, невозможностью учесть реакцию внешней среды на действия системы управления, а также неточное понимание своих целей ЛПР. Способ снятия неопределенности — субъективные оценки ЛПР и экспертов [21].

Пятое свойство — полиязыковый характер [1]. Поскольку сложные задачи — это задачи-системы, то и их решения — это система решений. Зависимость между двумя решениями определяется тем, что влияние одного из них на показатели функционирования системы обусловлено в свою очередь влиянием на него и другого решения. В сложных задачах недостаточно одной линии рассуждений, одного эксперта. Обычно для их решения привлекаются эксперты в разных предметных областях,

имеющих различные профессиональные языки. Директор института мозга и человека РАН С.В. Медведев отмечает, что работа мозга при решении сложной задачи — это как вспышка фейерверка: сначала — мириады разноцветных огней, которые начинают гаснуть, но вдруг вспыхивают ярче, разгораются, поблескивая и перемигиваясь друг с другом. Какие-то области остаются темными, другие переливаются всеми цветами радуги, охватывая новые и новые пространства [53]. Полиязыковый, неоднородный характер задачи требует и полиязыковых, неоднородных рассуждений.

Шестое — внутренняя несогласованность, как следствие неоднородности. По частичному решению сложной задачи, т.е. по решению подзадачи, не удастся сделать вывод о его пригодности, поскольку отсутствуют надежные способы проверки качества одних частных решений относительно других.

Рассмотренные особенности сложных задач требуют при создании автоматизированных систем обработки информации и управления других подходов и методов, значительно более эрудированных разработчиков и значительно больших усилий, а также тщательности при выполнении отдельных видов работ.

1.5. Решение задач в автоматизированном режиме

Информационные технологии — это совокупность методов, процессов, программных и технических средств, объединенных в технологическую цепочку, обеспечивающую сбор, обработку, хранение, распространение (транспортировку) и отображение информации с целью снижения трудоемкости процессов использования информационного ресурса, а также повышения их надежности и оперативности [54].

Информационная технология в своем развитии прошла несколько этапов: «ручная обработка», «механическая обработка», «электрическая обработка», «компьютерная (электронная) обработка».

Режим автоматизированных систем управления. Появление во второй половине 60-х годов больших производительных ЭВМ сместило акцент на обработку не формы, а содержания информации. Это было началом формирования «компьютерной» технологии, разработки автоматизированных систем управления для улучшения главным образом учетных функций управления.

Начиная с 70-х годов, центр тяжести развития АСУ сместился с учетных функций, отражающих только прошлое состояние объекта управления и не позволяющих оценить перспективы его развития, на аналитическую работу. В основе разработки таких АСУ лежала гипотеза формализуемости задач анализа и принятия решений. Внедрение по-

добных АСУ должно было повысить качество, полноту, объективность и своевременность информационного обеспечения ЛПР, эффективность работы которых будет расти по мере увеличения числа решаемых задач.

Первый опыт автоматизированного решения задач показал, что:

- 1) фон-Неймановские компьютеры, слабая периферия, неразвитые средства программирования интеллектуальных процедур обработки информации, несовершенство международных стандартов на межпрограммный обмен в принципе не позволяли перейти от автоматизированного решения учетных и оптимизационных задач к задачам выработки, выбора и анализа решений. На Всесоюзной конференции по АСУ в 1976 г. в г. Тбилиси заведующий кафедрой кибернетики МИФИ профессор, д-р техн. наук Л.Т. Кузин по этому поводу сказал, что те, кто занимается искусственным интеллектом, напоминают группу энтузиастов, которые на телеге пытаются достичь второй космической скорости;
- 2) экономико-математические модели имеют ограниченную сферу применения, в основном, для решения задач планирования на больших временных горизонтах;
- 3) поиск оптимальных решений рассматривался изолированно от других процедур принятия решений и наработки математиков и программистов складывались на полку;
- 4) развитие функциональной структуры АСУ (включение новых задач) требует разработки новых моделей, времени и плохо увязывается с совершенствованием процесса принятия решений;
- 5) методики оценки влияния АСУ на экономику не подходили для оценки качества и эффективности принимаемых решений;
- 6) следствием широкого распространения в административном аппарате информационных систем явилось стремление к сбору любой количественной информации и выражению всех данных в количественном виде. Особенно это характерно для США, где в ЭВМ хранится множество так называемых числовых индикаторов, на базе которых и строятся псевдообъективные модели [54];
- 7) уровень оснащенности ЭВМ (наличие «передового опыта») может позволить сотрудникам администрации не проводить основной работы по улучшению методов подготовки принимаемых решений. Наличие ЭВМ может дезориентировать совершенствование качества системы управления.

Таким образом, традиционные информационные системы, обеспечивавшие режим автоматизированного управления, мало что внесли инновационного в решение задач сложного выбора в новых, уникальных ситуациях.

Режим автоматизированных рабочих мест. Персональные ЭВМ принципиально изменили концепцию АСУ. Они обеспечили переход от

вычислительных центров и централизованного управления к локальным и глобальным вычислительным сетям и распределенному в пространстве и времени решению задач управления.

На программно-технической базе персональных компьютеров и их сетей с такой же, как и в АСУ фон-Неймановской архитектурой (надо отметить появление и применение к управлению и обработке информации ЛИСП-, ПРОЛОГ-, нейро- и нечетких процессоров, развитие идей квантовых и биокомпьютеров), разработаны мощные инструментальные среды объектно- и компонентно-ориентированного программирования, визуально-событийного проектирования приложений. Были созданы средства межпрограммного обмена данными, технологии использования средств одних программ другими, технологии разработки многоуровневых приложений, многозадачные операционные системы. Подобные технологии появились благодаря заметным успехам стандартизации в области информатики и средств вычислительной техники и созданию коммерческих программных продуктов, в том числе и поддерживающих методы и модели искусственного интеллекта.

Такой прогресс информационных технологий «доставил» на рабочее место конечного пользователя, в том числе ЛПР и специалистов: экспертные системы для имитации профессиональных рассуждений с использованием баз знаний, искусственные нейронные сети, обучающиеся решать задачи на примерах, генетические алгоритмы для поиска в условиях нелинейности, нескольких критериев и другой технологии ИИ.

Группы специалистов «получили» возможности коллективно, автоматизированно решать сложные задачи, используя программно-аналитическую обработку данных, «более удобные», чем базы данных хранилища данных, средства обнаружения знаний в базах данных. Это позволило получать от компьютера ответы на вопросы: «А что будет, если ...?» и объяснения ЭВМ-консультаций.

Опыт решения задач в режиме автоматизированных рабочих мест показал, что:

1) обозначилась устойчивая тенденция комбинаторного роста сложности решаемых индивидуально и коллективно задач управления. При этом степень этой сложности уже к концу XX в. превысила возможности наработанных наукой методов поиска решений. Становились все контрастнее особенности сложных задач (разд. 1.4) и влияние этих факторов на успешное завершение проектов интеллектуальных информационных систем;

2) усложнение информационных продуктов. Это следствие, по меньшей мере, двух обстоятельств. Во-первых, усложняются задачи, которые решает человек на практике, что является следствием постепенного перехода общества к информационной (постиндустриальной) модели. Во-

вторых, постепенно в науку приходит понимание того, что пора решения игровых, существенно упрощенных задач прошла и необходимо программировать обработку информации такой, какова она есть на практике. Оба обстоятельства неизбежно приводят к усложнению информационных продуктов, к внесению в них интеллекта человека, решающего задачу на практике. Мощных программно-технических средств уже недостаточно, чтобы бороться со все возрастающей сложностью создаваемых интеллектуальных информационных систем. По оценкам в 1998 г. Standish Group CHAos Report только 26% проектов информационных систем заканчивались успешно [55];

3) существующая узкая дисциплинарная специализация научных школ приводила к одностороннему, ограниченному, несистемному рассмотрению сложных явлений и не позволяла выработать методологии и технологии «борьбы» с комбинаторным взрывом сложности задач;

4) компьютерные технологии по-прежнему остаются средой реализации методов, а не инструментальным средством их синтеза;

5) по-прежнему актуально обеспечение совместимости, т.е. свободного обмена различной информацией, представленной в удобном для восприятия человеком виде: речь, зрение, осязание, обоняние. Решение этой проблемы требует полного соответствия и взаимной сопряженности компонентов телефонии, обработки данных, средств ввода-вывода информации, передачи данных, хранения и преобразования, аудио- и видеоинформации.

Таким образом, применение режима автоматизированных рабочих мест к индивидуальному и коллективному принятию решений, безусловно, повысило качество информационных технологий, однако контрастировало и серьезные методологические проблемы. Истоки этих проблем лежат во все еще слабой изученности явлений, процессов и закономерностей окружающего нас мира, влияния НЕ-факторов [44], в частности, неоднородности, на решение задач человеком в системах управления.

1.5. Новые подходы к решению сложных задач в автоматизированном режиме

Синергетический искусственный интеллект. Как отмечает В.Б. Тарасов, в 60—70-е годы происходит прорыв в понимании процессов самоорганизации в самых разных явлениях природы, в частности, разработаны теория генерации лазера (Г.Б. Басов, А.М. Прохоров, Ч. Таунс, Г. Хакен); колебательные химические реакции как основа биоритмов живого (Б.П. Белоусова, А.М. Жаботинский) и теория диссипативных структур (И. Пригожин); теория активных сред и биофизические при-

ложения (А.С. Давыдов, Г.Р. Иваницкий, И.М. Гельфанд, Д.С. Чернавский); динамический хаос в задачах прогноза погоды (Э. Лоренц), неустойчивость решения по начальным данным, знаменитый «эффект бабочки», когда взмах ее крыльев может радикально изменить дальний прогноз (Л.П. Шильников); теория катастроф, скачкообразных изменений состояний систем (Р. Том и В.И. Арнольд) и ее приложения в психологии и социологии; теория автопоэзиса живых систем (У. Матурана, Ф. Варелла) [13].

Круг этих методов и подходов в изучении сложных систем Г. Хакен и назовет в 1970 г. синергетикой — теорией коллективного, кооперативного, комплексного поведения систем.

В 80 — 90-е годы продолжается изучение динамического хаоса и проблемы сложности: развивается фрактальная геометрия для эффективного сжатия и хранения информации; обнаружены универсальные сценарии перехода к хаосу; открыт феномен самоорганизованной критичности в поведении сложных систем для моделирования биржевых кризисов, землетрясений, аварий сложных технических комплексов и т.д.; клеточные автоматы и нейрокompьютеры для моделирования активных сред и социальных явлений, распознавания образов и обучения.

Синергетика актуальна из-за отсутствия общенаучной, междисциплинарной, единой картины мира, как самосогласованной целостности, о чем шла уже речь в разд. 1.3. Есть ее отдельные фрагменты: философская, физическая, химическая, биологическая и другие картины мира, репрезентирующие предметы отдельных научных дисциплин. Необходимость выработки междисциплинарного подхода к осмыслению синергетически развивающихся человекомерных систем имеет огромное мировоззренческое и практическое значение.

В этой связи один из основоположников синергетики немецкий ученый Г. Хакен пишет, что синергетику можно рассматривать как науку о коллективном поведении, организованном и самоорганизованном, причем поведение это подчинено общим законам. Синергетика опирается на разные дисциплины, среди которых не только физика, химия и биология, но также социология и экономика. Можно поэтому ожидать, что открытые и описанные синергетикой закономерности уже так или иначе будут представлены в различных областях науки и появится возможность увидеть возникновение в свете синергетики новой, единой картины мира, составленной, подобно мозаике, из множества отдельных, собранных наукой фактов [56].

Предполагается, что синергетика позволит создать системы искусственного интеллекта с формальной логикой, образным мышлением и интуицией, моделирующие работу мозга как процесс самоорганизации,

в результате которой устанавливается определенная структура связей, настроенная на решение данной задачи.

Синергетический подход в интеллектуальных системах проявился не сразу. Изначально здесь преобладали индивидуалистический подход и разработка обособленных, централизованных программных систем. Если эти системы могли приобрести достаточный объем профессиональных знаний, то они могли решать практические задачи. «При этом игнорируются коммуникативные и регулятивные процессы, что принципиально ограничивает моделирование естественного интеллекта» [13, с. 71]. Кроме того, гипотеза интеллектуальных систем, основанных на знаниях, предполагает самодостаточность категории профессиональных знаний. Однако в процессе решения задач человек не ограничивается только категорией профессиональных знаний, а использует также и теоретические знания, опыт, убеждения, намерения, желания и т. п.

Результат естественного развития системного подхода — появление синергетического искусственного интеллекта, изучающего сложные, самоорганизующиеся интеллектуальные системы. В этом подходе исследуются процессы зарождения, формирования, деятельности, коммуникации, эволюции и кооперации сложных, открытых интеллектуальных систем различных классов. Примерами таких систем могут быть: гибридные интеллектуальные системы, системы мягких вычислений, системы с кооперативным поведением, интеллектуальные агенты, распределенные системы управления, многоагентные системы, виртуальные коллективы, интеллектуальные организации, эволюционирующие искусственные сообщества.

В синергетическом искусственном интеллекте изучаются нестационарные состояния, динамика, взаимные переходы, способы разрушения и создания сложных интеллектуальных систем. Это направление призвано обеспечить необходимые теоретические предпосылки для объединения интеллектуальных систем.

В русле системно-организационного подхода к интеллекту В.Б. Тарасова радикально изменяется точка зрения на объект и предмет ИИ. Классическая парадигма: объект — это человеко-компьютерные системы инженерии знаний; предмет — разработка теории, методологии и программно-аппаратных средств для решения, основанные на знаниях (правилах) задач для выполнения отдельных интеллектуальных функций человека-эксперта. Согласно синергетической парадигме: объект исследований — группа или сообщество неоднородных, взаимодействующих агентов; предмет — разработка теории, методологии и программно-аппаратных средств для: 1) синтеза индивидуальных свойств и поведения агентов по моделям групповой динамики на отношениях кооперации и конкуренции, конфликта и сотрудничества, субординации и

координации; 2) построения организационных структур из агентов на базе функциональной структуры организации, определения состава агентов и их ролей [13].

В.Б. Тарасов сформулировал и основные задачи синергетического искусственного интеллекта. Перечислим их.

1. Исследование кооперативных механизмов зарождения, самоорганизации и эволюции интеллектуальных систем и управления организаций.

2. Изучение задач и установление принципов формирования неклассических (неоднородных, открытых, распределенных, децентрализованных, локально организованных) интеллектуальных систем с гибридными архитектурами в процессе взаимодействия (кооперации) исходных систем и компонентов. Моделирование эффектов «резонансного возбуждения», вступающих во взаимодействие объектов.

3. Исследование путей и характера эволюции (дивергенции и конвергенции) различных направлений, школ и подходов в искусственном интеллекте, интеграция моделей искусственного интеллекта и других научно-практических дисциплин.

4. Определение сценариев развития и способов объединения различных интеллектуальных (и не только интеллектуальных) технологий, а также возможных видов гибридных архитектур.

5. Исследование различных НЕ-факторов знаний и взаимодействия между ними; построение гибридных логик — неклассических логик ИИ, где в явном виде сочетаются описания НЕ-факторов по крайней мере двух типов; разработка нетрадиционных (нейронечетких, эволюционных) семиотических моделей и систем; представление смысла на различных шкалах.

Рассмотрим подробнее несколько концепций, развивающихся в рамках парадигмы синергетического искусственного интеллекта: гибридные системы, гибридные (интегрированные) экспертные системы, многоагентные системы и гибридные интеллектуальные системы.

Гибридные системы. Гибридная система — записанная на математическом языке комбинация двух методов, один из которых имеет преимущества в моделировании непрерывных процессов, а другой — дискретных.

Состояние гибридной системы представляется вектором двух переменных — непрерывной и дискретной. Непрерывная динамика гибридных систем в непрерывном, дискретном и/или смешанном времени задается обычно дифференциальными уравнениями, а дискретная управляется цифровым автоматом или передаточной функцией с конечным количеством состояний. Оба (непрерывный и дискретный) процесса взаимодействуют в моменты возникновения «событий» или «переключений», когда очередное состояние достигает некоторых заданных

множеств в непрерывном пространстве состояний. Иными словами, гибридная система — это индексированная совокупность динамических систем и таблица «скачков» между ними.

Гибридная природа таких систем, например, автоматики с реле, ключами, гистерезисом, контроллеров перемещения, роботов, гибких производственных систем, автопилотов, определяется непрерывно-дискретным характером их поведения, для описания которого недостаточно аналитических знаний в виде дифференциальных уравнений и необходимо комбинировать их со знаниями из дискретной математики, математической статистики и логики. Гибридным системам присущи четыре феномена: автономные переключения, когда состояние изменяется мгновенно, если непрерывная компонента достигает некоторой границы; автономные скачки, если значение непрерывной компоненты состояния скачкообразно изменяется при достижении заданных областей; управляемые переключения, когда дискретная переменная состояния изменяется скачком в ответ на управляющую команду и управляемые скачки, во время которых значение непрерывной компоненты состояния мгновенно изменяется в ответ на управляющую команду.

Один из ранних результатов (1966 г.) по искусственной гибридации в детерминированных динамических системах с континуально-дискретным поведением был получен W. Witsenhausen. В последующем [57 — 61] были разработаны базовые модели гибридных систем.

Есть существенная терминологическая разница между использованием понятия «гибридная система» в российских и зарубежных работах. В первых это понятие используется как родовое и им обозначает более широкий класс систем, включающих, в том числе, и гибридные интеллектуальные системы, и интегрированные экспертные системы.

Гибридные экспертные системы. Экспертные системы решают трудные задачи на уровне хорошего специалиста. Они способны объяснять решения. Именно экспертные системы стали первым, широко используемым на практике результатом работ в области искусственного интеллекта. Однако в [62] Д.А. Поспелов указывает, что сегодня уже ясно, что поколение экспертных систем, сделавшее исследования в области ИИ прикладными и интересными для промышленности, сельского хозяйства, науки, проектных организаций, управления уже изживает себя. На смену ему должно прийти новое поколение экспертных систем.

Новое поколение экспертных систем получило у ученых и специалистов различные названия: «экспертные системы второго поколения» [62], «гибридные экспертные системы» [63], «интегрированные экспертные системы» [64], «мягкие экспертные системы» [65].

Определяя понятие «гибридная экспертная система», Д.А. Поспелов [62] указывает на необходимость сращивания ранее известных моделей и методов с теми, которые предлагают экспертные системы, и отмечает, что в таких системах обычная экспертная система первого поколения играет роль интеллектуального интерфейса, позволяющего пользователю выходить на модели и методы традиционного математического характера. По его мнению, парадигма экспертной системы второго поколения может быть сформулирована так: «интеллектуальный интерфейс + база знаний + решатель + пакет прикладных программ». Последняя составляющая и делает экспертную систему гибридной. Решатель в такой системе обслуживает не только базу знаний, но и выполняет функции монитора по отношению к пакету программ. Этот пакет может выполняться и видоизменяться не только извне, но и в результате работы самой экспертной системы. Например, решатель по результатам выполнения программ из пакета может видоизменять стратегию выбора программ из пакета. Таким образом, экспертные системы второго поколения будут реализовывать функции не только интеллектуального интерфейса, но и интеллектуального вычислителя. Такие системы будут объединять возможности символьной обработки и обработки зрительных образов. В них будет решена проблема структуризации баз знаний. В таких базах знаний существуют кластеры информации, в которых сосредотачиваются знания, относящиеся к стандартным типовым ситуациям и гиперсобытиям. В таких базах знаний действует механизм вероятностного прогнозирования, позволяющий сделать наиболее доступной для использования информацию из тех кластеров, к которым чаще всего обращается решатель. Наконец, экспертные системы второго поколения должны обладать способностью обучаться профессиональной деятельности путем прямого наблюдения за деятельностью эксперта.

Это был прогноз, сделанный в 1989 г., а в 1992 г. в Санкт-Петербурге прошел научно-технический семинар «Гибридные экспертные системы в задачах проектирования сложных технических объектов», на котором А.Н. Борисов сделал доклад «Принципы построения и реализации гибридных экспертных систем». В своем докладе он, в частности, отмечал, что повышенный интерес к гибридным экспертным системам можно объяснить двумя причинами. Во-первых, комплексностью знаний, используемых при решении прикладных проблем, и возможностью использования всех их составляющих. Во-вторых, наличием разработанного ранее программного обеспечения, способного решать часть задач

из состава проблемы. Эту часть необходимо сделать совместимой с разработкой экспертных систем без переписывания программ.

В этом же докладе А.Н. Борисов дает и определение гибридных экспертных систем через набор характерных для них свойств: 1) наличия различных возможностей представления знаний; 2) наличия различных средств поиска или вывода решения; 3) использования нескольких уровней абстракции знаний о проблеме; 4) возможности использования совокупности языков реализации системы.

До настоящего времени понятие «гибридная экспертная система» дошло практически без изменения смысла. Так, например, А.В. Гаврилов в [66, с.51] дает следующее определение. «Обычно под гибридными экспертными системами понимают экспертные системы, в которых реализуются разные парадигмы (методы, модели) представления и интерпретации знаний, а также подсистема взаимодействия с внешним миром не сводится только к пользовательскому интерфейсу». Специфика второй части определения, скорее всего, связана с необходимостью распространения определения не только на статические, но и на динамические экспертные системы [29].

В конце 90-х годов прошлого столетия стало ясно еще одно обстоятельство: традиционные компьютерные вычисления оказались «слишком точны» для задач реального мира. Для решения ряда проблем в сложных технических системах, в системах экономического планирования, в социальных системах большой размерности оказалось невозможно получить полную информацию, и Л. Заде ввел принцип несовместимости: с ростом сложности систем человеческая способность делать точные утверждения о них падает [67].

В этот период сформировалось и выросло новое направление в искусственном интеллекте, называемое мягкими вычислениями (англ. soft computing) или вычислительным интеллектом (англ. computation intelligence). Компоненты, входящие в комплекс мягких вычислений, используют подходы, свойственные человеку в его оценках окружающего мира, а также технологии, заимствованные у природы. В 1994 г. Л. Заде ввел термин «мягкие вычисления» в следующем виде: «мягкие вычисления = нечеткие системы + нейронные сети + генетические алгоритмы». Такое взаимное использование ряда технологий позволяет усиливать их достоинства и ослаблять недостатки их отдельных составляющих.

Эти события нашли естественное отражение и в технологии экспертных систем. Появились нечеткие системы [1, 67], нечеткие экспертные системы, использующие представление знаний в форме нечетких

продукций и лингвистических переменных [65]. Основу представления лингвистической переменной составляет терм с функцией принадлежности. Способ обработки знаний в нечетких экспертных системах — это логический вывод по нечетким продукциям. Особенность нечеткой экспертной системы — способ извлечения функций принадлежности, который сводится либо к методам математической статистики, либо к методу экспертных оценок.

В рамках мягких вычислений нечеткие экспертные системы трансформировались в мягкие экспертные системы. В своей работе [65, с. 201] Н.Г. Ярушкина дает следующее определение мягкой экспертной системы: «Мягкой экспертной системой будем называть нечеткую экспертную систему, обладающую следующими особенностями: 1) использует статистические данные, которые интерпретирует как обучающие выборки для нечетких нейронных сетей; 2) представляет знания в виде лингвистических переменных (функций принадлежности), нечетких продукций и обученных нейронных сетей».

Интегрированные экспертные системы. Интегрированная экспертная система — это программная система, в архитектуре которой наряду с традиционным компонентом «экспертная система», используемым для решения неформализованных задач, как правило, методологию простых продукционных экспертных систем, содержатся некоторые компоненты, расширяющие функциональные возможности экспертных систем, например, БД, пакеты прикладных программ, обучающие системы и т.д. [64].

Несколько основных причин обусловили интеграционные процессы, центральное место в которых отводится методам и средствам экспертных систем.

1. Мировое сообщество накопило хороший опыт создания и эксплуатации экспертных систем, и «расставаться» с таким заделом не имеет никакого смысла.

2. Экспертные системы хороши как управляющие модули в комбинациях программных продуктов.

3. Особенно ярко экспертные системы «технологически» отстали на фоне индустрии Web-технологий, технологии объектной модели компонентов (англ. component object model — COM), применяющейся для связи объектов различных языков и сред программирования в распределенных приложениях.

4. Неформализованные задачи, для автоматизированного решения которых традиционно используются экспертные системы, не существуют изолированно от формализованных задач. Эти задачи — составная часть задач реальной практической сложности и значимо-

сти, для решения которых используются комбинации методов традиционного программирования.

Рассмотренные причины и породили новые архитектуры программных систем, реализовывать которые в рамках методологии и инструментальных средств создания экспертных систем (например, оболочек) малоэффективно, трудоемко и дорого [64].

Среди отечественных фундаментальных исследований в области создания элементов концепции ИЭС наиболее известны работы А.И. Эрлиха [68] и Г.В. Рыбиной [64, 69, 70]. Из зарубежных работ можно выделить инструментальную систему G2 (Gensym Corp.), которая может быть использована для разработки динамических ИЭС.

Обзор основных классов программных систем «Интегрированные экспертные системы» дан в разд. 4.

Г.В. Рыбина отмечает [64], что в настоящее время существуют разночтения в использовании для ИЭС терминов «гибридный» и «интегрированный», что, по-видимому, объясняется историческими «корнями», связанными с ранней работой [71], в которой ИЭС были определены как «гибридные экспертные системы».

Термин «интегрированные экспертные системы» в [64] рекомендуется использовать для сложных программных систем, объединяющих методы и средства экспертных систем с технологиями традиционного программирования. Термин «гибридные экспертные системы» целесообразно применять как для экспертных систем, так и для инструментальных средств разработки экспертных систем с гибридными способами представления знаний, т.е. объединяющих, например, внутри одного инструментария различные модели представления знаний и механизмы функционирования.

Многоагентные системы. Многоагентные системы зародились на пересечении теории систем и распределенного искусственного интеллекта (англ. distributed artificial intelligence — DAI). Это открытые, активные, развивающиеся системы, в которых главное — процесс взаимодействия агентов как причина возникновения системы с новыми качествами. Многоагентные системы строятся по принципам распределенного интеллекта как объединение отдельных интеллектуальных систем, обладающих своими базами знаний и средствами рассуждений [13].

Основной элемент многоагентных систем — это «агент» как развитие известного понятия «объект», представляющего абстракцию множества экземпляров предметов реального мира, имеющих одни и те же свойства и правила поведения. Под интеллектуальным агентом понимаются физические или виртуальные единицы, поддерживающие взаимодействие с окружающим миром и получающие от него информацию. Агенты реагируют на эту информацию своими действиями, проявляют

собственную инициативу, посылают и получают сообщения от других агентов, вступая с ними во взаимодействие. Агенты действуют без вмешательства извне, в том числе и без вмешательства человека.

Известно множество классификаций агентов, например: 1) автономные агенты, персональные ассистенты, интеллектуальные агенты, социальные агенты и т.д.; 2) в зависимости от внутреннего представления внешнего мира и поведения агенты различаются на локальные, сетевые, мобильные, интерфейсные, транслирующие, маршрутизации и т.д.; 3) выделяют простые, умные (смышленные), интеллектуальные (интеллектуалы) и действительно интеллектуальные (табл. 1.9) агенты.

Таблица 1.9 — Типы и признаки агентов

Признак	Тип агента			
	Простой	Умный	Интеллектуал	Действительно интеллектуальный
Автономность	+		+	+
Взаимодействие с другими агентами (и/или пользователем)	+	+	+	+
Реактивность	+	+	+	+
Способность абстрагировать		+	+	+
Адаптивное поведение		+	+	+
Обучение			+	+
Толерантность к ошибкам			+	+
Функционирование в РВ			+	
Взаимодействие на ЕЯ			+	

Обозначения: РВ — реальное время; ЕЯ — естественный язык; + — наличие признака у агента

Интеллектуальный агент, помимо перечисленных в табл. 1.9, должен обладать и следующими свойствами: 1) социальным поведением — возможностью взаимодействия и коммуникации с другими агентами; 2) активностью, т.е. способностью генерировать цели и действовать рациональным образом для их достижения; 3) базовыми знаниями, т.е. знаниями о себе, окружающей среде (включая других агентов), которые не меняются в рамках жизненного цикла агента; 4) убеждениями, т.е. переменной частью базовых знаний, которые могут меняться во времени, хотя агент может об этом не знать и продолжать их использовать для своих целей; 5) целями, т.е. совокупностью состояний, на достижение которых направлено текущее поведение агента; 6) желаниями, т.е. состояниями и/или ситуациями, достижение которых для агента важно; 7) обязательствами, т.е. задачами, которые берет на себя агент по просьбе и/или поручению других агентов; 8) намерениями, т.е. тем, что агент

должен делать в силу своих обязательств и/или желаний; 9) индивидуальной картиной мира: каждый агент имеет собственную модель окружающего его мира, которую он строит на основе информации из внешней среды; 10) умением планировать.

Иногда в этот же перечень добавляются и такие свойства, как рациональность, правдивость, благожелательность и др.

Следовательно, каждый агент — это процесс, владеющий частью знаний об объекте и обменивающийся этими знаниями с другими агентами (процессами).

Система, в которой агенты могут общаться друг с другими, передавать друг другу информацию, взаимодействовать, называется многоагентной. В таких системах задача решается несколькими интеллектуальными подсистемами. При этом она разбивается на несколько подзадач, распределяемых между агентами.

По сути многоагентные системы можно рассматривать как совокупность взаимосвязанных программных модулей (агентов), являющихся фрагментами знаний, доступных другим агентам. Они координируют свои знания, цели, умения и планы при решении проблем.

Многоагентные системы подразделяются на кооперативные, конкурирующие и смешанные. Агенты в кооперативных системах — части единой системы и решают подзадачи общей задачи. При этом агент не может работать вне системы и выполнять задачи самостоятельно. Конкурирующие агенты — самостоятельные, объединяющие усилия для достижения целей системы. При этом связь с другими агентами не обязательна. Под смешанными агентами понимаются конкурирующие агенты, подсистемы которых также агенты.

Некоторые архитектуры многоагентных систем и их характеристики приведены в табл. 1.10.

Таблица 1.10 — Архитектуры многоагентных систем

Архитектура	Представление знаний	Модель мира	Решатель
Интеллектуальная	Символьное	Исчисление	Логический
Реактивная	Автоматное	Граф	Автомат
Гибридная	Смешанное	Гибридная	Машина вывода

При исследовании сложных систем с использованием технологии многоагентных систем реализуются следующие фундаментальные идеи.

Агенты могут работать как особи или как коллектив. В первом случае система очень проста: агенты делают то, о чем их просят. В возникающих структурах агенты организуются в целое, которое больше, чем просто сумма составляющих. Простые правила могут генерировать ло-

гически связанное новое явление. Устойчивые возникающие структуры могут стать компонентами более сложных возникающих структур и приводят к иерархии возникающих структур и их масштабируемости.

Множество агентов могут быть гомогентными или гетерогентными. Большинство систем создается на основе гетерогентности с использованием действия различных видов агентов.

В настоящее время многоагентные системы рассматриваются уже не только как принципиально новая информационная технология, сформированная на базе слияния информационных и телекоммуникационных технологий, но и как новая парадигма программирования, в какой-то мере альтернативная объектно-ориентированному программированию.

Разработка многоагентных систем идет по двум основным направлениям: системы на основе распределенного искусственного интеллекта и системы на основе сценариев.

Многоагентные системы на основе распределенного искусственного интеллекта строятся на базе систем, основанных на правилах и рассуждениях на основе прецедентов. В них агент — система, основанная на знаниях с добавлением компонентов, обеспечивающих безопасность, мобильность, качество обслуживания, взаимодействие с другими агентами, сетевыми ресурсами и пользователями.

Многоагентные системы на основе сценариев ориентированы на использование в компьютерных сетях с большим разнообразием предоставляемого сервиса. Эти системы рассматриваются как основной подход для реализации мобильной телекоммуникации через компьютерные сети общего пользования с помощью переносных компьютеров.

Актуальность многоагентных систем обусловлена следующими факторами. В некоторых предметных областях каждого участника какого-либо процесса логично представить агентом. Например, процессы коллективного (группового) принятия решений, социальные процессы, в которых каждый из участников играет свою роль. В других предметных областях нужны параллельные выполнения для решения задач. Тогда такая предметная область представляется совокупностью независимо решающих задачи агентов. В третьих предметных областях требуется устойчивая работа системы, когда контроль и ответственность за выполняемые действия распределены между несколькими агентами. При отказе одного агента система не перестает функционировать.

Проблемы многоагентных систем состоят в том, что при проектировании точной и полной модели представления мира, процессов и механизмов рассуждения в нем — очень тяжело создать адекватную и полную картину мира. Несмотря на явные трудности, идея использовать агентов для решения разноплановых задач очень популярна в последнее

время. Задача проектирования многоагентных систем и действительно интеллектуальных агентов требует специальных знаний и ресурсоемка.

Гибридные интеллектуальные системы. Термин «интеллектуальные гибридные системы» появился в 1992 г. [72]. Авторы вкладывали в него смысл гибридов интеллектуальных методов, таких как экспертные системы, нейросети и генетические алгоритмы. Экспертные системы представляли символьные, а искусственные нейронные сети и генетические алгоритмы — адаптивные методы искусственного интеллекта. Однако, в основном, новый термин касался достаточно узкой области интеграции — экспертные системы и нейросети. Ниже приведены несколько трактовок этой области интеграции другими авторами.

«Гибридный подход» предполагает, что только синергетическая комбинация нейронных и символьных моделей достигает полного спектра когнитивных и вычислительных возможностей (способностей) [73].

Термин «гибрид» понимается как система, состоящая из двух или более интегрированных подсистем, каждая из которых может иметь различные языки представления и методы вывода. Подсистемы объединяются вместе семантически и по действию каждая с каждой [74].

Ученые Центра Искусственного интеллекта Cranfield University (Англия) определяют «гибридную интегрированную систему» как систему, использующую более чем одну компьютерную технологию. Причем технологии накрывают такие области, как системы, основанные на знаниях, коннекционистские модели и базы данных. Интеграция технологий дает возможность использовать индивидуальную силу технологии для решения специфических частей задачи. Выбор технологий, внедряемых в гибридную систему, зависит от особенностей решаемой задачи.

Специалисты из University of Sanderland (Англия), входящие в группу HIS (англ. Hybrid Intelligent Systems), определяют «гибридные информационные системы» как большие, сложные системы, которые «бесшовно» (цельно) интегрируют знания и традиционную обработку. Они могут предоставлять возможность хранить, искать и манипулировать данными, знаниями и традиционными технологиями. Гибридные информационные системы будут значительно более сильными, чем экстраполяции концепций существующих систем.

В работе [75] авторы ввели понятие «гибридной интеллектуальной адаптивной системы», подчеркивая тем самым актуальность самоорганизации, обучения в ГиИС.

Н.Г. Ярушкина в работе [65, с. 9] вводит определение «мягкой интеллектуальной системы»: «мягкая интеллектуальная система = управление неопределенностью + обучаемость + самоадаптация». Смысл определения — это раскрытие основных процессов, протекающих в ГиИС через преимущества их трех основных компонентов: нечетких

систем, искусственных нейронных сетей и генетических алгоритмов соответственно.

Обобщая сказанное, можно утверждать, что под гибридной интеллектуальной системой принято понимать систему, в которой для решения задачи используется более одного метода имитации интеллектуальной деятельности человека. Такому определению соответствует графическое представление ГиИС на рис. 1.8.



Рис. 1.8 — К понятию гибридных интеллектуальных систем

Таким образом, междисциплинарное направление «гибридные интеллектуальные системы» объединяет ученых и специалистов, исследующих применимость не одного, а нескольких методов, как правило, из различных классов (см. разд. 3), к решению задач управления и проектирования.

Актуальность ГиИС аргументируем с различных точек зрения.

Аргументация с позиций теории систем. Очевидно, что изначально в искусственном интеллекте, взявшем на себя бремя теории решения творческих задач управления и проектирования, принцип системного подхода был нарушен [42], поскольку метод моделировал решение сложной задачи только частично, без учета ее состава и структуры, либо пытался вырасти до универсальной теории, чтобы объять необъятное — всю задачу-систему. В работе [76] отмечается, что процессы познания неоднородны. Поэтому уместно широкое разнообразие представлений и механизмов. Отсюда требуется плюрализм.

В синергетическом ИИ постулируется, что, имея плюсы и минусы (табл. 1.11), ни один из классов базисных методов (см. разд. 3) не может, да и не должен претендовать на универсальность. Преимущества одних представлений, компенсируя слабые стороны других представлений за счет взаимосвязей частей целого, дадут новое интегративное свойство — качественно новую, полезную сущность. В итоге можно ожидать более полного задачно-ориентированного описания исследуемой предметной области.

Таблица 1.11 — Преимущества и недостатки методов гибридных

интеллектуальных систем

Характеристики	Классы базисных методов					
	АМ	ИСМ	ИНС	ГА	ЭС	НС
Обучение на примерах	-	-	+	-	-	-
Адаптация	+	-	+	+	-	-
Аргументация	-	-	-	-	+	+
Интерактивность	-	+	-	-	+	-
Работа с шумом и неопределенностью	-	+	+	+	-	-
Производительность	Очень высокая	Низкая	Высокая	Низкая	Ниже высокой	Очень высокая
Трудозатраты	Высокие	Высокие	Средние	Низкие	Средние	Очень низкие

Обозначения: АМ — аналитические методы; ИСМ — имитационное статистическое моделирование; ИНС — искусственные нейронные сети; ГА — генетические алгоритмы; ЭС — экспертные системы; НС — нечеткие системы; + — преимущества, - — недостатки

Аргументация с позиций системного анализа. Обратимся к табл. 1.12, в которой творческая деятельность ЛПР условно разбита на три чередующихся в едином цикле этапа со своей, скорее всего, толерантной, процедурно-понятийной базой, сформированной из анализа литературы [4, 7, 77, 78, 80 — 84], а также по результатам участия авторов в многочисленных коллективных управленческих мероприятиях.

Табл. 1.12 показывает потенциал методов для реализации процедур того или иного этапа творческой деятельности ЛПР. Анализ таблицы показывает, что приблизительно из 25 процедур каждый класс методов «накрывает» только от четырех до девяти процедур. Отсюда видны слабые шансы (примерно 1 : 4) моделирования решения задач в рамках одного метода. Однако примененные совместно, комбинированно, они могут быть релевантны всему полю процедур.

Аргументация с позиций теории принятия решений, исследования операций. Идея оптимизации — центральная идея кибернетики, теории принятия решений и исследования операций. При очевидной полезности идеи оптимизации практика требует осторожного обращения с ней, особенно по мере усложнения задач [23].

Оптимальное решение часто оказывается «хрупким», незначительные изменения в условиях задачи могут привести к выбору существенно отличающихся альтернатив. Оптимизация всегда опирается на предположение, что участвующие в задаче критерии отображают поставленную цель. Даже если это и так, то обычно рассматриваемая система —

часть некоторой большей системы, и тогда локальная оптимизация совсем не обязательно приведет к тому же результату, который потребуется от подсистемы при оптимизации системы в целом. Для этого необходимо увязывание критериев подсистем с критериями системы, что делает ненужной локальную оптимизацию.

Таблица 1.12 — Этапы творческой деятельности ЛПП по решению задач и их отображение классами методов ГиИС

Этапы творческой деятельности ЛПП	Содержание этапа (процедуры)	ТПР	ИСМ	Символьные методы ИИ		Адаптивные методы ИИ	
				ЭС	НС	ИНС	ГА
Выработка (генерация) плана действий (альтернатив)	Абстрагирование					+	
	Дедукция			+	+		
	Индукция			+	+		
	Обобщение					+	
	Обучение					+	+
	Объяснения			+			
	Поиск			+			+
Предсказание результатов действий	Пополнение			+	+		
	Дедукция			+	+		
	Имитация		+				+
	Испытания		+				
	Проверка		+				
Выбор альтернативы (принятие решения)	Тестирование		+				
	Выбор критерия	+					
	Доминирование	+					
	Предпочтения	+					
	Полезность	+					
	Сравнение	+					
Анализ и оценка результатов решения задачи	Упорядочение	+				+	+
	Запоминание					+	
	Идентификация						
	Измерение		+				+
	Классификация					+	
	Кластеризация					+	
	Ранжирование	+	+				
	Распознавание					+	
Сопоставление	+				+		
Сумма	25	8	6	6	4	9	5

Обозначения: ТПР — теория принятия решений; ИСМ — имитационное статистическое моделирование; ЭС — экспертные системы; НС — нечеткие системы; ИНС — искусственные нейронные сети; ГА — генетические алгоритмы; + — метод релевантен процедуре

Использование ГиИС снимает рассмотренную проблематику, поскольку оптимизация может быть применена локально для решения подзадач, согласованных по целям с исходной задачей. Возможно и манипулирование приближенными критериями. Хрупкость оптимальных решений теперь естественно и легко преодолима за счет коррекции аналитических знаний адаптивными методами.

Аргументация с позиций психологии и физиологии. Человеческое мышление характеризуется функциональным взаимодействием сенсорной и языковой систем [85]. Поэтому интеграция преодолевает односторонность взглядов символизма, коннекционизма и других школ на познание и мышление. Объединение высокоуровневой, естественно языковой парадигмы (символизма) с низкоуровневой парадигмой восприятия и первичной обработки информации (коннекционизмом) — стартовая площадка для новых моделей понимания интеллекта и мышления человека. В. Донской отмечает, что когнитивная организация человеческого мозга, базирующаяся на взаимодействии левого и правого полушария, — вдохновляющий пример гибридной естественной системы, несмотря на то, что она очень сложна и еще слабо исследована [86].

Аргументация с позиций лингвистики. Анализ ЯПД в исследованных предметных областях [1] показывает их явную неоднородность. Даже в небольших фрагментах таких языков одновременно присутствуют формулы, таблицы, рисунки, четкие и нечеткие лингвистические переменные, статистические данные, гиперссылки — именно так в интегрированной форме специалисты выражают свои знания.

Аргументация с позиции опыта разработки и использования гибридов. Результаты экспериментов с ГиИС [73] также говорят в их пользу. Так, например, гибрид «дерево решений — многослойный персептрон» для медицинского приложения достиг оценки точности — 99%, в то время как искусственная нейросеть только — 89,3. Для другого гибрида использование символической препроцессии ускорило процесс схождения обучения нейросети в четыре раза. При этом достигнут коэффициент точности 96,4 % для обучающей и 66 для тестирующей выборки, в то время как нейросеть имела только 77,7 и 55 % соответственно. Гибрид с нейросетевой препроцессией для медицинского приложения имел точность 97,5, а экспертная система — 74,5 %. При этом сроки разработки сократились с трех до двух месяцев, однако время решения задачи возросло с 0,5 до 1,7 с. В [87] даны результаты сравнения автономного ГА, метода наискорейшего спуска и их гибрида на задаче однокритериальной оптимизации. Наискорейший спуск отыскал максимум за 663 итерации, ГА — вообще не нашел, а гибрид справился с задачей за 83 — 93 итерации. Результаты [88] говорят о том, что прогнозы урожая пшеницы и ячменя в био-производственной системе, полученные на гибридах, в три — четыре раза точнее прогнозов, выработанных продукционной ЭС.

2. РЕШЕНИЕ СЛОЖНЫХ ЗАДАЧ В СИСТЕМАХ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

2.1. Понятие системы поддержки принятия решений

Существует много задач, с которыми достаточно хорошо справится один человек. Но не меньше проблем эффективно можно решить, только работая в группе. В каких же случаях целесообразно решать задачи коллективно?

1. Когда задача затрагивает интересы многих людей, при этом каждый из них имеет свой взгляд на проблему. В этом случае желательно учесть все точки зрения, прежде чем прийти к какому-либо решению.
2. Если задача может иметь несколько верных решений, то правильный подход к решению проблемы должен учитывать взгляды разных людей.
3. Когда для дела важнее не само решение, а его согласованность между участниками.

Описанные выше ситуации все чаще встречаются в практике деятельности предприятий. Поэтому коллективное решение задач — стратегический фактор совершенствования деятельности систем управления.

Простейшая форма организации коллективов людей — широко распространенные на практике рабочие совещания, построенные по принципу «круглого стола».

Цель этих совещаний — выявление проблем (противоречий) и решение сложных задач. Амбиции и организационные иерархии с отношениями подчиненности здесь отходят на второй план, уступая место знаниям и опыту ЛПР и экспертов. Чем глубже дискуссия за «круглым столом», тем более остро обнажаются противоречия и тем более продуманными и согласованными будут принимаемые решения. Можно утверждать, что на время проведения любого совещания в компании как бы организуется новое виртуальное структурное подразделение — временный творческий коллектив специалистов. Высшая форма групповой коллективной работы — самоорганизация. Самоорганизация — основа интенсивного развития компании, ее способность чутко реагировать на изменения во внешней среде, обоснованно и своевременно корректируя не только свое внешнее поведение, но и основополагающие принципы собственного устройства и функционирования.

Множество открытий и изобретений делается на стыке различных направлений деятельности, если обеспечены условия для общения специалистов, пусть даже на уровне «курилки». Известны примеры, когда компании планируют свои производственные здания так, чтобы сотруд-

ники различных подразделений чаще могли бы, даже случайно, встречаться. Таким образом, деятельность современной компании (или группы компаний) в каждый момент времени можно представить конфигурацией «круглых столов» (рабочих групп), относительно постоянно действующих или создаваемых на самое короткое время.

Круглые столы известны давно и могут проводиться в форме специально организованных мероприятий, которые на профессиональном языке менеджера обозначаются словами и словосочетаниями: «планерка», «пятиминутка», «селекторное совещание», «диспетчерское совещание», «ученый совет», «военный совет», «жюри», «совет главного конструктора», «коллегия министерства», «заседание правительства», «заседание государственной думы» и другими подобными терминами. Несмотря на то, что эти мероприятия проводятся на различных уровнях иерархии управления страной, отличаются масштабностью и значимостью вырабатываемых и принимаемых решений, объемами данных, учитываемых в процессе мероприятий, а также применяемых для этого знаний и опыта всех, их объединяет нечто общее.

Во-первых, подобные мероприятия — типичная горизонтальная структура для коллективных решений по управлению, создаваемая в составе или над одной или несколькими бюрократическими вертикальными организациями. Подобная структура может работать на постоянной или временной основе.

Во-вторых, структура включает два основных типа элементов — ЛПР и экспертов.

В-третьих, несмотря на то, что в ходе мероприятия могут решаться и текущие задачи контроля, анализа, нормирования, прогнозирования, основное предназначение таких структур — решение сложных задач, регулярно и неожиданно возникающих в системе управления.

Участники коллективного решения задач. На рис. 2.1 изображена подобная структура для принятия решений, организаторы и участники которой — люди, играющие различные роли [21, 39]. Человек, руководящий мероприятием и выбирающий наилучшее решение сложной задачи, называется лицом, принимающим решения. Обычно в системах управления существует еще одно лицо — владелец проблемы, которое должно решать задачу в силу служебных обязанностей или морально-этических соображений и отвечает за качество решений. Отношения владельца с ЛПР порождают на практике многовариантность систем управления для коллективного принятия решений. В частности, известны ситуации, когда ЛПР и владелец проблемы — одно и то же лицо. Могут быть случаи, когда в ходе принятия решений ЛПР и владелец проблемы преследуют одну цель или разные цели и должны идти на компромиссы. Могут быть и еще более сложные варианты, когда владе-

лец проблемы решает только часть задачи из вышестоящей системы управления или когда он — эксперт другого коллективного мероприятия. В дальнейшем остановимся на варианте, когда ЛПР и владелец проблемы одно и то же лицо.



Рис. 2.1 — Состав и структура мероприятий для коллективного принятия решений

ЛПР присущ индивидуальный стиль принятия решений. Личность, как капля в море отражается в собственном решении. В этом смысле определённый интерес представляет личностный профиль управленческого решения, т.е. та совокупность индивидуальных особенностей руководителя, которую несут с собой решения к исполнителям. В науке выделены

следующие разновидности личностных профилей решений.

Импульсивные решения — характерны для людей, у которых процесс построения гипотез резко преобладает над действиями по их проверке и уточнению. Такой человек относительно легко генерирует идеи, но мало заботится об их оценке. Это приводит к тому, что процесс принятия коллективных решений проходит скачкообразно, без этапов обоснования и проверки. В практической работе импульсивность решений может привести к тому, что ЛПР будет стремиться внедрить в жизнь решения, которые недостаточно осмысленны и обоснованны.

Инертные решения — результат неуверенного и осторожного поиска. В таких случаях после проявления исходной гипотезы её уточнение идёт крайне медленно. Оценки при этом сверхкритичны, каждый свой шаг человек проверяет неоднократно. Это ведёт к растягиванию во времени процесса принятия коллективных решений.

Рискованные решения — похожи на импульсивные, но отличаются индивидуальной тактикой. Если импульсивные решения перескакивают через этап обоснования гипотез, то рискованные его не обходят, но к оценке человек приходит лишь после обнаружения несообразности. Таким образом, хотя и с опозданием, построение гипотез и их проверка уравниваются.

Осторожные решения — отличает особая тщательность оценки гипотез и критичность. ЛПР совершает множество разнообразных подготовительных действий, прежде чем прийти к выводу. Осторожные решения упреждает оценка. Осторожные люди чувствительны к отрицательным последствиям своих действий. Они больше боятся ошибок, чем их радуют успехи. Поэтому тактика осторожных — избегать ошибок, в

то время как импульсивные люди больше ориентируются на успех и менее чувствительны к неудачам.

На принятие коллективных решений помимо опыта, знаний и личностных особенностей ЛПР, оказывают влияние позиции активных групп - объединений людей с общими интересами по отношению к решаемой задаче. В качестве активной группы могут выступать государственные, общественные организации, конкуренты и другие, старающиеся оказать влияние на результат решения задачи. Дальновидное ЛПР всегда учитывает позиции и критерии выбора активных групп в процессе принятия решений. По своему усмотрению ЛПР может создавать и рабочие группы с разными целями, например, для подбора экспертов.

Важную роль в процессе коллективного принятия решений играют эксперты, профессионально лучше, чем ЛПР, знающие отдельные аспекты задачи, обладающие определенным объемом специальных знаний и опыта, а также данными, необходимыми для выработки и принятия решения. Эксперты могут выяснять причины возникшей проблемы, предлагать и оценивать варианты ее решения, делать прогнозы развития событий в случае принятия того или иного решения. Они отвечают за качество своих рекомендаций. При этом эксперты высказывают свое личное, субъективное мнение, что помогает ЛПР принимать окончательное решение. Мнения экспертов и ЛПР могут не совпадать.

В сложных ситуациях для организации принятия коллективных решений участвуют аналитики-консультанты. Они выполняют следующие основные функции [39]: 1) помогают ЛПР и владельцу проблемы понять задачу; 2) выявляют предпочтения ЛПР и помогают преодолеть несовершенство системы обработки информации у человека; 3) организуют работу с экспертами; 4) выявляют роли и позиции активных групп; 5) разрабатывают и применяют методы принятия решений.

Аналитик, в отличие от эксперта, может и не располагать профессиональными знаниями по решаемой задаче. Поэтому он не дает личных оценок, а помогает ЛПР понять свои предпочтения, оценить варианты, объединить экспертные мнения, сведя их к разумному компромиссу, или занять позицию отдельного эксперта, которая по его (ЛПР) мнению близка или полностью соответствует решению проблемы владельца.

Одна из центральных задач аналитика — оказание помощи в понимании задачи. Поскольку, как было показано выше, в разд. 1.4, сложные задачи, возникающие на практике, — это задачи-системы, у ЛПР могут возникать проблемы с углубленным пониманием этих субъективных сущностей, что требует применения методов и средств системного анализа и теории систем. К сожалению, еще раз подчеркнем, что подобные методы наукой еще не выработаны и поэтому ЛПР и аналитик чаще

основываются в своей работе по постановке задачи только на собственном опыте и интуиции.

Еще одна задача аналитика — выявление системы предпочтений ЛПР. Исследования показывают, что ЛПР без поддержки аналитика часто используют упрощённые или противоречивые критерии выбора. Доказанным фактом считается и то, что люди систематически переоценивают шансы редко происходящих событий и недооценивают более вероятные. Им присущи отвращение к потерям и риску, несовершенство памяти («слепота на изменения»), т.е. некоторые люди запоминают информацию, привнесенную уже после события, а не само событие [90]. Объективное несовершенство человеческой системы переработки информации объясняет ошибки и противоречия человека в процессе принятия решений. Все это указывает на актуальность услуг аналитика, располагающего профессиональными знаниями по психологии, теории принятия решений, логике и искусственному интеллекту. Все это указывает и на значение качества работы аналитика.

Окружение ЛПР — это сотрудники организации, от имени которой принимаются решения и имеющие, как правило, общие взгляды, систему ценностей, предпочтения. Являясь частью этой группы, ЛПР не может принимать решения, не прогнозируя своих объяснений качества принятого решения другим членам коллектива. Оно вольно или невольно, но учитывает предпочтения и политику коллектива.

Несмотря на то, что, как следует из вышесказанного, принятие решения путем организации и проведения специальных коллективных мероприятий «опирается» на мнение достаточно широкого круга (эксперты, окружение, активные группы) людей, принято считать [21], что оно относится к принятию решений в малых группах.

Процессы поддержки принятия решений. Модель принятия решений в малых группах изображена на рис. 2.2. В ней ЛПР и владелец проблемы — это одно и то же лицо, а активные группы и окружение отнесены к внешней среде. Аналитик-консультант рассматривается как один из экспертов или входит во внешнюю среду.

Эта модель совпадает с понятием «система поддержки принятия решений». В работе [91] Э.А. Трахтенгерц отмечает, что системы поддержки принятия решений существуют очень давно. Это военные советы, коллегии министерств, всевозможные совещания, аналитические центры, советы главных конструкторов и т. д. Хотя они никогда не назывались системами поддержки принятия решения, но выполняли именно их задачи.

Авторы настоящей работы придерживаются того же мнения, что суть термина «система поддержки принятия решений» раскрывается

через процессы, события и явления коллективного принятия решений, модель которых изображена на рис. 2.2.

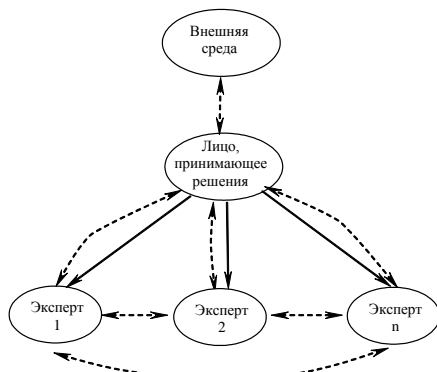


Рис. 2.2 — Модель принятия коллективных решений в малых группах

Такой же подход рассматривается и в работе О.И. Ларичева [21], который вводит понятие «групповых систем поддержки принятия решений» как локальных вычислительных сетей для членов малой группы, а также формальных алгоритмов для сравнения предпочтений на заданном множестве объектов. Аналогично рассуждает и Д. Пауэр [92], используя

понятие «компьютерная система поддержки процесса принятия решений», которое он считает полезным и многофункциональным термином для многих видов информационных систем, поддерживающих процесс принятия решений.

Таким образом, использование термина СППР в практике разработки систем управления и обработки информации в период бурного развития информатики привело к противоречию. С одной стороны, СППР существовали и продолжают существовать без компьютеров, средств вычислительной техники и телекоммуникаций, а с другой — качественная поддержка принятия решений уже и не представляется возможной без вычислительных машин, моделей, алгоритмов и программ. В этой связи два термина «система поддержки принятия решений» и «компьютерная система поддержки принятия решений» в современной литературе практически не различаются.

Использование компьютеров в процессах поддержки принятия решений — тенденция, безусловно, положительная, хотя и не безальтернативная. В частности, альтернатива — усиление возможностей СППР за счет людей с экстрасенсорными способностями. Тем не менее, можно обратить внимание на одну явно негативную сторону безудержной, а иногда и бессмысленной компьютеризации. Она состоит в том, что в последнее время получено мало новых фундаментальных научных результатов исследования процессов, событий и явлений коллективного принятия решений в СППР. Скорее всего, следует ждать таких результатов от психологии, искусственного интеллекта с последующим перенесением в теорию принятия решений и другие дисциплины.

Основываясь на сказанном, в настоящей работе будем различать понятия СППР и КСППР.

Обратимся к рис. 2.2 и рассмотрим особенности процессов, событий и явлений в системах «ЛПР — эксперты», функционирующих в условиях внешней среды.

Что отличает индивидуальное принятие решений от группового в СППР? Для лица, принимающего решения индивидуально, свойственны [93 — 95]: 1) способность принять решение в трудных условиях: за ограниченное время, с высоким риском и т.д.; 2) ограниченная рациональность, когда субъективные пристрастия ограничивают ход мыслей; 3) феномен Ирвина, т.е. завышение значимости и вероятности получения желаемого результата, а для нежелательного — занижение; 4) возможность аналитического паралича, когда усилия по поиску решения долго концентрируются на определенном этапе; 5) ослепление решением, т.е. сдвиг с цели решения на средство ее достижения; 6) феномен излюбленной альтернативы, когда используется метод, с помощью которого ранее уже были достигнуты положительные результаты.

Для группового принятия решений в СППР свойственны [93, 94]: 1) «группсинг», когда у людей в группе деформируется индивидуальное решение и им присуща иллюзия невиновности за некачественное решение; 2) безусловная вера в исповедуемые группой нормы поведения; 3) стереотипный взгляд на участника группы, характеризующийся открытым давлением на тех, кто мыслит в группе индивидуально; 4) феномен интегрального лидерства (проявляется, когда проблемная ситуация сложна и неоднозначна); 5) гипертрофированное доверие к лидеру (все, что говорит начальник, считается единственно верным); 6) ложное согласие (когда отдельным членам группы удобнее занять позицию лидера, даже если есть свое мнение); 7) демонстративное несогласие (несогласие с мнением группы и желание показать себя); 8) виртуальное вмешательство (решение вопроса якобы с подачи некоего лица сверху, при этом заранее учитывается его мнение).

Явление многоаспектного, стратифицированного рассмотрения сложной задачи. Ролевой состав СППР был обсужден выше. Что касается количественного состава, то здесь ясно одно, что лиц, принимающих решения, не может быть два. Остальные вопросы открыты. Это, прежде всего, количество экспертов. Можно ли назвать систему из ЛПР и одного специалиста системой поддержки принятия решений? Скорее всего, только в том случае, если ЛПР умеет «раздваиваться» и будет играть и еще одну роль — эксперта.

Несмотря на «предельность» этого случая здесь затронута одна важная неотъемлемая особенность СППР — явление многоаспектного или в терминах [1] стратифицированного рассмотрения сложной задачи.

Оно характеризуется тем, что изучение сложной системы, процесса, целесообразно выполнять с различных точек зрения, с позиций различных наблюдателей, на различных уровнях абстрагирования. В итоге получается не одна, а несколько относительно простых моделей, реализовать и исследовать которые в целях решения задачи, во-первых, значительно проще, а, во-вторых, качество решений зависит от количества таких выделенных аспектов-страт. Чем больше моделей, тем выше качество вырабатываемых СППР решений. То есть понятие страты связывается с понятием «эксперт». Такое явление многоаспектности или внесения избыточности известно и используется в программировании (функциональная избыточность), математической логике (плюралистическая логика [96]), теории вероятностей и исследовании операций (спор моделей [97]), системном подходе и других науках. Это же явление названо [13] и многоуровневым интеллектом и приводит к необходимости его рассмотрения как иерархии взаимодействующих систем. Основная особенность такой метаинтеллектуальной системы — способность предсказать, организовать и управлять коллективным поведением.

Явление многоаспектности ставит перед ЛПР и ряд задач. Поскольку стратификация, как процесс выбора страт, связана с глубиной понимания ЛПР конкретных действий, выполняемых реальным объектом, для которого решается задача, то всегда возникает вопрос, какую страту считать основной, и существует опасность выбора страт не представляющих интереса. Иными словами, как выбрать страты и сколько означает для ЛПР ответить на вопрос, как выбрать экспертов и сколько их должно быть? С нашей точки зрения, и что на сегодняшний день не вызывает сомнения — привлекаемые эксперты должны представлять различные профессии. Здесь в меньшей степени важна их должность в вертикали власти, а в большей степени важен авторитет в некоторой профессиональной, предметной и проблемной областях. Могут ли несколько экспертов представлять одну профессию? Да, могут, так как плюрализм мнений и спор — безусловно, полезны для СППР. Иными словами, экспертная среда должна быть неоднородной.

Проблема подбора экспертов — наиболее сложная. Здесь важно использовать тех людей, чьи суждения наиболее помогут принятию релевантного решения. Но как выделить, найти, подобрать таких людей? Надо прямо сказать, что сейчас нет методов подбора экспертов, наверняка обеспечивающих успех экспертизы. Мы не будем обсуждать эту проблему, а вместо этого дадим практические рекомендации по процедуре подбора экспертов.

Часто используют методы самооценки и взаимооценки компетентности экспертов. С одной стороны, кто лучше может знать возможности эксперта, чем он сам? Недаром часто можно услышать фразу: «Только я

один знаю, что я могу!»). С другой стороны, при самооценке компетентности скорее оценивается степень самоуверенности эксперта, чем его реальная компетентность. Тем более, что само понятие «компетентность» строго не определено.

При использовании метода взаимооценки, помимо возможности проявления личностных и групповых симпатий и антипатий, играет роль неосведомленность экспертов о возможностях друг друга. В современных условиях достаточно хорошее знакомство с работами и возможностями друг друга может быть лишь у специалистов, много лет работающих совместно. Однако привлечение таких пар специалистов не очень-то целесообразно, поскольку они слишком похожи друг на друга.

Использование формальных показателей подбора (должность, ученые степень и звание, стаж, число публикаций...) носит вспомогательный характер. Успешность участия в предыдущих экспертизах — также хороший критерий для деятельности дегустатора, врача, судьи в спортивных соревнованиях, т.е. таких экспертов, которые участвуют в длинных сериях однотипных коллективных мероприятий. Однако, увы, наиболее интересны и важны не регулярно повторяющиеся, а уникальные экспертизы в сложных задачах, не имеющих аналогов.

В коллективной, совместной работе экспертов большое значение имеют их личностные качества. Один «говорун» может парализовать деятельность всего мероприятия. В подобных случаях важно соблюдение регламента работы СППР. «Молчуна» надо уметь «разговорить».

Есть полезный метод «снежного кома», при котором от каждого специалиста, привлекаемого в качестве эксперта, получают несколько фамилий тех, кто может быть экспертом по рассматриваемой тематике. Очевидно, некоторые из этих фамилий встречались ранее в деятельности СППР, а некоторые — новые. Процесс расширения списка останавливается, когда новые фамилии перестают встречаться. В результате получается достаточно обширный список возможных экспертов. Ясно, что если на первом этапе все эксперты были из одного «клана», то и метод «снежного кома» даст, скорее всего, лиц из этого «клана». Аргументы других «кланов» будут упущены. Самое главное в этом процессе — необходимость предварительного информирования сотрудников о том, что они будут участвовать в работе мероприятия. Не менее важным является подготовленность участников к совместной работе. Ну и, кроме того, опытный руководитель должен дать гарантии, что отобранные люди будут поощрены и заинтересованы в решении задачи.

Необходимо подчеркнуть, что подбор экспертов, в конечном счете, — функция ЛПР или рабочей группы, и никакие методики подбора не снимают с него/нее ответственности. Другими словами, именно на рабочей группе лежит ответственность за компетентность экспертов, за их

принципиальную способность решить поставленную задачу. Важным является требование к ЛПР об утверждении списка экспертов.

Многоязыковые процессы принятия решений в СППР. Из рассмотренного вытекает еще одна важная особенность СППР — многоязыковый характер процессов принятия решений. Каждый эксперт вносит в работу СППР свой профессиональный язык, свой собственный набор терминов, концепций, принципов, знаний. Обратим внимание и на то, что аналитик-консультант также имеет свой профессиональный язык.

Известно [98], что «тексты пишут для посвященных». Для понимания профессионального языка нужна подготовка, как для ЛПР, так и для других экспертов. Как организовать мероприятие коллективного принятия решений в малой группе, чтобы, с одной стороны, на как можно большем количестве языков шло принятие решений, а с другой стороны — преодолеть коммуникационный, языковой барьер?

Проблематика многоязыковых управленческих процессов лежит в сфере одного из принципов системно-организационного подхода в искусственном интеллекте В.Б. Тарасова [13], а именно «Принципа многомерности интеллекта и дополнительности различных моделей и методов описания интеллектуальных процессов и систем». Он, в частности, утверждает, что никакое сколько-нибудь сложное явление, а тем более интеллектуальное поведение, нельзя описать средствами одного-единственного языка. Как правило, для этого требуются набор языков различной жесткости [99], сочетание различных логик анализа и множество интерпретаций интеллектуальной системы. Тогда может быть обеспечена «голографическая картина» системы, связанная с использованием различных точек зрения.

В этой связи актуален вопрос о возможной замене эксперта. Может ли эксперт быть заменен ЛПР в ходе организации СППР на эксперта другой профессии? Ответ отрицательный. Объясняется это тем, что в ходе многоаспектного рассмотрения задачи ЛПР не может понять (получить, вывести) принципы и законы, характеризующие рассматриваемую задачу экспертом некоторой профессии, на основании аналогичных принципов и законов, но эксперта с другой профессией.

Итак, ЛПР имеет дело с неоднородной, многоязычной средой, что позволяет ему организовывать и вести несколько линий рассуждений при решении сложной задачи.

Несмотря на важность исследований качественных особенностей ЛПР и экспертов, безусловно, то, что специфику СППР определяет их взаимодействие в ходе коллективного принятия решений. На рис. 2.2 это взаимодействие отображено двумя видами связей: сплошными и пунктирными стрелками.

Процессы самоорганизации и формирования СППР. Организационная структура СППР, как автономной междисциплинарной рабочей группы, формируется как следствие установления между участниками мероприятия, а также ЛПР и экспертами отношений кооперации и сотрудничества. Они показаны на рис. 2.2 сплошными стрелками. Можно ли считать, что здесь уместны и отношения типа «руководитель — подчиненный», т.е. отношения подчиненности? Если СППР самоорганизуется, как горизонтальная структура в пределах организации, то в некоторых случаях так можно считать, однако значительно чаще бывает, когда эксперты по служебным обязанностям либо вообще, либо непосредственно не подчинены ЛПР.

Организация сотрудничества может идти несколькими путями. Во-первых, ЛПР может пригласить эксперта не непосредственно (напрямую), а через его руководителя. Во-вторых, ЛПР может знать эксперта и договориться о его участии непосредственно. В-третьих, ЛПР может и не быть знакомым с экспертом, обратиться к нему и договориться об участии. Участие и работа эксперта могут оплачиваться либо может использоваться иная система стимулирования его труда. В этой связи представляется, что кооперация и сотрудничество носят «договорной» характер на время решения сложной задачи. Представляется, что именно эти отношения влияют на качество работы экспертов, именно они «работают» на авторитет и положение эксперта в обществе.

Процессы взаимодействия экспертов в СППР. Пунктирные стрелки на рис. 2.2, связывающие экспертов, показывают их взаимодействие. Это взаимодействие может быть многогранно.

Во-первых, какая-то часть экспертов может подчиняться одному или нескольким другим экспертам по службе, т.е. могут существовать отношения подчиненности. Во-вторых, могут существовать группы экспертов (по профессиям, по целям, по средствам решения задачи и т.п.), не связанных отношениями подчиненности, и группы могут иметь явных или неявных лидеров, что еще в большей степени «обостряет» неоднородность процесса коллективного принятия решений. При этом возникают процессы самоорганизации, направляемые отношениями кооперации, компромисса, содействия, конкуренции, конформизма (подобия), приспособления, солидарности, уклонения и др.

Кроме сказанного, взаимодействие экспертов в ходе решения задачи должно рассматриваться как процессы обмена: данными, знаниями, объяснениями и частичными решениями общей задачи.

Еще один вид процессов в СППР «запускают» отношения порядка решения задачи, устанавливаемые ЛПР. Обычно при этом оно произносит: «Сначала послушаем эксперта №1, затем экспертов №2 и №3, а потом опять эксперта №1». Очевидно, что такие отношения устанавли-

вают регламент работы СППР. Вариантов регламентов может быть достаточно много. Например, вариант 1 — «Все эксперты думают и высказываются независимо друг от друга по установленной ЛПР очереди», вариант 2 — «Эксперт №1 решает одну часть задачи и передает результат эксперту №2...» и др.

Обобщая сказанное о взаимодействии экспертов, выдвинем предположение о том, что это взаимодействие идет в соответствии с тем, как ЛПР редуцировало сложную задачу и какая в результате этого получилась декомпозиция. Именно граф-сеть такой декомпозиции и определяет систему взаимодействия экспертов.

Процессы взаимодействия ЛПР и экспертов в ходе решения сложных задач. Следующая отличительная особенность процессов, событий и явлений в СППР следует из специфики взаимодействия ЛПР и экспертов. Это взаимодействие показано на рис. 2.2 пунктирными стрелками. Известно [12, 21, 22], что процессы коллективного принятия решений в СППР могут быть сведены к процессам диагностики задачи, синтеза метода ее решения, получения результатов решения задачи, согласования мнений, прогнозирования последствий результатов решения задачи, выбора окончательного решения, реализации решений, мониторинга и оценки результатов (см. рис. 2.3, 2.4).

Анализ литературы показывает, что к настоящему времени часть этих процессов хорошо изучена в теории принятия решений [21,80], однако процессы диагностики и понимания задачи, синтеза метода ее решения не формализованы, исследованы недостаточно и чаще всего относятся к предметной области системного анализа [23,48].

В [1] были предложены подход и концептуальная модель процесса понимания сложной задачи. Он оправдал себя при решении нескольких задач планирования в транспортных, био-производственных системах и системах проектирования. Суть подхода может быть сформулирована в виде следующей гипотезы № 1: «Для понимания сложной задачи она должна быть редуцирована, построены ее декомпозиции, специфицированы подзадачи и построена их координация».

По этой гипотезе в СППР актуальны процессы взаимодействия ЛПР и экспертов, специфицируемые отношениями субординации, определения и соответствия. Результатом этих процессов является: 1) членение, раздробление (редуцирование) сложной задачи на составные части— подзадачи. При этом представляется, что ведущую роль в этом играет ЛПР. В итоге редуцирования строится декомпозиция (одна или несколько), представляющая собой взаимосвязанную совокупность подзадач, которая и является «прообразом», отображаемым на множество экспертов; 2) установление соответствия на множествах подзадач и экспертов. Иными словами, за каждой из подзадач «закрепляется» один

или несколько экспертов, что обеспечивает плюрализм мнений и «спор моделей» в ходе профессионального решения подзадач; 3) спецификация (постановка) подзадач выполняется в результате сбора и обработки информации по их входам (дано), выходам (найти), целям и привлекаемым методам, опыту или знаниям и является частью деятельности экспертов. Таким образом, отношения ЛПР и экспертов выражаются их взаимодействием при редуцировании, спецификации и установлении соответствия.

В [1] предложены подход и концептуальная модель синтеза метода решения сложной задачи в СППР. Суть подхода может быть сформулирована в виде гипотезы № 2: «Нет, и не может быть одного единственного метода, используемого СППР для решения сложных задач. Вместо этого СППР располагает набором инструментальных средств, над которыми такой метод может быть сконструирован». В соответствии с этой гипотезой в СППР актуальны процессы взаимодействия ЛПР и экспертов, определяемые отношениями интеграции (комбинирования) и координации.

Эта гипотеза исходит из смены парадигмы моделирования интеллекта человека. Особое значение придается таким особенностям, как распределенность (децентрализованность) интеллекта. В этой связи В.Б. Тарасов отмечает, что распределенность интеллекта — фундаментальное свойство живых систем и для адаптации к сложной, плохо определенной, динамической среде формируются сообщества живых систем: группы людей, стада животных, стаи птиц и пр. При этом каждая особь располагает фрагментарной информацией о внешней среде, а совместно они синтезируют адекватную интегральную модель внешней среды и прогноз процесса взаимной адаптации со средой. Отсюда разработаны системно-организационный подход в ИИ и новая синергетическая парадигма, смещающая акцент исследования на группу, сообщество из неоднородных, взаимодействующих объектов [13].

Одна из целей группового взаимодействия — выработка (синтез) метода сложной задачи. Как отмечалось выше, сложные задачи имеют варьирующиеся во времени нефиксированные предметную и проблемную области, т.е. изменяющиеся данные и знания для решения задачи, а также состав подмножеств задач-элементов. Из этого следует, что нельзя вести речь о методе, один раз и навсегда разработанном и применяемом для решения сложной практической задачи. Метод, выработанный СППР для решения сложной задачи в данный момент времени, в другой момент времени уже может быть не актуальным. Значит, в СППР существует множество методов, нарабатываемых за некоторый период времени решения задачи.

Такой метод синтезируется в ходе проведения мероприятия СППР из имеющегося в распоряжении экспертов арсенала методов, существует только во время коллективной работы и на следующем мероприятии СППР, даже для решения одной и той же задачи, будет изменен (т.е. синтезирован заново) в силу изменений в самой задаче, СППР и внешней среде.

Наконец, решение подзадач экспертами в СППР координируется. Этот процесс обусловлен, прежде всего, тем, что, как показали исследования [1] феномена СППР, чтобы декомпозиция была релевантна сложности исходной (редуцированной) практической задачи, она должна быть дополнена процедурой координации. Так как «носителем» образа сложной задачи во время работы СППР можно считать ЛПР, а «носителями» образов подзадач — экспертов, то образ задачи-системы нельзя отрывать от образов подзадач в ходе их совместного решения. Поскольку решение задач всегда идет в условиях целеполагания и ограниченных ресурсов, в том числе и временных, ЛПР должно уметь координировать цели и ресурсы, путем распределения и перераспределения их между решаемыми экспертами подзадачами.

Таким образом, отношения редуцирования, спецификации и установления соответствия в СППР могут быть дополнены отношениями интеграции, комбинирования мнений экспертов для окончательного решения задачи и координации. Именно эти отношения и позволяют выразить суть термина «поддержка» принятия решений, определяют интегративное свойство (эмерджентность) СППР и позволяют дать следующее определение: «Система поддержки принятия решений — система «лицо, принимающее решения — эксперты», предназначенная для анализа сложных задач, синтеза и реализации метода их решения».

Плюсы и минусы решения сложных задач в СППР. Сформулируем потенциальные плюсы и минусы решения задач в СППР. К преимуществам отнесем: 1) рассмотрение проблем группой позволяет шире взглянуть на проблему и провести тщательный ее анализ; 2) в ходе работы группы участники демонстрируют больше знаний, находят больше доводов и выдвигают больше вариантов решения; 3) в проведении дискуссии уточняется постановка проблем и уменьшается неопределенность в отношении возможных вариантов действий; 4) участие в принятии решений способствует удовлетворению работников и стимулирует активность в их претворении; 5) овладение участниками группы навыками совместной работы; 6) развитие творческого мышления у сотрудников; 7) более высокий уровень согласования конечного решения.

К недостаткам отнесем: 1) процесс группового обсуждения занимает много времени; 2) компромиссные решения могут не удовлетворить никого (выбранный вариант выгоден не организации, а самим участни-

кам процесса); 3) в процессе совместной деятельности участники мероприятия могут стать настолько лояльными ей, что групповые нормы начинают препятствовать высказыванию различий во взглядах и разнообразию мнений; 4) когда решение принимается коллективно, не всегда ясно, кто конкретно несет ответственность за принятое решение.

Системы поддержки принятия решений и системы поддержки переговоров. Вступление в переговоры объясняется потребностью в совместных действиях. Они могут потребоваться из-за смены обстановки, чрезвычайного происшествия, падения спроса на рынке, проигрыша выборов и т.д. Осознав возможную цель совместных действий, проанализировав ситуацию и выработав свою позицию, ЛПР формирует и передает свои предложения потенциальным партнерам. Последние, получив предложения и оценив их как полезные для себя, вырабатывают свои линии поведения. После этого начинаются собственно переговоры.

Традиционно модель переговоров хорошо структурирована и основывается на поиске консенсуса. Здесь используются известные алгоритмы согласования решений.

Переговоры конфликтующих сторон по наиболее сложным проблемам протекают иначе — в виде уникальной последовательности интенсивных, эволюционирующих переговорных процессов, шагов преодоления существенных разногласий. На каждом из таких шагов могут использоваться свои методы и схемы рассуждений экспертов. Тогда общая модель переговоров будет носить комбинированный, интегрированный характер.

Сложность задач ведения переговоров, длительность и напряженность процессов взаимодействия партнеров и определили появление систем поддержки переговоров (СПП) (англ. negotiation support systems) или систем поддержки групповых решений (СППР) (англ. group decision support systems). Они по идеологии достаточно близки к СППР, однако имеют особенности, связанные с необходимостью добиваться взаимопонимания между участниками коллективного принятия решений.

Взаимодействие СППР и СПП может быть различным: первая может использоваться в переговорах или СПП может включаться в состав СППР для выработки согласованного мнения участников мероприятия.

Широкий спектр вопросов, моделей и результатов, возникающих при этом, рассмотрены Э.А. Трахтенгерц в работе [100].

2.2. Особенности работы экспертов в системах поддержки принятия решений

Работа экспертов в СППР зачастую похожа на бег по кругу вместо поступательного движения по спирали. Типичная ошибка заключается в

том, что коллективная деятельность СППР рассматривается как естественный, не требующий организации и заботы процесс. Иными словами, предполагается, что все люди способны работать в группе. На самом деле это не так. Есть много примеров, когда, работая в СППР, талантливые люди не смогли в полной мере использовать свои возможности, и работа таких коллективов была неудовлетворительной.

Почему возникают ситуации, которые участники СППР предпочли бы избежать? Они складываются из-за того, что участники СППР не понимают и недостаточно учитывают факторы, влияющие на их совместную работу: 1) направление усилий на решение одной задачи; 2) «зацикливание» экспертов на своей задаче или решении; 3) неспособность ЛПР и экспертов видеть вокруг себя ничего, не имеющего прямого отношения к рассматриваемой им проблеме; 4) поле групповой деятельности достаточно обширно, и не каждый человек может быть привлечен как эксперт в любом вопросе [93 — 95].

Эти факты указывают на то, что участники СППР должны быть хорошо организованной и сплоченной командой, преследующей определенные цели. Команда строится на следующих основных принципах: 1) участники СППР объединены общими намерениями, задачами и целями; 2) участники СППР взаимозависимы, т.е. они нуждаются друг в друге для достижения цели; 3) участники СППР понимают, что для достижения цели они должны эффективно работать вместе.

Стереотипы мышления в СППР. Эксперты склонны к аналитическому и индивидуальному мышлению. Если они работают в одиночку, то пытаются решить задачи логическими рассуждениями. Однако существует случаи, когда это малоэффективно и бывает важно взглянуть на задачу под другим углом зрения. Этому способствуют склонность экспертов к фантазиям, творческое отношение к делу, стремление предлагать не одно, а несколько вариантов решений задачи, если ее условия не содержат строгих ограничений.

Способностью к творческому мышлению обладают не все эксперты. Индивидуальные решения всегда зависят от способностей, взглядов, угла зрения конкретного человека. А решение задач коллективно идет с учетом различных точек зрения, различных аспектов. Поэтому считается, что творческое мышление носит коллективный характер. Новая трактовка того или иного события побуждает экспертов мыслить активно. Именно уход от привычных и формирование новых взглядов на задачу дает толчок для выработки новых идей.

Личностные барьеры творческого мышления. Творческую активность в СППР сдерживают несколько ограничений личностного характера [93, 95, 101].

Самоналагаемый барьер (самостоп) состоит в том, что при коллективном рассмотрении задачи эксперты невольно думают о дополнительных условиях, которые ограничивают ее решение, хотя ни одно из этих ограничений не было упомянуто ЛПР в условиях. Если ограничения отбросить — решение может быть легко найдено. Примерно 95% идей, рождающихся в голове, человек отвергает сам в момент их возникновения, налагая на себя ограничения.

Ограничение «Попытка найти один — единственный правильный ответ» — это неотъемлемая и почти непреодолимая черта экспертов с аналитическим складом ума. Им присущи логика, поиск единственного решения и рассмотрение нескольких возможностей. Экспертам с творческим складом ума присуще воображение, и они обычно ищут несколько вариантов ответа. Человеку кажется, что все думают так, как он, и выбирает именно это решение, хотя, возможно, ему видится другое. Это происходит из-за того, что люди склонны пассивно группировать в уме одинаковые объекты.

Оглядка на мнения других участников СППР — барьер, связывающий творческую активность экспертов. Это зависит от нескольких обстоятельств. Например, молодой сотрудник или приспособленец будут предлагать решения, которые нравились бы начальству. В процентном соотношении среди работающего населения гораздо меньше людей, способных отстаивать свою точку зрения, независимую от мнения ЛПР.

Отсутствие попыток изменения очевидного — барьер, состоящий в том, что если ЛПР уже заранее имеет решение задачи, то оно проявляет склонность избегать других вариантов решения. Решение любой проблемы человеком, как правило, состоит из двух этапов. Сначала, решая задачу, он стремится к ясному и определенному ответу, который был бы принят без всяких вопросов и возражений. Затем, уже имея ответ, на втором этапе, чаще всего стараются избежать его изменений, хотя в дальнейших рассуждениях могут появиться лучшие решения. Более того, при попытке оспорить его решение или предложить новый вариант ЛПР, скорее всего, даст отрицательный ответ или назовет причины невозможности выполнения иного решения. Высшая степень отсутствия попытки изменения очевидного — автоматическое «нет». Если найденное решение не считается единственно правильным, то лучше отложить его реализацию и проверить правильность этого решения.

Слишком быстрая оценка в недалеком будущем — барьер, возникающий из того, что многие руководители, скептики, пессимисты по натуре не любят рассуждать над решением проблемы и склонны к быстрым оценкам событий. Они считают оценки экспертов нерациональными: «Это глупо», «Это не будет работать», «Мы пробовали делать это раньше, но это не работало». Такие мнения ЛПР «гасят» идеи.

Не надо быстро принимать решение. Быстрая оценка убивает творческое начало в экспертах. Поэтому в СППР важны рассуждения и критика.

Страх выглядеть глупо — самый трудный барьер для преодоления. Он вызван комплексом неполноценности, возникающим еще в раннем возрасте, когда дети начинают смеяться над кем-нибудь из своей компании за высказанную идею или мысль. Затем это усиливается в более позднем возрасте: школе, университете. Этот барьер на высказывание собственного мнения почти непреодолим при совместной работе экспертов в СППР. Другие причины, также сдерживающие экспертов от высказывания своей точки зрения: 1) младшие по возрасту и положению в присутствии старших не предлагают «дикие» идеи из-за боязни выглядеть глупо; 2) у кого-то срывает опасение разрушить карьеру; 3) сотрудники предпенсионного возраста часто боятся высказывать идеи в присутствии младших (не хотят казаться старыми, «выжившими» из ума); 4) эксперты-руководители по должности не любят выступать против общепринятых точек зрения, в частности, высказанных известными людьми, из-за риска быть неправильно понятыми. Лишь немногие (5 — 10%) в среднестатистическом трудовом коллективе способны отстаивать свою точку зрения при любых обстоятельствах.

Организационные барьеры творческого мышления. Кроме внутренних, личностных ограничений, творческий человек, работая в СППР, сталкивается с рядом организационных барьеров [101].

Один из них — недостаточный период времени обсуждения или созревания идеи — объясняется тем, что на многих совещаниях в условиях оперативного решения вопросов высказанная экспертами информация не рассматривается как проблема и решения принимаются без обстоятельного их обсуждения. Иначе говоря, текучка оказывает существенное давление на уровень понимания задачи и принятие решения.

Стиль управления, стабильность среды, в которой принимаются решения, также могут стать барьером, оказать влияние на творчество в СППР. В устойчивых организациях нет потребности в принятии творческих решений. В них, как правило, работники сталкиваются с одними и теми же задачами: в технологии, реализации и т. д. Эксперты уже не решали эти проблемы, а новые, творческие решения не всегда одобряются из-за нежелания иметь лишние хлопоты. Стиль управления должен стимулировать творчество.

2.3. Режимы и организация работы систем поддержки принятия решений

Системы поддержки принятия решений могут функционировать в различных режимах [102, 103]: 1) режиме проблемного мониторинга, предполагающего непрерывное наблюдение ЛПР и экспертов за поступающей из внешней среды информацией; 2) режиме планового обсуждения, как запланированного и регламентированного обсуждения сложных задач экспертами и в случае необходимости принятие решений ЛПР обычно в виде планов, регулирующих или нормирующих воздействия на объект управления; 3) режиме чрезвычайных, непредвиденных, кризисных ситуаций с приглашением экспертов, оперативным обсуждением сложной задачи, принятием решений ЛПР, выработкой действий и контролем их исполнения.

Режим проблемного мониторинга. Проблемный мониторинг (анализ) используется для наглядного постоянного слежения за ситуацией в объекте управления и внешней среде и информирования руководителей о ее развитии, а при необходимости — акцентирования их внимания на актуальных истораживающих процессах и событиях. Мониторинговый режим регламентируется заранее определенной, но постоянно изменяющейся тематикой и относительно постоянным набором источников информации. Мониторинг может вестись во времени или на множестве наблюдаемых объектов.

Обычно организация мониторинга сложностей не составляет: ограничиваются внутренняя и внешняя сферы, выделяются благоприятные и негативные факторы, определяются источники информации и начинается слежение за ситуацией. Труднее «мониторить» в новых ситуациях, когда изначально не понятно, что делать, с чего начать. В этом случае приглашаются эксперты и специальными методами (см. разд. 6) их деятельность направляется на углубление понимания, уяснение проблемы, ее структуризацию, выделение определяющих факторов. Это позволяет внести в мониторинг целеполагание и направить его на решение конкретной задачи. При этом устойчивость процесса принятия решений обеспечивается слежением, как за обобщенной, так и первичной информацией. Не рекомендуется также неоправданно быстро изменять совокупность показателей для слежения за ситуацией. Помимо функций отображения и слежения мониторинг нацелен на выявление, формулирование и последующее использование для выработки и принятия решений существенных закономерностей в поведении объекта управления. Для облегчения выявления закономерностей применяются средства визуализации, в частности, средства многомерного отображения, а также методы математической статистики и искусственного интеллекта.

Режим планового обсуждения. Плановое обсуждение — это заслушивание на заседании СППР сообщений и докладов экспертов по регулярно (в соответствии с горизонтом планирования) решаемым сложным задачам. Доклад обсуждается, и затем следует принятие решений.

Для организации планового обсуждения заблаговременно готовится сценарий доклада с прогнозом хода обсуждения. Однако дискуссия может «уйти» от сценария, поддерживаемого информационной технологией и сделанными заранее заготовками. Чтобы повысить «устойчивость управления» по сценарию обсуждения, работа СППР должна быть максимально информационно открыта для сведений и материалов из внешних источников. Однако увеличение открытости может привести к снижению целенаправленности обсуждения задачи в СППР.

Успеху в работе СППР служат предварительные исследования психологических профилей приглашаемых экспертов. Изучение публикаций экспертов, высказываний, их реакций на различные ситуации помогает прогнозировать возможные отклонения коллективного обсуждения задачи от намеченного сценария и выбрать альтернативный сценарий или соответствующую справочную информацию.

Поскольку для планового обсуждения обычно применяется «т-образный» подход к изучению проблемы, т.е. она изучается как «вширь» по всем имеющим отношения с ней другим сложным задачам, так и «вглубь» по всему спектру ее подзадач. В СППР с информационной точки зрения реализуется замена растянутой во времени последовательности событий одномоментным представлением экспертам и ЛПР обобщенной информации. Иными словами, обеспечивается временная компрессия информации, активирующая интуитивное, творческое мышление участников коллективного принятия решений.

Особое внимание в режиме планового обсуждения уделяется иллюстрированному представлению докладов экспертов. Отличительная черта такого режима работы СППР — высококачественное оформление материалов, а также их телекоммуникационное и информационно-справочное обеспечение для получения информации из удаленных и локальных баз данных, а также привлечения к участию удаленных от места проведения СППР экспертов.

Режим чрезвычайных, непредвиденных, кризисных ситуаций. Решение сложных задач в данном режиме отличается от запланированного обсуждения несколькими обстоятельствами: 1) сжатыми сроками работы СППР, когда на выработку и принятие решений динамичное развитие ситуации оставляет мало времени; 2) важность возможных последствий нежелательного развития ситуации в объекте управления; 3) состав экспертов, работающих в данной организации, обычно недостаточен и привлекаются эксперты из других организаций; 4) нет запла-

нированного докладчика или у него не было времени продумать сценарий выступления; 5) существенная новизна возникшей перед СППР задачей и неопределенность в исходных данных; 6) эксперты в ходе обсуждения часто применяют вероятностные оценки и оценки своей уверенности (неуверенности) в правильности своих рекомендаций; 7) акценты смещаются в большей степени не на сбор и анализ данных, перечень необходимых из которых вырабатывается только по ходу работы СППР, а на рассуждения экспертов, владеющих данными, знаниями и опытом, что определяется их профессиональной квалификацией; 8) особая роль личностных портретов ЛПР и экспертов, а также проявление рассмотренных выше психологических факторов, как отрицательно, так и положительно влияющих на результаты работы.

В кризисных ситуациях ЛПР помимо «стеснения» указанными выше обстоятельствами также не может заблаговременно редуцировать сложность возникшей задачи и формулирует ее содержательно, зачастую нечетко, в качественных категориях. При этом предполагается, что каждый из экспертов должен сам для себя построить упрощенный образ задачи, т.е. выделить ее подзадачу и дать рекомендации по ее решению.

Работа СППР организуется «за круглым столом», где все должны всё видеть и слышать. Огромная нагрузка (психологическая, моральная, ответственности) ложится на ЛПР. В условиях ее непрерывного и давящего на рассуждения ЛПР характера оно обязано внимательно следить за множеством разнообразных, экспертных линий рассуждений. Кроме этого, ЛПР должно оценить и сепарировать их, обобщать (интегрировать) несколько результатов в один общий и, что особенно важно, — координировать работу экспертов, предоставляя (отбирая) ресурсы: время, информацию, материальные, энергетические и финансовые средства.

При чрезвычайном режиме работы СППР формирование сценария принятия решений практически совмещается с процессом принятия решения. Нет предварительно построенной модели оценки ситуации, Однако эта модель должна быть построена оперативно из имеющихся «под рукой» признаков, характеристик ситуации. Из-за уникальности ситуации ограничена возможность использования прецедентов, опыта экспертов. Противоречива логика происходящих событий, нет времени и возможности построить строгую логическую линию рассуждений. В рассуждения вкрапляются многозначные оценки и оценки уверенности и неуверенности. Теоретические знания и модели играют только роль сдерживающего безудержную фантазию фактора. Актуальны быстрые, низкоуровневые реакции человека на происходящее, такие эффекты, как «озарение», «интуиция» и «категоризация ситуации».

Конечно, функционирование СППР в одном из рассмотренных режимов — идеальная модель. Обычно они могут самым неожиданным образом переплетаться и комбинироваться. Так, например, работая в режиме мониторинга или планового обсуждения, СППР может перейти в режим чрезвычайной ситуации, или, наоборот, плановое обсуждение может потребовать достаточно длительного мониторинга.

Организация работы систем поддержки принятия решений. Мероприятия СППР осуществлялись и еще будут проходить без каких-либо специальных, дорогостоящих средств, т.е. не автоматизированно. Место проведения заседаний СППР — это не обязательно компьютеризованное помещение. Известны комнаты для «мозгового штурма» (англ. *brine storming room*) с обычным столом, классной доской и мелом. Главное — правильно подобрать информацию и организовать умственную активность специалистов.

Однако по мере развития общества государство, правительство, оборона, финансовые организации, корпорации и группы компаний будут все больше средств выделять на автоматизацию работы СППР в различных режимах, создание наилучших условий для аккумуляции интеллектуальных возможностей участников на решении сложных задач.

Как одно из средств организации работы СППР за последние тридцать лет наибольший и неугасающий интерес вызывают ситуационные комнаты [102]. Они позволяют быстро «погрузить» участников в рассматриваемую проблему, «научить их говорить» на одном языке, помочь разобраться в проблеме, правильно сформулировать запросы к источникам информации и совместно принять рациональное (не обязательно наилучшее) решение.

Такие ситуационные комнаты популярны за рубежом — там они работают во властных структурах, в корпорациях и банках. В России существуют ситуационные центры: Президента РФ, Совета Безопасности, МЧС. Начинают создаваться ситуационные центры в субъектах России.

Ситуационные комнаты — это специальное место заседаний СППР, оперативного построения и «проигрывания» сценариев, быстрой оценки проблемной ситуации на основе использования специальных методов обработки больших объемов информации. Эффект от компьютеризации ситуационных комнат зависит от развитости используемых методов, технологий информатики, структурирования данных и знаний, построения сценариев.

С учетом потребности в реализации вышеизложенных режимов работы СППР и имеющихся информационных, интеллектуальных, материальных ресурсов компоновка ситуационной комнаты может быть осуществлена с различным уровнем сложности.

В наиболее простом виде ситуационной комнатой можно, например, считать кабинет руководителя, в котором проводятся обсуждения проблем и ситуаций: плановых, чрезвычайных и авральных.

Для увеличения эффективности ситуационной комнаты необходима ее ресурсная модернизация. Важнейшее устройство ситуационной комнаты — экран коллективного доступа для аккумуляции образного мышления участников СППР на решении сложной задачи. На экране с использованием средств картографии отображается ранее заготовленная информация.

Еще одна важная составляющая ситуационной комнаты — пользовательские интерфейсы, устройства взаимодействия информационно-моделирующих и технических средств ситуационной комнаты с экспертами и ЛПР. Такими устройствами могут быть алфавитно-цифровые клавиатуры и дисплеи для каждого из участников. Клавиатуры используются участниками, чтобы оперативно реагировать на происходящее, выбирать факторы и вводить оценки.

Для ситуационной комнаты разрабатываются и ведутся соответствующие базы данных и знаний, содержащие информацию по рассматриваемым проблемам. Базы данных пополняются в режиме мониторинга или формируются при подготовке определенного вопроса. Сценарии и планы проведения совещаний СППР также сохраняются в базе данных.

Кроме рассмотренных средств, предусматривается оперативное телекоммуникационное взаимодействие лиц, находящихся в ситуационной комнате, с удаленными экспертами.

Наиболее сложный и дорогостоящий атрибут ситуационной комнаты — это модельное, алгоритмическое, программное и методическое обеспечение. Здесь применяются разнообразные методы, технологии и инструментальные средства из разд. 3, 4.3 и 6.

2.4. Модели процесса принятия коллективных решений

Принятие управленческих решений — важный вид деятельности, осуществляемой в системах управления. В работе [104] подчеркнуто, что из всех управленческих обязанностей руководителей и менеджеров, акт принятия решений не имеет себе равных по своей значимости, и что принятие решений — неотъемлемая черта каждого проявления взаимоотношений людей, машин, материалов и технологий.

Однако до настоящего времени не существует общепринятого понимания того, что представляют собой управленческие решения, кто их принимает и где они осуществляются. Авторы согласны с Гербертом А. Саймоном (Herbert A. Simon), который в своей классической работе о

науке принятия решений определяет управленческое решение как процесс, синонимичный с самим процессом руководства.

Ниже приведены три из множества известных определений управленческого решения [104].

1) Сказать, что ЛПР приняло решение, может означать, что он (она) начал (а) серию действий в пользу чего-либо, или это может означать, что он (или она) решил (а) сделать определенную вещь ...

2) Сделать (принять) решение означает решить, что человек должен делать в определенной ситуации, после рассмотрения им альтернативных возможностей своих действий.

3) Управленческое решение — момент в постоянно текущем процессе оценки различных возможностей для достижения цели. При этом имеющиеся ожидания какого-либо конкретного образа действий, заставляют человека, принимающего решение, выбрать именно данный образ действий, который, по его мнению, предоставляет ему наилучшие возможности для достижения цели.

Трудно недооценить значимость управленческих решений. Решения — это основные операции организаций. Успешные организации, способны «перерешать» конкурентов в трех аспектах: они принимают более правильные, оправданные решения; они принимают решения быстрее; и они более успешно реализуют принятые решения» [104].

Исход управленческого решения — это результаты, ожидаемые руководством (владельцем проблемы) при принятии решения. Исход — основной аспект решения, в нем отражается преобладающее отношение менеджеров к проблеме на момент принятия решения [104]. Управленческие решения осуществляются посредством динамического и внутренне взаимосвязанного процесса, состоящего из некоторого количества этапов (шагов, действий) принятия решений.

Рассмотрим две модели процесса принятия решений: 1) модель О.С. Виханского, А.И. Наумова (рис. 2.3); 2) модель М.В. Самсоновой, В.В. Ефимова (рис. 2.4).

Обратные связи на рис. 2.3 отражают итеративный циклический характер принятия решений. Итерации в процессе принятия решений обусловлены необходимостью уточнения и корректировки данных после выполнения определенных процедур и возврата к работам на любой предшествующий этап.

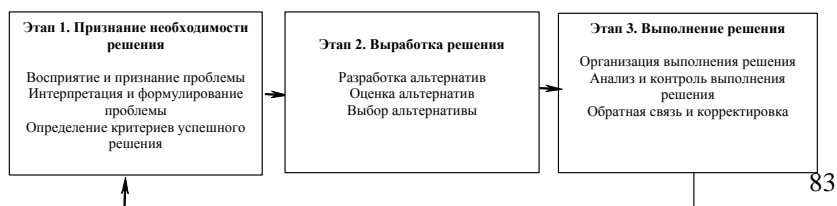


Рис. 2.3 — Модель процесса принятия решений О.С. Виханского, А.И. Наумова

Признание необходимости решения. Если исходить из того, что решение — это реакция на возникшую проблему, то этап изучения направлен на признание или непризнание проблемы. Для структурированной задачи признание проходит довольно прямолинейно, т.е. ее существование в предшествующие периоды времени дает повод утверждать, что она будет существовать и в последующие моменты времени. Если речь идет о сложной задаче, то признание ее существования само по себе — задача. Признание существования задачи — необходимое условие для ее решения, если владелец проблемы не признает ее, то и принятие решений не состоится.

Признанная задача далее интерпретируется и ей придается значимость: рутинная, плановая, кризисная, уникальная и др., см. рис. 1.2.

Формулирование проблемы предполагает поиск ответов на вопросы: 1) что же происходит в действительности?; 2) каковы причины происходящего?; 3) что за этим всем стоит? Очевидно, что острота проблемы, возможности ее эффективного решения зависят от ситуации — сочетания условий и обстоятельств возникновения проблемы.

Определение критериев успешного решения задачи помогает избежать многих ошибок, проявляющихся на последующих этапах. Известны критерии вида «мы должны» и критерии вида «мы хотим». Первый вид критериев приводит к порождению множества альтернатив. Критерии из второй группы заставляют рассматривать все возможные варианты, в том числе и идеальные.

Выработка решения. Состоит из разработки (выработки), оценки и выбора альтернатив. Как только определены факторы, ограничивающие решение, ЛПР может начинать работу по поиску альтернатив. Многие из альтернативных решений легко обнаружить. Они обычно известны из предыдущего опыта, стандартны и легко вписываются в критериальные границы лучшего решения. Однако решение сложных задач не уместится в привычные и стандартные рамки. В этом случае необходим творческий подход, использование специальных приемов активизации творческого мышления человека (см. разд. 3, 6).

Все предложенные альтернативы на следующем этапе сравниваются друг с другом или оцениваются для выбора лучшей из них. Оценка предполагает определение отрицательных и положительных сторон рассматриваемых альтернатив и установление между ними некоего уровня компромисса. В итоге получается общая картина решения: выборы, риски и исходы, которые могут иметь место.

Выбор альтернативы — «вершина» в процессе принятия решения. Многие пытаются избежать этот этап или переложить его на другого, так как в ходе данного этапа ЛПП вынуждено брать на себя обязательства по будущим действиям. Хороший анализ альтернатив резко сужает рамки выбора. При выборе альтернативы используются три подхода: учет прошлого опыта; проведение эксперимента; исследование и анализ [12].

Выполнение решения — состоит из организации выполнения решения, анализа и контроля выполнения. Наиболее распространенная ошибка — предположение, что если выбор сделан, то решение будет выполнено. Выполнение решения — это устранение проблемы, его породившей, по отношению к которой было принято решение. Организация выполнения решения предусматривает координацию усилий многих людей, установление ответственности за действия, планирование мероприятий, превращающих решение в реальность.

Контроль предполагает выработку и использование системы обнаружения отклонений в действиях по выполнению решения.

Еще одна модель процесса принятия решений изображена на рис. 2.4. Ее отличают ориентация на коллективное решение задач и увязывание отдельных процедур как составных частей сложных этапов из модели О.С. Виханского и А.И. Наумова с методами их выполнения. Таким образом, модель на рис. 2.4 носит методологический характер.

Индивидуально каждый может решать задачи, выбирая различные варианты своих действий. Для того чтобы решать задачи коллективно, нужны специальные методы конструктивной работы, иначе вряд ли система управления добьется хороших результатов.

В Японии, европейских странах, США с успехом используются принципы и методы коллективного решения задач: методы систематизированного поиска и психологической активизации творчества.

Для каждого этапа процесса коллективного решения задач модель на рис. 2.4 предлагает использовать один или несколько методов, что хорошо отражает идеи плюрализма и «спора» моделей, актуальных при многоаспектном рассмотрении сложных задач. Необходимо заметить, что некоторые из данных методов могут применяться на разных этапах.

Систематизированно и подробно все методы рассмотрены в [22]. В настоящей работе некоторые методы сгруппированы в разд. 6.4 — «Методы выяснения причин возникновения проблем». Другая часть методов диагностики задач вошла в разд. 6.5 — «Методы системного анализа сложных задач». Метод коллажей и фантазий выделен в самостоятельный разд. 3.5 применительно к ситуации моделирования невербальных рассуждений экспертов. Наконец, модифицированный метод Дельфи и

метод обмена мнениями кратко рассмотрены в разд. 4.3, как инструмент ЛПР в выработке консенсуса и работе с коалициями экспертов в СППР.

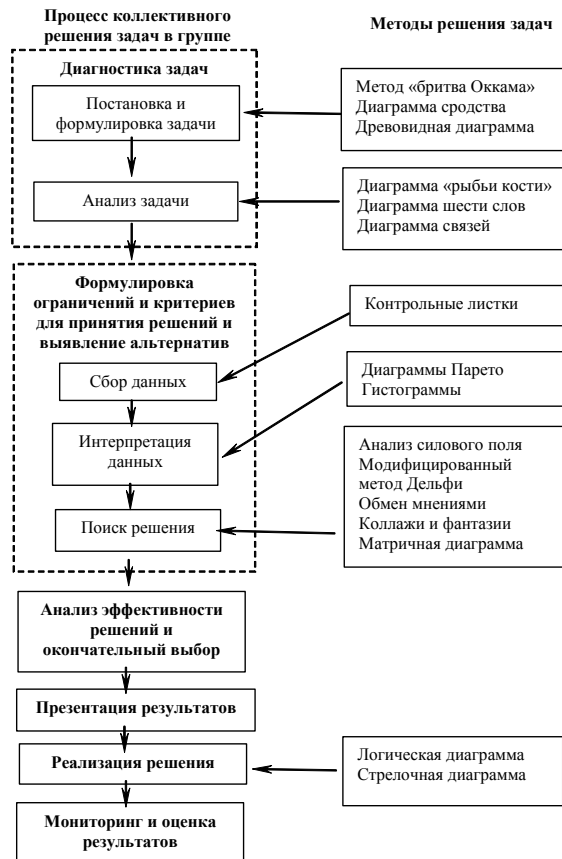


Рис. 2. 4 — Модель процесса принятия решений М.В. Самсоновой, В.В. Ефимова

2.5. Компьютерные системы поддержки принятия решений

Компьютерные системы поддержки принятия решений — результат развития систем обработки информации и управления и систем управления базами данных в начале 70-х годов прошлого века. В настоящее время КСППР используются в различных областях человеческой деятельности, а темпы их разработок возрастают. Следует отметить, что хотя КСППР широко применяются во всем мире, в России и СНГ системам этого типа пока еще не уделяется должное внимание.

Несмотря на распространенность КСППР, общепризнанное определение данного термина пока не найдено и дискуссионно так же, как и вопрос отнесения различных типов систем к КСППР. Ниже приведены несколько определений КСППР.

1. КСППР — это компьютерная информационная система, используемая для поддержки различных видов деятельности при принятии решения в ситуациях, где невозможно или нежелательно иметь автоматические системы, которые полностью выполняют весь процесс принятия решения. КСППР не заменяет ЛПР, автоматизируя процесс принятия решения, а оказывает ему помощь в ходе решения поставленной задачи [105].

2. КСППР — это интерактивная компьютерная система, предназначенная для поддержки принятия решений в слабоструктурированных и неструктурированных проблемах различных видов человеческой деятельности [106].

3. КСППР — комплекс взаимосвязанных моделей с соответствующей информационной поддержкой исследования, экспертные и интеллектуальные системы, включающие опыт решения задач управления и обеспечивающие участие коллектива экспертов в процессе выработки рациональных решений [107].

4. КСППР — это средство для «вычисления решений», которое основано «на использовании ряда процедур по обработке данных и суждений, помогающих ЛПР в принятии решения» [108].

5. КСППР предназначена для решения задач классификации большого количества вариантов (альтернатив) или выбора из них наилучших вариантов на основе информации, получаемой от ЛПР [109].

6. КСППР — это организационно-технический комплекс, обеспечивающий ситуационное управление объектом, основанное на всестороннем анализе политической, экономической, социальной и технологической ситуации, проводимом человеком или группой людей с привлечением информационно-аналитических технологий [103].

Анализ приведенных определений позволяет сделать выводы.

Существует терминологическая путаница, которая возникла в связи с эволюцией понимания КСППР. На это обращено внимание и в работе [110]. Английский эквивалент этого понятия — Decision Support System

— DSS. Например, к системам этого типа относят системы, реализующие технологию OLAP (англ. on-line analytical processing). Эта категория программного обеспечения — интерфейс доступа к многомерным, многосвязанным данным в больших (и сверхбольших) базах данных, которые выдаются в форме, заданной пользователем и удобной для анализа. Однако с начала 1970-х годов до начала 1990-х в Советском Союзе публиковались оригинальные и переводные статьи, также посвященные системам поддержки принятия решений. Тогда чаще применялся другой английский эквивалент — “Decision-Making Support System — DMSS”. Несмотря на полную идентичность русского названия, речь идет о различных аспектах систем одинаковой направленности. Раньше под поддержкой принятия решений понимался инструментарий выработки рекомендаций для ЛПР: формирования альтернатив и множества оценочных критериев; получения оценок альтернатив по критериям. Эта точка зрения зафиксирована, например, в определениях 1, 2, 4, 5. Сейчас то же понятие означает инструментарий обслуживания ЛПР: подготовки объемных со сложными взаимосвязями баз данных, организации гибкого и удобного доступа к ним мощными средствами формирования запросов, получения результатов запросов в максимально удобной форме, использования мощных генераторов отчетов. Эта точка зрения зафиксирована, например, в определении 6.

Очевидно, что оба рассмотренные инструментария призваны обеспечить процесс принятия решений. Однако первый сосредоточен на сравнении альтернатив с целью выбора лучшей, т.е. «берет» на себя часть процедур процесса принятия решений. Второй — на подготовке данных для последующего анализа, т.е. выполняет «обслуживающие», второстепенные, вспомогательные функции. Фактически второй инструментарий не предполагает выдачу рекомендаций. Он выдает только данные, а процесс формирования альтернатив, их сравнения и выбора лучшей остается «за скобками». Первый инструментарий предполагает, что, во-первых, вся информация, необходимая для выдачи рекомендации, должна быть собрана и, во-вторых, она должна быть оформлена в виде модели выбора: «альтернативы + критерии + оценки». Поэтому можно сказать, что второй инструментарий, по сути, есть подготовительный этап к первому. Для второго инструментария больше подошло бы название «система подготовки данных для принятия решения».

В настоящей работе придерживаются точки зрения, когда развитая КСППР должна объединять оба инструментария.

КСППР применяются в системах управления производственными мощностями, непредвиденными ситуациями, вспомогательными службами, взаимоотношениями с клиентами и поставщиками. По имеющимся оценкам, в сфере приложений КСППР попадает почти половина за-

дач, возлагаемых на информационные службы организаций и предприятий [103]. Это подтверждается при анализе рынка информационных систем. Так, мировой рынок, например, ERP-систем (англ. enterprise resource planning systems), оценивается в настоящее время оборотами порядка 25 млрд. долларов. Рынок DSS-систем, возникший в середине 1990-х годов, сейчас оценивается суммой около 10 млрд. долларов и растет существенно большими темпами, чем рынок корпоративных систем управления: порядка 30% в год против 10 — 15% роста ERP-рынка. Следует предположить, что в ближайшие пять лет ожидается достижение паритета [103].

Известны и продолжают появляться новые классификации КСППР. Ниже рассмотрены некоторые, наиболее популярные из них и дано краткое описание основных классов.

Классификация Д.Д. Power. Предложено различать следующие КСППР: ориентированные на данные (англ. data-driven, data-oriented DSS); ориентированные на модели (англ. model-driven DSS); ориентированные на знания (англ. knowledge-driven DSS); ориентированные на документы (англ. document-driven DSS); ориентированные на коммуникации и групповые СППР (англ. communications-driven, group DSS); интер-организованные и интра-организованные (англ. inter-organizational, intra-organizational DSS); специфически функциональные или общего назначения (англ. function-specific, general purpose DSS); на базе Web (англ. web-based DSS) [111].

Классификация по обрабатываемым данным. В ней принято различать следующие КСППР: стратегических решений (англ. executive information systems — EIS) и многофункциональные DSS-системы анализа и исследования данных.

EIS-системы — управленческие информационные системы для обслуживания стратегических информационных потребностей высшего руководства. Высшее руководство получает необходимую информацию из многих источников: писем, телефонных звонков, периодических изданий и докладов, подготовленных вручную и компьютерными системами, встреч и общественной деятельности. Таким образом, большая часть информации исходит из некомпьютерных источников. Цель компьютерных систем поддержки принятия стратегических решений — обеспечить высшее руководство непосредственным и свободным доступом к информации, актуальной при реализации стратегических целей фирмы. Отсюда EIS должны быть просты в эксплуатации, понимании и обеспечивать доступ к базам данных.

DSS-системы — предполагают более глубокую обработку данных для использования в процессе принятия решений. Системы этого типа, в отличие от EIS-систем, рассчитаны на пользователей, имеющих про-

фессиональные знания о предметной области, и умеющих использовать компьютерные технологии. Этим системам присущи черты искусственного интеллекта, за счет возможности отображения исходных данных в рекомендации решений поставленной задачи. Подобные системы иногда называют динамическими из-за способности обрабатывать неожиданные (ad hoc) запросы. В последнее время к КСППР относят только DSS-системы.

Классификация по группам обрабатываемых данных. В ней принято различать КСППР, работающих с областями: детализированных данных OLTP-системы (англ. online transaction processing), агрегированных показателей OLAP-системы, а также закономерностей-шаблонов (англ. data mining).

OLTP-системы — это информационно-поисковые системы, используемые как надстройки над системами обработки данных или как хранилища данных. Задачи OLAP-систем — обобщение, агрегация, гиперкубическое представление информации и многомерный анализ. Это обычно многомерные СУБД и реляционные базы с предварительной агрегацией данных.

Интеллектуальный анализ данных (англ. data mining) основан на поиске в данных скрытых закономерностей для принятия решений. При этом накопленные в базах данных сведения автоматически обобщаются до информации, характеризующейся как знания. Это означает, что найденные шаблоны должны отражать неконтрастные регулярности в данных.

Классификация по архитектурам. В ней принято различать КСППР: функциональные; независимые витрины данных; двухуровневые хранилища данных; трехуровневые хранилища данных.

Функциональная КСППР наиболее проста с архитектурной точки зрения: «Пользователи — источники данных». Характерная черта — анализ осуществляется с использованием данных из оперативных систем. Такие системы часто встречаются на практике, в особенности в организациях с невысоким уровнем аналитической культуры и недостаточно развитой информационной инфраструктурой.

Независимые витрины данных: «пользователи — витрины данных (для ответов на конкретные вопросы) — источники данных». Витрины данных встречаются в крупных организациях с большим количеством независимых подразделений, имеющих свои собственные отделы информационных технологий.

Двухуровневые хранилища данных: «пользователи — хранилище данных — источники данных» строятся централизованно для представления информации в рамках компании. При этом все подразделения организации должны согласовать спецификации и процессы преобразо-

вания данных. Обеспечивать такую архитектуру должна команда профессионалов в области хранилищ данных.

Трёхуровневые хранилища данных: «пользователи — витрины данных — хранилище данных — источники данных» — единый централизованный источник корпоративной информации. Витрины данных представляют подмножества данных из хранилища, организованные для решения задач отдельных подразделений компании. Конечные пользователи имеют возможность доступа к детальным данным хранилища, в случае если данных в витрине недостаточно, а также для получения более полной картины состояния бизнеса.

Архитектурно-технологическая схема КСППР. Основу КСППР составляет комплекс взаимосвязанных моделей, обеспечивающих информационную поддержку решений, экспертные и интеллектуальные системы, включающие знания и опыт решения задач управления и обеспечивающие участие коллектива экспертов в процессе выработки рациональных решений.

На рис. 2.5 приведена архитектурно-технологическая схема информационно-аналитической поддержки принятия решений.

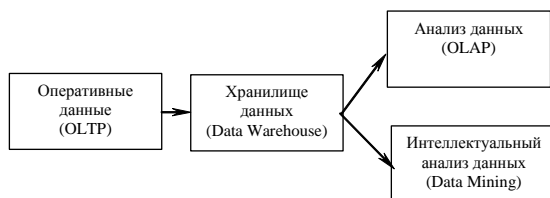


Рис. 2.5 — Архитектурно-технологическая схема КСППР

Первоначально информация хранится в оперативных базах данных OLTP-систем. Агрегированная информация организуется в многомерное хранилище данных. Затем она используется

в процедурах многомерного OLAP-анализа и для интеллектуальной обработки данных.

Рассмотрим более подробно каждый элемент этой схемы.

Хранилища данных. Для эффективного функционирования системы управления необходима актуальная информация в удобном для анализа виде и в реальном масштабе времени. Доступность такой информации важна как для оценки текущего положения дел, так и прогнозов на будущее. К тому же, основа для принятия решений — реальные данные. Однако, если они хранятся в базах данных информационных систем, при их анализе возникают сложности: 1) значительно возрастает время обработки запросов; 2) трудно поддерживать различные форматы данных; 3) усложняется кодирование; 4) невозможно анализировать объемные ряды ретроспективных данных и др.

Эти сложности преодолеваются в КСППР созданием хранилища данных. Его функции — интеграция, актуализация и согласование оперативных данных от разнородных источников для выработки общего, непротиворечивого взгляда на объект управления, составление разнообразной отчетности, оперативная аналитическая обработка и интеллектуальный анализ данных. Данные в хранилище представлены многомерными структурами «звезда» или «снежинка» [106].

Б. Инмон (B. Inmon) определяет хранилища, как «предметно ориентированные, интегрированные, неизменяемые, поддерживающие хронологию наборы данных, организованные с целью поддержки управления» и призванные выступать в роли «единого и единственного источника истины», обеспечивающего ЛПР и аналитиков достоверной информацией для оперативного анализа и принятия решений [106].

Предметная ориентация хранилища данных означает, что данные объединены в категории предметных областей, а не приложений, с ними работающих. Интегрированность означает, что данные удовлетворяют требованиям всей системы управления, а не только одной из ее функций, что гарантирует одинаковость результатов в отчетах, сгенерированных по запросам разных аналитиков. Привязка ко времени означает, что хранилище отображает систему управления в ретроспективной динамике и обеспечивает восстановление данных на любой момент времени в прошлом. При этом атрибут времени есть в хранилищах данных явно. Неизменность означает, что данные в хранилище могут только добавляться и не могут модифицироваться.

Р. Хакаторн, другой основоположник концепции хранилищ данных, писал, что цель этой специальным образом организованной информации — обеспечить «единый образ существующей реальности».

Хранилище данных функционирует следующим образом. По заданному регламенту в него поступают данные из различных источников — баз данных систем оперативной обработки, упорядочиваемые хронологически. В результате доступные и актуальные данные об объекте управления собираются в одном месте, в одном формате, согласовываются и иногда агрегируются.

На основе хранилища данных составляется отчетность, идут их аналитическая обработка и интеллектуальный анализ.

Концепция оперативной аналитической обработки информации в хранилищах данных КСППР. В основе концепции оперативной аналитической обработки лежит многомерное представление данных. Термин OLAP ввел E.F. Codd в 1993 г. В 1995 г. по результатам анализа недостатков традиционной реляционной модели и в первую очередь невозможность «объединять, просматривать и анализировать данные с точки зрения множественности измерений, т.е. самым понятным для

корпоративных аналитиков способом», был сформулирован тест быстрого анализа разделяемой многомерной информации (англ. fast analysis of shared multidimensional information — FASMI). Быстрый — означает, что КСППР отводится на ответ по запросу около пяти секунд, «анализ» — это способность логического и статистического, доступного для пользователя анализа. Слово «разделяемый» означает многоуровневую защиту информации. Наконец, система должна поддерживать концептуальное представление множественных иерархий и «доставлять» информацию туда, где она необходима.

По Е. Кодду, многомерное концептуальное представление (англ. multi-dimensional conceptual view) — естественный взгляд управляющего персонала на объект (рис. 2.6).

Основные понятия многомерной модели данных: 1) предмет анализа — один или несколько количественных показателей, например, объём продаж средств вычислительной техники на рис. 2.6; 2) аналитические измерения (по осям) — множества иерархий объектов, обеспечивающих информационный контекст числового показателя; 3) члены измерений — точки или участки, откладываемые на осях гиперкуба; 4) ячейки — мельчайшая часть куба, соответствующая конкретному значению показателя. Ячейка идентифицируется комбинацией членов всех измерений и показателем.

В итоге, многомерное концептуальное представление — множественная перспектива нескольких независимых измерений, по которым анализируются совокупности данных.

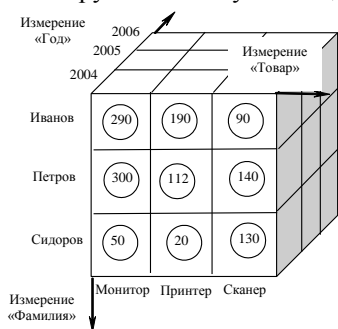


Рис. 2.6 — OLAP - куб

В многомерной модели определены операции выбора желаемого уровня детализации информации по измерениям. Каждое измерение при этом включает несколько направлений консолидации данных, как серии последовательных уровней обобщения, где высший уровень — большая степень агрегации данных. Операция спуска (англ. drilling down) соответствует движению от высших ступеней консолидации к низшим. Операция подъема (англ. rolling up) — движение

от низших уровней к высшим.

Традиционные реляционные серверы не обеспечивают эффективное выполнение сложных OLAP-запросов и поддержку многомерных представлений данных. В этой связи используются три типа специальных серверов баз данных — реляционной (ROLAP), многомерной (MOLAP)

и гибридной (HOLAP) оперативной аналитической обработки. Они позволяют выполнять OLAP-операции в хранилищах данных, построенных с использованием реляционных СУБД.

ROLAP-серверы размещаются между основным реляционным сервером с хранилищем данных и клиентской программой. Они определяют материализуемые представления, возможные запросы пользователей в их терминах, генерируют SQL-серверы для основного сервера, планируют запросы и распределяют ресурсы.

MOLAP-серверы напрямую поддерживает многомерные представления с помощью специального механизма памяти. Серверы MOLAP обладают хорошей производительностью и функциональностью, но плохо масштабируются для больших баз данных.

HOLAP-серверы имеют гибридную архитектуру и применяют подход ROLAP для разреженных областей многомерного пространства и MOLAP — для плотных областей. Для этого они разделяют запрос на части, отсылают их к соответствующим фрагментам данных, комбинируют ответы и предоставляют результат пользователю.

Интеллектуальный анализ данных в КСППР. Наибольший интерес в КСППР представляет интеллектуальный анализ данных, что дает полный и глубокий анализ проблемы, контрастирует скрытые закономерности и позволяет принять качественное решение.

Промышленные предприятия, корпорации, ведомства, органы государственной власти и местного самоуправления уже накопили значительные объемы данных. Они хранят в себе неявные, неконтрастные причинно-следственные связи, зависимости, закономерности, тенденции — основу для выработки в СППР стратегии развития и поиска новых, эффективных решений.

Интеллектуальный анализ данных — это один из процессов поддержки принятия решений, основанный на поиске в данных скрытых закономерностей (шаблонов информации), которые после накопления и обобщения рассматриваются как знания.

В табл. 2.1 представлены для сравнения несколько задач аналитической обработки и интеллектуального анализа данных.

Различают пять стандартных типов закономерностей, которые позволяют выявлять методы интеллектуального анализа данных: ассоциация, последовательность, классификация, кластеризация и предсказание.

Ассоциация — несколько событий, связанных друг с другом. Если существует цепочка событий, упорядоченных во времени, то говорят о последовательности. Классификация — выявление признаков группы принадлежности объекта путем анализа уже классифицированных объектов и выработка соответствующего набора правил. Кластеризация

отличается от классификации тем, что однородные группы объектов не известны, и их необходимо найти. Такие группы и рассматриваются как контрастируемые знания.

На основе временных рядов, исторической информации отыскиваются шаблоны динамики целевых показателей, и с некоторой вероятностью предсказывается поведение системы управления в будущем.

В общем случае интеллектуальный анализ данных в КСППР включает выявление: 1) закономерностей и их использование для предсказания неизвестных значений; 2) исключений для толкования аномалий в найденных закономерностях.

Новые компьютерные технологии интеллектуального анализа данных — экспертные системы, нейронные сети, нечеткие системы, генетические алгоритмы, системы рассуждений с базами прецедентов.

Перспективно применение в КСППР междисциплинарных (комбинированных методов) принятия решений: интегрированных экспертных систем, гибридных и гибридных интеллектуальных систем, сочетающих достоинства математических и эвристических моделей, использующих резонансные эффекты улучшения качества рекомендаций и развивающихся в рамках социальной парадигмы искусственного интеллекта.

Таблица 2.1 — Задачи аналитической обработки и интеллектуального анализа данных

Аналитическая обработка (OLAP)	Интеллектуальный анализ данных (Data Mining)
Каковы средние показатели брака по постам сборочного автомобильного производства?	Есть ли точные шаблоны в квалификационных описаниях комбинаций людских ресурсов постов сборочного автомобильного производства, которым присущ брак?
Каковы средние зарплаты уволившихся рабочих-сборщиков, в сравнении с зарплатами рабочих вспомогательных производств?	Имеются ли характерные портреты сборщиков, которые, по всей вероятности, собираются уволиться?
Какой численный и квалификационный состав сборочных постов, дающих наименьший и наибольший брак?	Существуют ли стереотипные схемы «Квалификация — состав — заработная плата — опыт работы на модели автомобиля» для случаев отклонения от заданного качества продукции?

Интеграция аналитической обработки и интеллектуального анализа данных в КСППР. Несмотря на то, что аналитическая обработка сосредоточена на доступе к многомерным данным, а методы ин-

теллектуального анализа в основном обрабатывают «плоские» одномерные таблицы и реляционные данные, интеграция этих подходов взаимобогащает их функциональность, чтобы интегрированная информационная технология обеспечивала одновременно многомерный доступ и поиск закономерностей. По словам N. Raden многие компании создали прекрасные хранилища данных, идеально разложив по полочкам горы неиспользуемой информации, которая не обеспечивает ни быстрой, ни грамотной реакции на рыночные события [112].

В этой связи K. Parsaye для обозначения такой интеграции вводит составной термин «многомерный интеллектуальный анализ данных» (англ. OLAP data mining). Инструментальное средство такого анализа должно находить закономерности как в детализированных, так и в агрегированных с различной степенью обобщения данных. Анализ многомерных данных идет над специальным гиперкубом с ячейками, хранящими не численные значения показателей, а вероятности соответствующего сочетания значений атрибутов. Проекция такого гиперкуба, исключающие из рассмотрения отдельные измерения, также должны исследоваться на предмет поиска закономерностей.

J. Han предлагает несколько вариантов интеграции двух подходов к разработке КСППР [112]: интеллектуальный анализ любого результата над любым фрагментом любой проекции гиперкуба (англ. cubing then mining); результаты интеллектуального анализа должны представляться гиперкубом для многомерного анализа подобно данным, извлеченным из хранилища (англ. mining then cubing); над результатом каждого шага многомерного анализа автоматически активируются однотипные механизмы интеллектуальной обработки (англ. cubing while mining) .

Применение компьютерных систем поддержки принятия решений. Основные сферы применения КСППР на сегодняшний день: крупноформатная торговля, банки и финансовые компании, телекоммуникации, промышленность и государство [113].

Крупноформатная торговля и компании электронной коммерции одни из первых начали применять КСППР на практике для решения задач: анализа ассортимента (селективный маргинальный доход, оборачиваемость запасов, статистическое управление запасами, фондоотдача); распределения площадей, раскладки; анализа эффективности деятельности менеджеров и мотивации персонала; планирования и анализа эффективности рекламы, акций, распродаж и т.п.; управления ценообразованием.

Банки и финансовые компании применяют КСППР для банковского ритейла (платежные пластиковые карты и чеки); анализа рисков; предотвращения мошенничества, прежде всего с пластиковыми картами; анализа потребительского поведения и проектирования новых финансовых услуг. Последнее основано на анализе и формировании по-

требительских групп, которые характеризуются сходным поведением. Результатом этой работы являются проекты, например, молодежных жилищных кредитов, условия овердрафтов, VIP-программы клиентского обслуживания. При этом надо отвечать на вопросы: что такое «молодежь?», кто такой VIP-клиент? и т.д. Предотвращение мошенничества — это перспективная зона использования методов ИИ. В страховых компаниях КСППР еще широко не распространены, но это только подчеркивает потенциальную перспективность данного рынка.

В телекоммуникационных компаниях, прежде всего мобильной связи, роль КСППР связана с проектированием новых услуг, которое основано на выявлении устойчивых клиентских групп и преимущественного клиентского поведения. Этот рынок по времени жизни неисчерпаем.

В промышленности к сферам применения КСППР относятся: управление взаимоотношениями с клиентами; статистическое управление запасами; финансовое, бюджетное, ресурсное планирование и управление; анализ и риски.

Применения КСППР в промышленности определяются условиями перехода к модели постиндустриального (информационного) общества (см. разд. 1). Здесь КСППР позволяют: 1) уделять внимание стратегическому планированию, т.е. анализировать исторические данные по структуре себестоимости, динамике цен; 2) изучать стратегию иностранных конкурентов, т.е. анализировать динамику рынков; 3) уделять больше внимания производственной функции, т.е. анализировать затраты по управлению активами, динамику тарифов, эффективность использования оборудования и фондоотдачу; 4) устранить коммуникативные барьеры, т.е. анализировать исторические данные по параметрам реализации внутренних бизнес-процессов и эффективность результатов; 5) признать ценность развития информационных связей, т.е. анализировать исторические данные взаимоотношений с клиентами и поставщиками.

Государство использует КСППР, прежде всего, в области обороны для решения задач планирования и управления операциями и военно-промышленными ресурсами. Так, по результатам первой войны в Ираке экономический эффект от использования систем искусственного интеллекта был оценен в сумму порядка 100 млн. долларов. Это привело приблизительно к трехкратному увеличению ассигнований на развитие данных информационных технологий в интересах Министерства обороны США. Сегодня в данной области ассигнования уже оцениваются суммами в миллиарды долларов [113].

В области государственного строительства роль КСППР пока невелика. Потенциально их область использования связана с оценкой эффективности государственных и муниципальных программ. Это связа-

но, прежде всего, с тем, что государственные и муниципальные программы не сводятся к экономическому эффекту, как таковому.

В начале этого века начал приобретать новое звучание процесс электронизации деятельности российских органов власти, деловой сферы и населения. Концепция архитектуры «электронного правительства», складывающаяся из перечня услуг органов государственной власти, показателей эффективности и качества реализации государственных услуг, электронных административных регламентов деятельности органов государственной власти, информационных ресурсов, технического и программного обеспечения, требует расширения использования аналитических возможностей информационных технологий [113].

Концепция электронной демократии, предполагающая: гражданский контроль, выборы и референдумы, самоорганизацию населения, участие населения в принятии государственных решений, расширение технологической возможности обмена мнениями, также предполагает увеличение значения информационной аналитики [113].

Интеллектуальные системы поддержки принятия решений. Компьютеры и информатика (англ. computer science) предоставили возможность проводить эксперименты и манипулировать базами данных и знаний так же, как в прошлом ученые-математики оперировали числами, функциями и операторами. Это значительно продвинуло решение проблем моделирования управляемых процессов и тем самым существенно расширило круг доступных для изучения систем управления в технике, экономике, экологии и других прикладных областях [114 — 116]. При этом формализация понятия системы S должна быть достаточно общей, чтобы, с одной стороны, охватить максимально широкий круг реальных объектов, включая и те из них, которые могут быть пока описаны только в качественных, лингвистических терминах, а с другой — иметь необходимые структуры и операции для использования аппарата математики в тех случаях, когда допустимо описание в рамках существующих математических теорий.

Такой формализм системы можно ввести на языке теории множеств [117], наделяя их теми или иными свойствами. Пусть заданы множества: $\{V_i, \mid i \in I$, где I — множество индексов. Тогда системой [117] называется множество:

$$S \subset \times \{V_i \mid i \in I, \quad (2.1)$$

а все компоненты $V_i, i \in I$ декартова произведения $\times V_i$ называются ее элементами.

В [114] определено понятие общей временной системы. При этом под множеством моментов времени понимается линейное упорядочен-

ное множество T , наделенное отношением порядка и имеющее минимальный элемент 0 . Пусть A и B — произвольные множества, A^T и B^T — множества всех возможных отображений T на A и B соответственно, а $Z \subset A^T$ и $\Omega \subset B^T$. Тогда общей временной системой на Z и Ω называется множество $S \subset Z \times \Omega$.

Используя это определение, можно рассматривать динамику системы, независимо от того, в какой конкретной форме она задана: дифференциальным уравнением, компьютерной базой знаний, аппроксимирована нейросетью или другим способом. Именно это определение лежит в основе получившего широкое распространение «интеллектуального управления», «интеллектуальных систем управления» [115, 118 —120].

А.А. Ерофеев и А.О. Поляков отмечают, что теория интеллектуальных систем управления возникла на стыке двух фундаментальных научных направлений — теории автоматического управления и теории систем в связи с необходимостью организации управления как системно-сложного взаимодействия субъектов и объектов. Для обеспечения решения практических задач теория интеллектуальных систем управления разрабатывалась, как прикладная теория и, сохраняя, тем не менее, возможную строгость построений. Основным критерием успеха и полезности своего существования эта теория считает обеспечение создания прикладных систем эффективного управления в наиболее сложных ситуациях, когда заведомо невозможно обеспечить полезное управление другими методами [115].

Такие методы актуальны при невозможности установить и (или) скорректировать до необходимой степени точности модель системы S . Имеющуюся неопределенность возмущений и характеристик компенсируют, например, за счет внедрения в систему управления некоторого заимствуемого у человека «знания» или «интеллекта».

Обычно считается, что система управления — интеллектуальная, если она ориентирована на использование привносимой от человека (на уровне эвристик, прецедентов, аналогий, схем и других подобных приемов) информации для компенсации разного рода неопределенностей.

Термин «интеллектуальная система управления» используется для того, чтобы подчеркнуть, что рассматривается именно основное, управляющее свойство интеллекта, а также для выделения класса систем, в которых, начиная с некоторого уровня сложности, невозможно иное управление, кроме как интеллектуальное [115].

В отличие от фундаментальных, прикладная теория интеллектуального управления не абстрагируется от неукладывающихся в аналитические зависимости факты и ситуации. Напротив, она делает их основой и строится для преодоления математической непредставимости или не-

возможности достижения адекватного математического представления управления. В рамках прикладной теории интеллектуальных систем управления и как один из классов интеллектуальных информационных систем [121] развиваются компьютерные интеллектуальные системы поддержки принятия решений (ниже — интеллектуальные системы поддержки принятия решений).

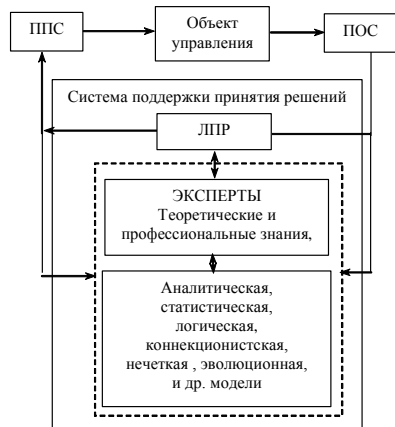
Их главная отличительная особенность состоит в применении компьютерных средств обработки информации, телекоммуникаций, методов математики, теории принятия решений, исследования операций, искусственного интеллекта для анализа сложных задач, поиска решений, анализа их эффективности, реализации, мониторинга и оценки результатов в малых графпах «ЛППР — эксперты».

В работе [122] для динамических проблемных областей и реального масштаба времени дано следующее определение: интеллектуальные системы поддержки принятия решений относятся к классу интегрированных интеллектуальных систем, сочетающих строгие модели и методы поиска решения с нестрогими, эвристическими (логико-лингвистическими) моделями и методами, базирующимися на знаниях специалистов-экспертов, моделях человеческих рассуждений, неклассических логиках и накопленном опыте.

В работе [103] выделяются основные направления моделирования процессов интеллектуализации процессов процедур поддержки принятия решений: 1) моделирование диалога между человеком и КСППР, включая процессы взаимодействия, взаимопонимания и взаимного учета целей, а также взаимообучения партнеров по диалогу; 2) моделирование совместного использования различных способов представления знаний (процедурных, продукционных, семантических); 3) моделирование многокритериального анализа альтернатив; 4) моделирование процессов хранения и доступа к накопленным формализованным знаниям (управление знаниями); 5) построение заключений экспертами; 6) обработка не полностью определенной информации; 7) построение и использование не только баз данных, знаний и моделей, но и баз прецедентов, что позволит в обозримом будущем перейти от управления знаниями к управлению отношениями; 8) моделирование процессов развития КСППР как сложных человеко-машинных систем.

Разработка интеллектуальных КСППР (рис. 2.7) предполагает разработку информационной технологии взаимодействия коллектива (малой группы) людей, включающей ЛППР и экспертов, средств вычислительной техники, систем телекоммуникации и отображения информации в процессе цикла принятия решений, рассмотренного в разд. 2.4, а также специального программного обеспечения.

Такое программное обеспечение помимо поддержки возможных сценариев работы СППР в различных режимах (см. разд. 2.3) и обеспечения участников СППР функциями хранилищ данных, аналитической обработки, интеллектуального анализа данных, которые были рассмотрены выше, должно выполнять и специфические функции, в частности: 1) обеспечивать консультациями ЛПР по выполняемой им в составе СППР функциональной нагрузке (см. разд. 2.1); 2) обеспечивать консультациями экспертов в процессе рассуждений над подзадачами сложной задачи, решаемой в СППР; 3) обеспечить возможность моделирования работы СППР на общем табло, т.е. в месте, доступном для общего обозрения участниками СППР.



Обозначения: ППС, ПОС – преобразователи информации в каналах прямой и обратной связи, соответственно

Рис. 2.7 — Интеллектуальное управление в системах поддержки принятия решений

Функция обеспечения ЛПР консультациями предполагает, что ЛПР может обратиться к компьютеру за помощью в анализе и углублении своего понимания сложной задачи. Режим актуален как при подготовке ЛПР к заседанию СППР, так и во время самого заседания. Функция возможна как тренажер ЛПР.

Функция консультирования обеспечивает доступ эксперта конкретной профессии как к теоретическим, так и узкопрофессиональным знаниям и опыту его лично, так и других экспертов с аналогичной профессией, и имитацию рассуждений при решении подзадачи с использованием широкого спектра методов (см. разд. 3). Становится возможным и автоматическое выполнение в этом режиме функций ЛПР. То есть эксперт, работая над подзадачей, может обеспечиваться функциями со стороны «компьютерного ЛПР», имитирующего работу реального ЛПР-человека.

Функция моделирования работы СППР дает возможность всем участникам мероприятия в ходе анализа и решения сложной задачи наблюдать за компьютерным вариантом коллективного обсуждения с отображением на табло рассуждений ЛПР и экспертов. Такой вариант помимо преимуществ сжатия временной шкалы и возможности «заглянуть вперед» полезен и для подбора экспертов, выработки сценария проведения заседания, оценки деятельности экспертов, выявления «уз-

ких мест» и «слабых звеньев» в работе СППР и повышения эффективности их функционирования.

Концепция интеллектуальных КСППР зародилась в научной среде, когда появилось ощущение того, что при решении практических задач в автоматизированном режиме не удастся обеспечить качество консультаций ЛПР, работающего не в глобальных, «среднестатистических», «региональных» условиях, для которых математике удалось сформулировать и записать причинно-следственные связи существенных для управления параметров, а в локальных, специфических условиях конкретной системы управления, когда такие формализованные представления позволяют вычислить только «ориентир», общую схему действий. Полагалось, что «довести до ума» этот ориентир, схему — функция человека, владеющего эвристиками.

В этой связи и появилась идея эвристической коррекции фундаментальных математических (аналитических) моделей. Для нее характерно, что в ходе разработки модели эксперт или ЛПР предоставляют свои знания или примеры своей узкопрофессиональной деятельности (эвристики), которые используются совместно с фундаментальными моделями. Иногда от последних вообще отказываются либо включают их в эвристические модели.

В настоящее время для организации эвристической коррекции в интеллектуальных СППР разработано много интересных методов в прикладных теориях: исследовании операций (например, методы многокритериальной оптимизации), теории принятия решений (например, деревья решений), системном анализе (метод деревьев целей, экспертных оценок, деловых игр, мозговых штурмов и др.) и особенно в искусственном интеллекте (нечеткие и нейросистемы, генетические и эволюционные алгоритмы, СВР-системы, системы с «хаосом» и др.).

В связи со сменой “isolation assumption” парадигмы ИИ на социальную парадигму, обеспечивающую переход «от существующего к возникающему», идет переосмысление и накопленного при разработке КСППР опыта. В.Б. Тарасов отмечает, что разработка различных схем интеграции существующих направлений и моделей в рамках системного подхода приводит к возникновению гибридного (синергетического) искусственного интеллекта. Именно синергетический ИИ обеспечивает необходимые предпосылки для создания коллективов и сообществ интеллектуальных систем и формирования интеллектуальных организаций [13].

Таким образом, устранить недостатки эвристической коррекции призваны системы, основанные на «теории интегрированных компьютерных систем поддержки принятия решений» [13], которые позволяют

решать задачи, связанные со сложными процессами, функционирующими в среде разнотипных данных.

3. МЕТОДЫ МОДЕЛИРОВАНИЯ РАССУЖДЕНИЙ ЭКСПЕРТОВ ПРИ РЕШЕНИИ ЗАДАЧ В СИСТЕМАХ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

3.1. Знания и опыт — основа творческого процесса в системах поддержки принятия решений

Системы поддержки принятия решений накапливают и сохраняют, используют и обрабатывают, обмениваются знаниями и опытом в машинно-приемлемой форме, «черным золотом» XXI в. Те люди, которые хотят открыть самую мощную скважину, будут в числе самых первых, кто займется массовым производством знаний и опыта с помощью компьютера [123].

Рассмотрим несколько определений понятия «знания».

1. Знания состоят из описаний, соотношений и процедур, относящихся к представляющей интерес сфере. Знания = Факты + Гипотезы + Эвристики [124].

2. Знания — это закономерности предметной области (принципы, связи, законы), полученные в результате практической деятельности и профессионального опыта, позволяющие специалистам ставить и решать задачи в этой области [98].

Второе из определений трактует знания чрезвычайно узко, что характерно для инженерии знаний, стремящейся выделить этот вид информации в свой предмет исследования, а первое — достаточно широко, что в большей степени характерно для науки в целом. Аристотель утверждал, что некоторые считают, что нет знания, так как для этого необходимо знать первое [посылки], другие же — что знание возможно, но для всего есть доказательство. Ни одно из этих мнений не истинно и не необходимо [125].

На практике знания не встречаются в каком-то очищенном, рафинированном виде, они могут принять самую разнообразную форму.

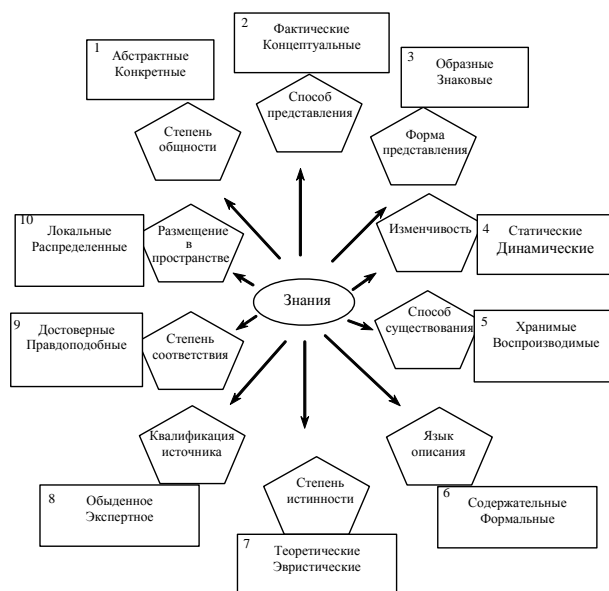
Одна из самых старых, восходящих к Канту классификаций знаний и связанная с ней проблема естествознания — разделение и взаимодействие «гуманитарного» и «точного» знания [126], т.е. философии и математики, качественной и количественной сторон реальности.

С точки зрения получения от специалиста знаний для использования в компьютерных системах иногда различают «глубинные» (абстракции, аналогии, схемы, отображающие структуру и природу процессов, протекающих в предметной области) и «поверхностные» (знания о видимых взаимосвязях между отдельными событиями и фактами в предметной области) знания [98]. Первые, еще не вербализованы, интуитивны и представляют собой понятийную структуру предметной области, поэтому скрыты от разработчика и труднее добываемы, а вторые — осознаны и сформулированы (или могут быть сформулированы) экспертом на ЯПД, хотя по качеству уступают глубинным знаниям.

В последнее время появились классификации знаний по источнику и способу их получения [127]. Например, знания, извлекаемые из специалиста — автоматизированно, и знания, извлекаемые из эмпирических баз данных, нейросетей автоматически. При этом есть мнение, что качество второго вида знаний выше качества первого вида знаний [127].

Можно классифицировать знания и по степени осознанности на три класса: неосознанное знание; превращенное знание (человек вспоминает не то, что было на самом деле, или описывает свой мыслительный процесс не так, как он проходил в действительности); отрефлексированное, адекватное знание [128].

Многие авторы вводят десять и более оснований классификации знаний. Пример — классификация С.В. Микони [129] на рис. 3.1.



Если рассматривать знания с точки зрения решения задач в некоторой предметной области, то их удобно разделить на две большие категории: факты (точные [130], теоретические [129], достоверные [129] знания) и эвристики (эвристические [129], экспертные [129], правдоподобные [129]). Факты — это хорошо известные в данной предметной области обстоятельства, они есть в учебниках и литературе [131]. Эвристики, правила — собственный опыт специалиста, и из-за недостаточной научной обоснованности и отсутствия исчерпывающих сведений пользоваться такими знаниями нужно весьма осмотрительно.

Разделение знаний на факты и эвристики — признак неоднородности знаний, неоднородности рассуждений и неоднородности мышления субъекта в ходе профессиональной управленческой или проектной деятельности (рис. 3.2). На рис. 3.2 вся деятельность человека (ЛПР, эксперты) в СППР условно разделена на три группы процессов: формализованных, частично формализованных и неформализованных [132].

Примером формализованных процессов могут служить жесткие, алгоритмические вычисления компьютера или рассуждения специалиста, решающего задачу и выполняющего вычисления в соответствии с багажом точных знаний.

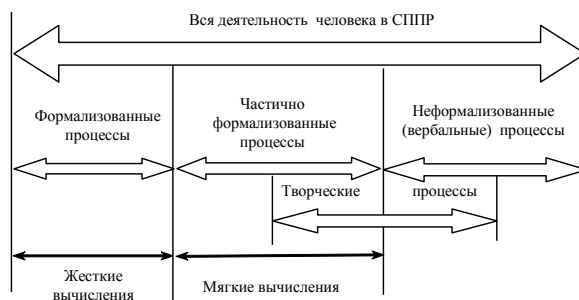


Рис. 3.2 — Виды деятельности человека в системах поддержки принятия решений

Формализованные процессы моделируются алгоритмическими, жесткими компьютерными вычислениями и выполняются с использованием формализованных знаний, аналитических закономерностей. В них можно выделить две подгруппы. Первая подгруппа знаний (назовем их «аналитическими», *A*-знаниями) формируется как результат человеческой деятельности с использованием аналитических методов.

A-знания возникли и развиваются в классическом математическом анализе, математическом программировании, теориях игр, расписаний и применяются, если свойства системы отображаемы зависимостями и, зная поведение объекта управления в некотором интервале времени, можно определить поведение вне этого интервала. *A*-знания используются при решении задач движения и устойчивости, оптимального размещения, распределения работ, ресурсов, прикладными теориями автоматического управления, оптимальных решений и др.

Вторая подгруппа знаний, используемых в формализованных процессах (назовем их «статистическими», *C*-знаниями), — результат отображения свойств и поведения системы случайными событиями, стохастическими переменными и статистическими закономерностями. Такие знания формируются в теориях математической статистики, статистического имитационного моделирования, выдвижения и проверки статистических гипотез, массового обслуживания и статистического анализа. Однако уровень оформления знаний и в этой области далек от систематизации и формализации.

Третья подгруппа знаний — (назовем их «логическими», *J*-знаниями), возникла как следствие теоретико-множественных представлений Г.Кантора, допускающих любые отношения. Конкретизацией отношений и правил их использования получают одну из алгебр логик, а доказываемые в ней теоремы приобретают силу логических законов, логических знаний. Используя их, можно, например, получить систему с меньшим числом элементов без потери требуемой функциональности. Логические знания относятся к детерминистским, хотя возможно их расширение и в сторону вероятностных. Они применяются при исследовании естественных и профессиональных языков, технических и человеко-машинных систем, в которых взаимодействие элементов еще не настолько ясно, чтобы выразить его аналитическими или статистическими зависимостями [48]. Логические знания широко применяются в практике создания средств вычислительной техники, экспертных систем, автоматов, автоматических систем контроля, сигнализации и управления. Они основа теории алгоритмов. Тем не менее, с помощью логических алгоритмов можно описать не все отношения, а лишь предусмотренные законами логики и подчиняющиеся требованиям логического базиса. К сожалению, и смысловыражающие возможности логических знаний ограничены базисом и функциями алгебры логики и не всегда релевантно отображают реальную задачу [48].

Рассмотренные три подгруппы знаний принято называть теоретическими знаниями. Они составляют основу, фундамент, базис творческого процесса (определение понятия «творческий процесс» см. ниже). Их

сознательное и точное применение субъектом исключают абсолютный произвол, необузданную и необоснованную фантазию.

Следующую группу процессов — частично формализованных процессов и частично формализованных знаний — можно выделить, переходя от методов классической математики к методам дискретной математики, математической лингвистики, системного анализа, рассматривая естественно-языковые понятия тезауруса, грамматики, семантики, прагматики и исследуя знаки, а также знаковые системы. В этой еще молодой научной области получены «лингвистические знания» или *L*-знания в виде семантических моделей «смысл-текст» предложений естественного языка, моделей синтаксического анализа языков программирования и моделирования. Так, в [128] рассмотрены многоуровневая структура и неоднородность знаний с точки зрения лингвиста.

В настоящее время методы дискретной математики, математической лингвистики и системного анализа широко применяются к процессам в сложных системах, где ощущается дефицит аналитических, статистических или логических знаний. Так, например, продукционные системы, логические знания (правила «модус-поненс», «модус-толленс» и метод резолюций), а также методы поиска в пространстве состояний — основа экспертных систем, как одного из крупных достижений ИИ.

В рамках лингвистических знаний Л. Заде был создан метод математического представления и логической обработки смысла слов человека. Четкая логика была расширена до нечеткой, введено понятие «нечеткого вывода» или «приближенных рассуждений». Л. Заде утверждал, что нечеткость — это, скорее достоинство, а не недостаток моделирования.

В середине восьмидесятых годов трудами японских ученых Т. Takagi, М. Sugeno и др. выразительная сила нечетких множеств и лингвистических переменных была объединена с продукционным представлением и специальной управляющей процедурой в «нечетких системах». Скорость вычислений в нечетких системах сравнима со скоростью вычислений традиционными методами.

Еще одним видом частично-формализованных знаний следует считать «генетические знания» — *G*-знания [6, 133], биологическую информацию, как опыт, приобретенный популяцией в процессе развития в условиях постоянно изменяющейся среды в бесчисленном ряду поколений. Этот опыт закреплен в соответствующей структуре и функциях организма и находит проявление в совокупности реакций, отвечающих на возмущающие воздействия, флуктуации, как внутреннего порядка, так и среды, в которой организм функционирует и развивается [134]. С использованием *G*-знаний развиваются генетические представления в биокибернетике для решения задач совершенствования био-

производственных систем и адаптивного поиска экстремумов функций в исследовании операций. Один из методов использования генетических знаний предложил в 1975 г. Дж. Холланд (J. Holland) в виде репродукционных планов, позднее названных генетическими алгоритмами.

На частично формализованных знаниях, часто называемых эвристическими, строятся мягкие машинные вычисления и приближенные рассуждения специалиста, играющие существенную роль в интеллектуальных КСППР. Именно жесткость или мягкость вычислений и позволяют отличить классическую парадигму управления от интеллектуального управления. Тем не менее, жесткие вычисления (рис. 3.2) формируют фундамент любого управления. Мягкие вычисления вносят интеллект человека в контур управления СППР.

Еще одну группу процессов деятельности человека в СППР составляют невербализованные процессы с использованием невербализованных, образных знаний. Это «сорт» наименее изученного на практике и обобщенного в теории знания, играющего, однако, огромную роль в творчестве выработки, оценки и принятия решений, поскольку оригинальность, непредсказуемость, гениальность решений пока объясняются именно такими знаниями. Образ, по мнению Д.А. Поспелова, по своей сути непрерывен, его нельзя заменить отношениями символов и его формализация, возможно, лежит в области синергетики, термодинамики [135]. В перцептивном образе в качестве составляющих выступают ощущения равновесия, симметрии, простоты, «веса» и значимости. Образ в принципе отличается от понятия — основы символьных систем — целостностью восприятия и существованием эталонных образов и представлений. Поэтому и логика образов должна в принципе отличаться от логики формирования понятий.

Наряду с образами сейчас активно исследуются и другие виды невербальных знаний, например, знания-запахи.

Теоретические модели и эвристические правила — два типа информации, которые человек-эксперт использует для решения задач. Еще одна мощная стратегия, используемая экспертами в СППР — рассуждения на основе опыта решения задач.

Приведем несколько определений понятия «опыт».

1. Опыт — основанное на практике чувственно-эмпирическое познание действительности, единство знаний и умений, навыков [136].
2. Опыт — совокупность практически усвоенных знаний, навыков, умений, полученное в результате активного практического взаимодействия с объективным миром отражение в сознании людей законов этого мира и общественной практики [20].
3. Человеческий опыт — это чрезвычайно сложная смесь теоретических знаний, эвристик решения задач, примеров прошлых проблем и их ре-

шений, навыков восприятия и интерпретации, а также других аспектов, которые можно описать лишь как интуитивные [137].

4. Опыт — «знания в себе», лежащие на наиболее глубоком, неосознанном уровне и отвечающие за те решения эксперта, которые внешне (в том числе и для него самого) выглядят как мгновенный «инсайт», интуитивный творческий акт [138].

С годами человек-эксперт вырабатывает точные правила поведения в обычных ситуациях. Часто эти правила принимают форму прямых ассоциаций вида «случай — решение (действие)» или «ситуация — (решение) действие». Эти ассоциации могут представляться правилами «Если... то...» и, по сути, отражают в памяти человека известные ему в прошлом решения задач, еще называемые аналогиями (прецедентами).

Прецедент (лат. «предшествующий») — случай, служащий примером, оправданием для последующих случаев этого же рода [20].

Аналогия (греч. «соответствие», «сходство») — сходство предметов (явлений, процессов и т.д.) в каких-либо свойствах [139].

При умозаключении по аналогии знание, полученное из рассмотрения какого-либо объекта (модели), переносится на другой, менее изученный в каком-либо смысле объект. По отношению к конкретным объектам, заключения, полученные по аналогии, носят, вообще говоря, лишь вероятностный характер. Они являются одним из источников научных гипотез, индуктивных рассуждений. Эти решения могут быть собраны экспертами в ходе практической деятельности в конкретной предметной области. Они отображают предыдущие, удачные и неудачные попытки решения подобных задач. Так, например, программисты используют старые программы-аналоги для написания новых программ, а эксперты-конструкторы судов, при заказе верфи нового судна, прежде всего, ищут готовые проекты похожих судов.

Рассуждения с использованием опыта — еще и важный элемент, как самообучения экспертов, так и обучения менее квалифицированных работников.

Теоретические, эвристические знания и опыт — триединый базис формализованных, частично формализованных и неформализованных рутинных и творческих процессов в СППР. Особую роль в решении сложных задач, как уже подчеркивалось в разд. 2.1 и 2.2, играют творческие процессы.

Рассмотрим свойства творческого процесса [140].

Творческий процесс — это естественный, «вездесущий» и «самодостаточный» процесс. Это диалектическое взаимодействие между двумя составляющими. Первая — это случайный процесс изменения физических или интеллектуальных параметров системы в некоторых областях ее функционирования. Вторая составляющая — это детерминизация

случайного процесса управления, т.е. увеличение его плотности вероятности. Это достигается посредством обратной связи в системах управления, которая уменьшает и сдвигает области параметров в соответствии с положительными или отрицательными для системы результатами, достигаемыми этими изменениями.

Творческий процесс — это самовозникающий и саморазвивающийся процесс без внешнего создателя или управления. Он приостанавливается при достижении не наилучших, а только достаточных результатов в конкретной ситуации. Это дополнительно ускоряет адаптацию, исключая проверку других возможных альтернатив решения сложных задач и экспоненциальный «комбинаторный взрыв». Кроме того, множество достаточных решений не уступает в среднем множеству наилучших решений в экономичности расходования ресурсов системы.

Из-за своей случайной составляющей творческий процесс — необратимый, неформализуемый и невычислимый полностью процесс. Но это позволяет находить решения и приобретать знания, которые логически не следуют из знаний, известных системе управления, вырабатывать нелогичные суждения, гипотезы и аксиомы.

Творческий процесс идет на базе предыдущих организации и знаний системы управления, дополняя, корректируя их, и, таким образом, имеет еще и «генетическую» составляющую.

3.2. Моделирование теоретических знаний экспертов

Теоретические знания — это багаж, приобретенный высшим профессиональным образованием, повышением квалификации и самообразованием. Наиболее значимое место среди теоретических знаний занимают зависимости, закономерности, выработанные математикой с использованием аналитических методов. Подходы, основанные на моделях в виде аналитических зависимостей, позволяют представить процессы и явления на причинном и функциональном уровнях, а не ограничиваться рассуждениями о причинах, например, плохого урожая на основе только наблюдаемых событий и признаков.

Аналитические методы. Общеизвестно, что многие отрасли науки и техники своими успехами в значительной степени обязаны широкому использованию математических методов. Прежде всего, это относится к так называемым точным наукам — механике твердого тела, теоретической физике, квантовой механике и др. На наших глазах возникли и бурно развиваются математическая экономика, математическая биология, математическая лингвистика и многие другие науки. Что же делает математику столь универсальным и мощным инструментом исследования?

Одно из самых глубоких и точных высказываний, определяющих ее место в системе наук, принадлежит Н. Бору: «Математика — это больше, чем наука, — это язык». На первый взгляд может показаться, что в этом определении нет ничего особенного. Однако язык математики имеет одну отличительную, ставящую его в особое положение черту: над ним усилиями многих поколений математиков воздвигнуто стройное здание дедуктивных построений. Поэтому всякий раз, когда та или иная задача в любой области науки может быть сформулирована на данном языке, к услугам исследователя оказывается и определенная часть здания в виде соответствующего математического аппарата. Благодаря этому, как правило, удается сэкономить огромную массу абстрактной мыслительной работы.

Математика (от греч. *máthēma* — знание, наука) — наука о количественных соотношениях и пространственных формах действительного мира. Элементарная математика зародилась в VI—V вв. до н.э. в Греции, а новая математика переменных величин развилась в XVII в. силами И. Ньютона и Г. Лейбница, Р. Декарта, в XVIII в. — Л. Эйлера, Ж. Лагранжа и современными математиками — академиками П.Л. Чебышевым, А.М. Ляпуновым, П.Л. Стекловым, А.Н. Колмогоровым, Л.В. Канторовичем, Л.С. Понтрягиным, А.И. Мальцевым и другими учеными.

Принципиально новая страница истории математики началась с середины 50-х годов в связи с изобретением ЭВМ, значительно усиливших потенциальные возможности человека в области дедуктивных построений. Несмотря на бесспорные успехи вычислительной математики, все еще имеется существенное различие между аналитическим, формальным и численным, в виде машинной программы, решением задачи.

Аналитический (греч. *analytikos*) — относящийся к анализу, свойственный ему; применяющий анализ [141].

Для аналитического моделирования [142] характерно то, что функционирование системы записывается алгебраическими, интегродифференциальными, конечно-разностными и другими соотношениями или логическими условиями. Аналитическая модель может быть исследована широким спектром методов: аналитическими, когда получены явные зависимости между исходными характеристиками; численными, когда нет методов решения уравнений в общем виде; качественными, когда нет решения в явном виде, но можно исследовать свойства решения.

Помимо большей сжатости и наглядности формульного языка аналитических моделей по сравнению с языком произвольных алгоритмов и программ, между ними есть и два более существенных различия: 1) когда решение представлено формулой, можно дедуктивным путем вы-

водить его некоторые общие свойства; 2) формулы можно преобразовывать из одного вида в другой, например, по правилам алгебры.

Несмотря на огромное разнообразие формульных языков, наиболее часто для моделирования объекта управления с целью принятия решений в КСППР используются языки алгебры и дифференциального исчисления [1]. Решение многих задач в системах управления приводит не к одному, а к системе уравнений с несколькими неизвестными.

Очень часто в практике моделирования систем управления возникают ситуации, когда предположение о линейной зависимости различных характеристик рассматриваемого в объекте управления процесса от свойств ресурсов оказываются слишком грубым. Это делает необходимым применение нелинейных функций и систем нелинейных уравнений. Используются также и другие виды аналитических моделей — конечно-разностных, интегральных, регрессионных и других уравнений.

Таким образом, аналитические методы — это методы, отображающие реальные объекты и процессы системы S (2.1) в виде точек (безразмерных в строгих математических доказательствах), совершающих какие-либо перемещения в пространстве или взаимодействующих между собой [48], как показано на рис. 3.3.

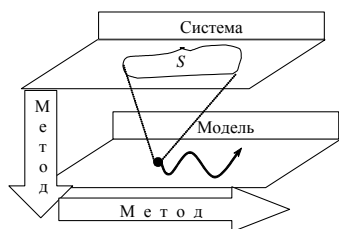


Рис. 3.3 — Аналитические методы

При этом ряд свойств многомерной, многосвязной системы или какой-либо ее части отображаются в n -мерном пространстве одной единственной точкой, совершающей какое-то движение. Это отображение осуществляется либо с помощью функции $f(S)$, либо посредством функционала $\Phi[S]$. На рис. 3.3 это обозначено вертикальной

фигурной стрелкой.

Две или более системы (их части) можно отобразить точками и рассматривать взаимодействие этих точек, каждая из которых совершает какое-то движение, имеет свое поведение. Поведение точек и их взаимодействие описывается аналитическими зависимостями, как правило, имеющими силу закона. Методы анализа аналитических зависимостей обозначены на рис. 3.3 горизонтальной фигурной стрелкой.

Основу понятийного аппарата аналитических методов составляют понятия классической математики и некоторых новых ее разделов: величина, функция, уравнение, система уравнений и т.п.

Аналитические методы применяются в тех случаях, когда свойства системы S можно отобразить детерминированными величинами, т.е.

когда знания о процессах и событиях в некотором интервале времени позволяют полностью определить поведение вне этого интервала.

На базе аналитических представлений возникли и развиваются математические теории различной сложности от аппарата классического математического анализа (методов исследования функций, поиска экстремумов функций и т.п.) до таких новых разделов математики, как математическое программирование (линейное, нелинейное, динамическое и т.п.), теории игр. Эти направления стали основой многих прикладных теорий — теории автоматического управления, исследования операций и теории принятия решений.

Достоинства и недостатки аналитических методов даны в табл. 1.12 и [1].

Поскольку аналитические методы, сами по себе, — объекты анализа при гибридизации, как прототипы (см. разд. 5.4), и синтеза ГиИС, то ниже представлены концептуальная модель (3.1) и модель вычислений (3.3) для класса аналитических методов.

Концептуальная модель для класса аналитических методов, полученная по схеме ролевых концептуальных моделей (5.12), приведена ниже (с используемыми обозначениями можно подробно ознакомиться в разд. 5):

$$\dot{m}_a^a | Te = {}^1x_2^0 \circ {}^{11}r^n \circ {}^1x_1^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{11}^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{21}^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{31}^0 \circ \quad (3.1)$$

$$\circ {}^1x_2^0 \circ {}^{13}r^n \circ {}^3x_{15}^0 \circ {}^1x_2^0 \circ {}^{18}r^n \circ {}^8\pi_a^h \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^2O_a^m,$$

$${}^8\pi_a^h = A \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2K_a^h \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2O_a^h \circ B, \quad (3.2)$$

где ${}^1x_1^0, {}^1x_2^0$ — «ресурс», «метод»; ${}^8\pi_a^h$ — схема ролевых концептуальных моделей задач, решаемых аналитическими методами; ${}^2K_a^h$ — после заполнения ролей включает, например, фазу — «нормирование», переменные — «детерминированные»; класс задач — (табл. 7.2 в [1]); переменные-отношения — «сравнения»; ${}^2O_a^h$ — специфицируется в ходе разработки; A и B — фрагменты (3.2), уточняемые в ходе разработки; ${}^2O_a^m$ — спецификатор, включающий знания о возможной погрешности решения в целом для класса методов; знания о гибридных возможностях — (табл. 8.6 в [1]); знания о преимуществах и недостатках даны в табл. 1.12 и [1]; ${}^{11}r^n$ — «быть», ${}^{12}r^n$ — «иметь свойство», ${}^{13}r^n$ — «иметь процедуру», ${}^{18}r^n$ — «предназначаться для».

Учитывая, что широкий круг аналитических моделей, используемых для решения сложных задач на практике, требует применения численных методов, то программу, выполняющую эти вычисления, можно назвать интерпретатором и предложить следующую модель вычислений с использованием аналитических зависимостей:

$$\dot{m}_a^a \Big|^{Ca} = \langle F, I^a \rangle, \quad (3.3)$$

где $F = \{f_1, \dots, f_{N_F}\}$ — множество аналитических зависимостей, записанных на языке описания моделей, принятом в некотором методе, теории; I^a — интерпретатор формул из F , т.е. компьютерная программа численного метода как процедуры получения решений на моделях из F .

В (3.3) и ниже будем различать две составляющих: декларативные описания, т.е. F , и процедурную составляющую, т.е. I^a .

Статистические методы. Данный класс инструментариев развивается в теории вероятностей, основная задача которой — разработка методов определения вероятностей некоторых событий по заданным вероятностям других событий, и математической статистике, вырабатывающей методы оценки вероятностей событий и принятия решений о характере событий на основе статистических данных [143].

Первые результаты по теории вероятностей и математической статистике принадлежат французским ученым Б. Паскалю, П. Ферма, голландскому ученому Х. Гюйгенсу, швейцарскому математику Я. Бернулли. Дальнейшее развитие этот класс методов получил в работах А. Муавра, П. Лапласа, К. Гаусса, С. Пуассона, а также русских математиков П.Л. Чебышева, А.М. Ляпунова, А.А. Маркова, а также советских математиков С.Н. Берштейна, А.Я. Хинчина и А.М. Колмогорова [143].

Если аналитические методы, рассмотренные выше, оперировали детерминированными моделями, то методы теории вероятностей и математической статистики позволяют учесть неопределенность в функционировании системы S (2.1). Это особенно важно, поскольку эксперты, решающие задачи в СППР, почти никогда не имеют доступа ко всем необходимым сведениям. Они действуют в условиях многообразия проявлений неопределенности. Различают стохастическую и лингвистическую неопределенности [67].

Стохастическая неопределенность присуща ситуациям, когда событие в реальном мире может произойти, а может и не произойти. Со временем эта неопределенность может измениться. Дополнительно событие должно отвечать условиям статистической устойчивости, т.е. сохранения частоты события в течение достаточно длительного времени повторных испытаний. Например, в высказывании: «Вероятность того, что осенью будет получен урожай озимой пшеницы 40 ц с гектара, равна

0,5» предполагается, что: 1) семена, почва, погода идеальные; 2) агротехнология идеальная; 3) из рассмотрения исключаются события получения 10, 20, 30, 50, 60 и т.д. ц с гектара.

Рассматриваемое высказывание имеет смысл только по отношению к будущему. Изменение условий во время работы СППР может изменить и высказывание. Предельные теоремы теории вероятностей предназначены для оценки идеальных условий в длинной серии испытаний, однако обеспечить идеальные условия на практике невозможно.

Таким образом «вероятностный» образ мышления эксперта может привести к некорректным рассуждениям и неверным ответам.

Еще один тип неопределенности — это лингвистическая (языковая) неопределенность. Этот тип неопределенности связан с неточностью, присущей естественному языку, на котором рассуждают люди в СППР. Например, понятию «высокий урожай» невозможно дать точное количественное определение, поскольку: 1) разные эксперты разных профессий или из разных регионов имеют свои собственные субъективные представления о том, какой урожай считать высоким; 2) если формально установить, что высокий урожай — это урожай более 50 ц с гектара, то будет ли урожай 49,99 ц с гектара высоким или нет? 3) как быть с колебаниями урожая от участка к другому участку сельскохозяйственного поля?

Подобные субъективные оценки изучаются психолингвистикой, полагающей, что люди используют слова в качестве субъективных категорий. И хотя слова точно не определены, эксперты в СППР с успехом используют их для оценок и принятия решений.

Для моделирования не количественных, а качественных, словесных рассуждений экспертов используются нечеткие множества и нечеткая логика, дающие возможность вычислений со словами (англ. computing with words) и составляющие основу методов вычислений в нечетких системах (см. разд. 3.3.3).

Следствие работы экспертов в условия стохастической и лингвистической неопределенности — проявление их уверенности (или неуверенности) в результатах решения задач.

Выбор правильной стратегии, рационального решения зависит как от важности целей, так и от вероятности того (степени уверенности в том), что они могут быть достигнуты.

Наличие или отсутствие знаний позволяет эксперту формировать высказывания только с определенной степенью уверенности (англ. degree of belief). Основным инструментом учета степени уверенности — теория вероятностей, в которой каждому высказыванию присваивается числовое значение уверенности от 0 до 1. Однако в некоторых ситуациях она не позволяет провести грань, разделяющую знание и незнание. В

этих случаях применяют рассуждения на основе фактора уверенности (см. разд. 3.4).

Методы рассуждений в теории вероятностей. Вероятности предоставляют способ суммарного учета неопределенности.

Присваивание вероятности 0 высказыванию соответствует безусловной уверенности в том, что высказывание ложно, а присваивание вероятности 1 соответствует безусловной уверенности, что высказывание истинно. Значения между 0 и 1 соответствуют промежуточным степеням уверенности в истинности высказывания. В действительности само высказывание может быть либо истинным, либо ложным независимо от этого. Важно отметить, что степень уверенности отличается от степени истинности. Вероятность 0,8 не означает «истинно на 80%», а просто указывает на 80%-ную уверенность, т.е. на довольно обоснованные ожидания. Таким образом, теория вероятностей вносит такой же онтологический вклад, как и математическая логика, т.е. позволяет указать, являются ли факты действительными в этом мире. Степени истинности, в отличие от степеней уверенности, — предмет нечеткой логики, рассматриваемой в разд. 3.3.3.

В теории вероятностей такое высказывание, как «Вероятность завершения мелиоративных работ до начала посевной кампании равна 0,8», касается степени уверенности эксперта, а не относится непосредственно к реальному миру. Эта степень уверенности зависит от результатов восприятия (свидетельств), полученных экспертом до сих пор. Поэтому во всех вероятностных рассуждениях указывается свидетельство, с учетом которого оценивается данная вероятность. При получении экспертом новых результатов восприятия его вероятностные оценки обновляются для отражения новых свидетельств. Вероятности, оцениваемые до получения свидетельства, называются априорными, безусловными, а оцениваемые после получения свидетельства — апостериорными, условными. После получения определенных свидетельств эксперт может вычислять апостериорные вероятности интересующих его результатов.

Оценка степеней уверенности применяется к высказываниям (утверждениям о том, что нечто имеет место) языка, основной элемент которого — булевы, дискретные и непрерывные случайные переменные. В дальнейшем сосредоточимся на дискретных переменных.

Атомарное событие характеризует состояние системы S (2.1), в отношении которого знания эксперта неопределенны. Оно может рассматриваться как присваивание значений всем переменным вектора состояния. Например, если таких переменных две: *дожди* и *западный ветер*, то существует всего лишь четыре атомарных события в модели

системы S эксперта-агронома, одно из которых — высказывание: *дожди* = ИСТИНА \wedge *западный ветер* = ИСТИНА».

Атомарные события имеют следующие свойства [18].

1. Они взаимно исключающие и фактически может иметь место только одно событие.
2. Множество всех возможных событий исчерпывающее, т.е. должно иметь место, по крайней мере, одно из событий. Это означает, что дизъюнкция всех атомарных событий логически эквивалентна истине.
3. Из любого атомарного события следует истинность или ложность каждого высказывания: простого или сложного. Например, из атомарного события *дожди* \wedge \neg *западный ветер* следует истинность высказывания *дожди* и ложность высказывания *дожди* \rightarrow *западный ветер*.
4. Любое высказывание эквивалентно дизъюнкции всех атомарных событий, из которых следует истинность этого высказывания. Например, событие *дожди* эквивалентно дизъюнкции атомарных событий: *дожди* \wedge *западный ветер* и *дожди* \wedge \neg *западный ветер*.

Априорная вероятность, связанная с высказыванием a , есть степень уверенности, относящаяся к этому высказыванию в отсутствии любой другой информации, и записывается как $P(a)$. Например, если априорная вероятность дождей в конце мая равна 0,2, то можно записать $P(\text{дожди} = \text{ИСТИНА}) = 0,2$ или $P(\text{дожди}) = 0,2$. Сложные высказывания образуются из простых и логических операций. Важно помнить, что вероятность $P(a)$ может использоваться только в условиях отсутствия другой информации.

После получения экспертом свидетельства-факта о неизвестных случайных переменных, важных для решаемой задачи, априорные вероятности применять уже нельзя и должны использоваться апостериорные вероятности, обозначаемые $P(a|b)$, где a и b — любые высказывания. Это обозначение читается так: «вероятность a при условии, что все, что нам известно, — это b ». Например, следующее выражение: $P(\text{дожди}|\text{западный ветер}) = 0,8$ показывает, что если наблюдаются метеоусловия при западном ветре и еще не получена какая-либо информация, то вероятность дождей составляет 0,8.

Условные вероятности могут быть определены в терминах безусловных вероятностей:

$$P(a|b) = P(a \wedge b) / P(b). \quad (3.4)$$

Это выражение остается истинным, если $P(b) > 0$, и может быть записано в виде правила произведения: $P(a \wedge b) = P(a|b) \cdot P(b)$, читаемое так: «для того, чтобы a и b были истинными, необходимо, чтобы b было истинным, и необходимо, чтобы a было истинным, если дано b ».

До сих пор был определен синтаксис высказываний, априорных и условных вероятностных утверждений о них. Определим «семантику» вероятностных утверждений, вводимую аксиомами А.М. Колмогорова: 1) все вероятности находятся в пределах от 0 до 1, и для любого высказывания a справедливо: $0 \leq P(a) \leq 1$; 2) безусловно истинные, т.е. выполнимые, высказывания имеют вероятность 1, т.е. $P(\text{ИСТИНА}) = 1$, а безусловно ложные, т.е. невыполнимые, высказывания имеют вероятность 0, т.е. $P(\text{ЛОЖЬ}) = 0$; 3) вероятность дизъюнкции высказываний равна: $P(a \vee b) = P(a) + P(b) - P(a \wedge b)$.

Обратим внимание на то, что в аксиомах А.М. Колмогорова речь идет только об априорных вероятностях, а не об условных.

Простейшие рассуждения с использованием аксиом вероятностей. Из приведенных аксиом можно вывести полезные для рассуждений факты: 1) $P(a \vee \neg a) = P(a) + P(\neg a) - P(a \wedge \neg a)$ — согласно аксиоме 3, когда $b = \neg a$; 2) $P(\text{ИСТИНА}) = P(a) + P(\neg a) - P(\text{ЛОЖЬ})$ — согласно правилу логической эквивалентности; 3) $1 = P(a) + P(\neg a)$ — согласно

аксиоме 2; 4) $\sum_{i=1}^{N_x} P(X = x_i) = 1$, т.е. вероятностное распределение по одной переменной в сумме должно составлять 1, где дискретная переменная X имеет область определения $\{x_1, x_2, \dots, x_{N_x}\}$, как обобщение п.3; 5) $P(\neg a) = 1 - P(a)$ — согласно алгебраическому определению.

Кроме этого, любое высказывание a эквивалентно дизъюнкции всех атомарных событий, в которых a истинно. Назовем эту дизъюнкцию множеством событий $SE(a)$. Поскольку атомарные события — взаимоисключающие, то вероятность любой конъюнкции атомарных событий равна нулю (аксиома 2). Таким образом, из аксиомы 3 можно вывести уравнение:

$$P(a) = \sum_{e_i \in SE(a)} P(e_i), \quad (3.5)$$

т.е. вероятность высказывания равна сумме вероятностей атомарных событий, в которых оно истинно.

Уравнение (3.5) дает простой метод вычисления вероятности любого высказывания при наличии полного совместного распределения, задающего вероятности всех атомарных событий.

Моделирование рассуждений эксперта с использованием полных совместных распределений. Рассмотрим простой метод вероятностных рассуждений (вероятностного вывода), т.е. вычисления апостериорных вероятностей для высказываний, на основании наблюдаемых

свидетельств. Полное совместное распределение при этом может рассматриваться как «база знаний» [18], из которой могут быть получены (выведены) в явном виде в ней не содержащиеся ответы.

Пусть состояние системы S экспертом-агрономом описывается тремя булевыми случайными переменными: *дожди*, *западный ветер* и *полегание злаков*, а их полное совместное распределение задано в табл. 3.2.

Таблица 3.2 — Полное совместное распределение

События	<i>западный ветер</i>		\neg <i>западный ветер</i>	
	<i>полегание злаков</i>	\neg <i>полегание злаков</i>	<i>полегание злаков</i>	\neg <i>полегание злаков</i>
<i>дожди</i>	0,2	0,1	0,07	0,3
\neg <i>дожди</i>	0,02	0,2	0,01	0,1

Сумма вероятностей в этом распределении в соответствии с аксиомами вероятностей равна 1.

Уравнение (3.5) — способ прямых вычислений вероятности любого высказывания, простого или сложного. Например, имеется шесть атомарных событий, в которых истинно высказывание *дожди* \vee *западный ветер*: $P(\text{дожди} \vee \text{западный ветер}) = 0,2 + 0,1 + 0,07 + 0,3 + 0,02 + 0,2 = 0,89$.

Часто у эксперта возникает задача извлечения из «базы знаний» распределения вероятностей по множеству переменных. Пусть он задумался о том, как часто идут дожди. Тогда, складывая элементы первой строки, получим безусловную вероятность события *дожди*: $P(\text{Дожди}) = 0,2 + 0,1 + 0,07 + 0,3 = 0,67$.

Однако в большинстве случаев эксперта интересует задача вычисления условных вероятностей переменных из вектора состояния системы S при наличии свидетельств о других переменных. Условные вероятности можно рассчитать, воспользовавшись уравнением (3.4) и табл. 3.2. Пусть эксперта интересует вероятность дождя после получения свидетельства о том, что будет западный ветер. Тогда:

$$P(\text{дожди}|\text{западный ветер}) = \frac{P(\text{дожди} \wedge \text{западный ветер})}{P(\text{западный ветер})} = \frac{0,2 + 0,1}{0,2 + 0,1 + 0,02 + 0,2} = 0,58. \quad (3.6)$$

Если эксперт в рассуждениях задумался о том, как часто при западном ветре нет дождей, то:

$$P(\neg \text{дожди}|\text{западный ветер}) = \frac{P(\neg \text{дожди} \wedge \text{западный ветер})}{P(\text{западный ветер})} = \frac{0,02 + 0,2}{0,2 + 0,1 + 0,02 + 0,2} = 0,42. \quad (3.7)$$

Из (3.6, 3.7) видно, что при вычислении вероятности $P(a/b)$, терм $1/P(\text{западный ветер})$ остается постоянным, независимо от того, какое значение вероятности дождей вычисляется. Этот терм принято называть

константой нормализации для распределения $P(\text{дожди}|\text{западный ветер})$. Она гарантирует, что полученные вероятности в сумме составят 1.

Тогда можно записать два уравнения (3.6) и (3.7) в виде одного:

$$P(\text{дожди}|\text{западный ветер}) = \alpha \cdot P(\text{дожди}|\text{западный ветер}) = \alpha \cdot (P(\text{дожди, западный ветер, полегание злаков}) + (P(\text{дожди, западный ветер, } \neg \text{полегание злаков})) = \alpha \cdot (0,2;0,02) + (0,1;0,2) = \alpha \cdot (0,3;0,22) = (0,58;0,42).$$

В работе [18] даны процедура и алгоритм вероятностного вывода, имеющие теоретическое значение. Для практических реализаций необходимы другие, более эффективные процедуры и алгоритмы.

Моделирование рассуждений эксперта с использованием правила Байеса. Выше было определено правило произведения, как иная запись (3.4). Благодаря коммутативности конъюнкции оно может быть записано в двух видах:

$$P(a \wedge b) = P(a|b) \cdot P(b), \quad P(a \wedge b) = P(b|a) \cdot P(a). \quad (3.8)$$

Приравняв две правые части и разделив их на $P(a)$, получим уравнение:

$$P(b|a) = (P(a|b)P(b))/P(a). \quad (3.9)$$

Уравнение (3.9) известно как правило Байеса или теорема Байеса и широко используется в интеллектуальных системах для моделирования вероятностных рассуждений.

Практическая значимость правила Байеса для КСППР определяется тем, что обычно эксперты располагают оценками термов в правой части правила и могут вычислить искомую оценку из левой части. Например, пусть эксперту-агроному нужно предсказать урожай на будущий год в зависимости от некоторых признаков. Ему известно, что если до начала посевной хозяйство получило денежные дотации в необходимом объеме, то можно собрать богатый урожай. Такое бывает в 60% случаев. Эксперту также известны и безусловные факты: априорная вероятность того, что богатый урожай будет убран, равна 1/10, т.е. он бывает раз в десять лет, а априорная вероятность того, что будет получена дотация, равна 1/5. Предположив, что a — высказывание, согласно которому до начала посевной получена дотация, а b — высказывание, согласно которому будет богатый урожай, получим $P(a|b) = 0,6$; $P(b)=1/10$; $P(a)=1/5$ и по (3.9) имеем: $P(b/a)=(P(a/b)P(b))/P(a)=(0,6 \times 0,1)/0,2=0,3$.

Отсюда следует предполагать, что вероятность богатого урожая равна 0,3, если единственное известное — получение дотации до посевной. Отсюда следует, что дотации позволят убирать богатый урожай раз в три года.

Один из очевидных вопросов, касающихся правила Байеса, состоит в том, почему в распоряжении эксперта может оказаться доступной условная вероятность в одном направлении, но не в другом. В проблемной области растениеводства, возможно, агроном знает, что из благоприятного финансового прогноза богатый урожай следует один раз в три года. Это означает, что эксперт располагает количественной информацией в направлении от финансирования к причинам богатого урожая. Такой эксперт не будет использовать (3.9). К сожалению, такие знания встречаются намного реже по сравнению с причинными знаниями [18]. Если случается финансовый кризис, лишаящий хозяйство возможности получить дотацию, то безусловная вероятность богатого урожая снижается. Это бывает нечасто, однако эксперт, который вывел вероятность $P(b/a)$ непосредственно из наблюдений в ходе своей «бескризисной» практики, не будет иметь представления о том, как обновить это значение, а агроном, который вычисляет $P(b/a)$ исходя из трех других значений, т.е. $P(a/b)$, $P(a)$, $P(b)$, обнаружит, что значение $P(a/b)$ должно уменьшаться пропорционально уменьшению $P(b)$. Еще более важно, что причинная информация $P(a/b)$, т.е. $P(\text{результат}|причина)$, остается незатронутой данным финансовым кризисом, поскольку она просто показывает, как влияют денежные дотации на деятельность хозяйства. Использование прямых причинных знаний, или знаний, основанных на теоретической модели, позволяет достичь надежности результатов моделирования вероятностных рассуждений эксперта в реальном мире.

Трудности применения правила Байеса возникают в рассуждениях с множеством возможных результатов (гипотез) $RE = \{re_1, re_2, \dots, re_m\}$ и множеством причин (оснований) $CA = \{ca_1, ca_2, \dots, ca_n\}$. При рассмотрении каждого результата из RE и каждой причины из CA отдельно необходимо собрать и интегрировать $m \times n$ измерений (в действительности $m \times n$ апостериорных вероятностей плюс $(m \times n)$ априорных вероятностей). В этом случае анализ становится сложнее. Эксперт должен учитывать много различных комбинаций причин. Эту ситуацию описывает правило Байеса с комплексными причинами:

$$P(b|a_1 \wedge a_2 \wedge \dots \wedge a_n) = \frac{P(b) \times P(a_1 \wedge a_2 \wedge \dots \wedge a_n|b)}{P(a_1 \wedge a_2 \wedge \dots \wedge a_n)}. \quad (3.10)$$

Для обработки одной гипотезы с одной причиной необходимо $m \times n$ измерений. Для каждой пары причин a_i и a_j и гипотезы b необходимо знать $P(a_i \wedge a_j|b)$ и $P(a_i \wedge a_j)$. Таких причин — n . Тогда число пар бу-

дет приблизительно $n \times (n-1) \approx n^2$. Если разработчик захочет использовать правило Байеса, то придется вычислить около: $(m \times n^2$ условных вероятностей) + $(n^2$ вероятных причин) + $(m$ вероятных гипотез) или собрать $(m \times n^2 + n^2 + m)$ единиц информации. Однако, возможно, некоторые пары будут независимыми, т.е. $P(a_i | a_j) = P(a_i)$.

Еще одна проблема использования сложных байесовских систем рассуждений — необходимость переопределения таблицы вероятностей, в частности табл. 3.2, при выявлении новых взаимосвязей между гипотезами и причинами. Для корректности заключений байесовских рассуждений требуется пересчет полных вероятностей, включая объединенные вероятности. Во многих прикладных областях такой обширный сбор и верификация данных невозможны или стоят очень дорого. Однако эти сложности преодолимы, например, байесовскими сетями доверия [18].

Время и неопределенность в рассуждениях экспертов. Выше был рассмотрен один из методов моделирования вероятностных рассуждений. Главное ограничение состояло в том, что система S представляется такому эксперту в статике, где каждая случайная переменная имеет одно фиксированное значение. Например, в ходе прогнозирования урожая на будущий год предполагалось, что эксперт в составе СППР решает эту задачу один раз, например, в октябре, после сбора урожая текущего года. При этом результаты решения задачи оставались неизменными до момента начала реализации плана работы на будущий год. Предположим, до марта будущего года. С момента принятия плана действий истечет четыре месяца (ноябрь — декабрь). Произойдут ли за это время события, способные изменить принятую и зафиксированную в плане стратегию поведения агрофирмы? Вполне вероятно. Как действовать СППР в подобных условиях? Казалось, что ответ напрашивается сам собой: провести заседания в ноябре, декабре, январе, феврале, всякий раз корректируя или оставляя неизменной избираемую на предыдущем заседании стратегию. Так обычно и поступает ЛПР, добавляя тем самым категорию «время» в процесс решения задачи. Таким образом, сложные стохастические, динамические процессы в био-производственной системе заменены процессом многократного использования во времени модели системы S в статике. В таком подходе есть очевидные проблемы: 1) нет никакой информации, которая указывала бы на регулярность таких заседаний; хотя можно заседать всякий раз, по мере появления в системе и внешней среде новых, имеющих важное значение, событий; 2) трудности со сбором и обработкой информации, необходимой для

принятия решений (см. байесовские рассуждения); эти трудности будут трансформировать понятие «желаемая» регулярность заседаний в понятие «возможная» регулярность заседаний; 3) для принятия решений становятся существенными не только причинно-следственные связи в статичном мире, но и динамические аспекты задачи, например, динамика изменения продаж полученного урожая, состояние машинного парка агрофирмы, оценка изменений динамики зимовки растений (для озимых сортов), динамика закупки семян, удобрений и др. Чтобы оценить текущее состояние био-производственной системы по хронологии накопленных фактов и предсказать результаты финансовых операций и агротехнологических действий, необходимо моделировать эти изменения.

Как можно моделировать подобные динамические ситуации в КСППР? Процесс изменений может рассматриваться как последовательность снимков — состояний, временных срезов системы S в данный конкретный момент времени. Каждый такой снимок содержит множество случайных переменных, как наблюдаемых, так и нет. Будем полагать, что во временном срезе наблюдается одно и то же множество переменных. Интервал времени между срезами зависит от задачи и остается фиксированным во время моделирования, что позволяет исчислять модельное время, начиная с нуля, и обозначать его целыми числами. Вводится ряд предположений об отличительных свойствах модельного времени и ограничений на характер процессов поведения системы: 1) изменения в состоянии системы вызваны стационарным процессом [18], т.е. изменения подчиняются законам, которые сами не меняются во времени; 2) процесс изменения носит марковский характер [18], т.е. текущее состояние зависит только от конечной истории предыдущих.

В условиях выполнения этих предположений существуют различные методы моделирования вероятностных динамических рассуждений. Один из них — динамические байесовские сети (англ. dynamic Bayesian network — DBN), рассмотрен, например в [18]. Реляционные вероятностные модели (англ. relational probability model — RPM) [18]. Другой — имитационное статистическое моделирование (метод Монте-Карло) [51, 144, 145]. Этот метод универсален, широко и успешно применяется для решения задач в системах массового обслуживания. Разработка и эксперименты с моделями ИСМ автоматизированы во многих языках моделирования, в частности в GPSS.

Суть метода состоит в построении искусственного вероятностного процесса, параметры которого давали бы решение поставленной задачи и включали бы следующие этапы: 1) для отдельных параметров процесса выбираются их случайные значения по заданным законам распределения; 2) случайному процессу ставится в соответствие детерминированный процесс, в котором используются полученные на первом этапе

значения параметров и определяется результат процесса; 3) первые два этапа многократно повторяются, и каждый раз фиксируется результат; 4) на основании статистической обработки зафиксированных результатов отдельных реализаций определяются вероятностные характеристики случайного процесса.

Таким образом, статистические методы, примеры которых рассмотрены выше, отображают случайные, стохастические явления, процессы, события и поведение системы S статистическими характеристиками и статистическими закономерностями (рис. 3.4).

Статистические отображения системы S в модель представляются уже областью (она показана на рис. 3.4 окружностью с черной точкой внутри), границы которой специфицированы с некоторой вероятностью. Эта область может быть задана набором из N статистических распределений величин, характеризующих свойства и поведение системы S . Обычно для моделирования используют не законы распределения случайных величин, а их моменты — математическое ожидание и дисперсию. Это позволяет заменить исследование поведения области изучением поведения точки по аналогии с аналитическими моделями.

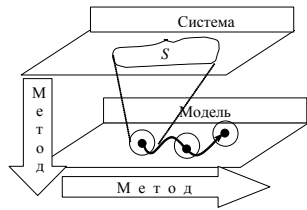


Рис. 3.3 — Статистические методы

Основа понятийного аппарата статистических методов — термины теории вероятностей и математической статистики: случайная величина, вероятность события, закон распределения, математическое ожидание, среднее квадратическое отклонение, выборка и др.

На базе статистических представлений развиваются прикладные теории статистических решений. Эти представления используются в задачах массового обслуживания, оптимизации, прогнозирования в управлении и проектировании, оценки альтернатив и др.

Достоинства и недостатки ИСМ даны в табл. 1.12 и [1].

Концептуальная модель для класса методов рассуждений экспертов с учетом неопределенности и времени применительно (метод Монте-Карло) к системам массового обслуживания с использованием языка моделирования GPSS, ориентированного на события, приведена ниже:

$$\dot{m}_s^a | Te = {}^1x_2^0 \circ {}^{11}r^n \circ {}^1x_1^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{12}^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{22}^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{32}^0 \circ {}^1x_2^0 \circ {}^{13}r^n \circ {}^3x_{16}^0 \circ {}^1x_2^0 \circ {}^{18}r^n \circ {}^8\pi_s^h \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^2O_s^m, \quad (3.11)$$

$${}^8\pi_s^h = A \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2K_s^h \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2O_s^h \circ B,$$

где ${}^8\pi_a^h$ — схема ролевых концептуальных моделей задач, решаемых имитационным статистическим моделированием; ${}^2K_s^h$ — включает фазу — «прогнозирование», переменные — «детерминированные», «стохастические»; класс задач — (табл. 7.2 в [1]); переменные-отношения — «сравнения», «времени»; ${}^2O_s^h$ — специфицируется в ходе разработки; A и B — фрагменты (3.11), уточняемые в ходе разработки; ${}^2O_s^m$ — спецификатор, включающий: знания о возможной погрешности решения в целом для класса методов — $0,01 \div 0,05$ от \max значения моделируемой величины; знания о гибридных возможностях — (табл. 8.6 в [1]); знания о преимуществах и недостатках даны в табл. 1.12 и [1].

Модель вычислений для методов ИСМ систем массового обслуживания на языке моделирования GPSS, ориентированного на события, приведена ниже:

$$\dot{m}_s^a|Ca = \langle DB, M_b, M_e, I^s \rangle, \quad (3.12)$$

где $DB = \{db_1, \dots, db_{N_{DB}}\}$ — база транзактов; M_b — блочно-сетевая модель СМО; M_e — событийно-временная модель СМО; I^s — интерпретатор.

Транзакт — носитель информации, перемещаемой от блока к блоку. Блочно-сетевая модель M_b — смешанный граф с вершинами трех видов: истоков $ИС$, стоков $СТ$ и преобразователей $ПП$ транзактов. Заданы также функции $F = \{F^{ИС}, F^{СТ}, F^{ПП}\}$ — функционирования истоков, стоков и преобразователей соответственно и связи между блоками R_b . Тогда: $M_b = \langle ИС, СТ, ПП, F, R_b \rangle$. Модель M_e включает два основных списка транзактов: Ls_1 — перемещающихся от блока к блоку в данный момент модельного времени, Ls_2 , где транзакты упорядочены по возрастанию времени движения и модель времени — T . Тогда: $M_e = \langle Ls_1, Ls_2, T \rangle$. Интерпретатор I^s работает с DB, M_b, M_e и может быть определен совокупностью процессов: $I^s = \langle I^{S1}, I^{S2}, I^{S3}, I^{S4} \rangle$, где I^{S1} — инициализация DB, M_b, M_e ; I^{S2} — коррекция значения таймера T , т.е. таймер устанавливается на время движения первого в списке Ls_2 транзакта, который переводится в Ls_1 ; I^{S3} — просмотр Ls_1 и имитация движения транзактов из Ls_1 по M_e и выполнение функций F ,

ввод транзактов из DB через истоки и вывод в DB через стоки; I^{S4} — сбор, обработка и вывод статистической информации о блоках и транзактах, которая и рассматривается как результаты испытаний. Работа интерпретатора I^S завершается, когда модельное время достигнет априори заданного значения. Таким образом, работу интерпретатора I^S можно рассматривать как последовательное выполнение рассмотренных выше процессов:

$$\begin{array}{c}
 \downarrow \text{-----} \downarrow \\
 I^{S1} \rightarrow I^{S2} \rightarrow I^{S3} \rightarrow I^{S4}.
 \end{array} \quad (3.13)$$

3.3 Моделирование профессиональных знаний экспертов

3.3.1 Методы рассуждений в экспертных системах

В 1970-х гг. группа ученых, возглавляемая Э. Фейгенбаумом (E. Feigenbaum) из Стэнфордского университета (США), начала искать средства преодоления недостатков общего решателя задач GPS, работавшего в мире математических головоломок с относительно небольшим множеством состояний и хорошо известными формальными правилами. Вместо поиска эффективных универсальных эвристик они занялись сужением проблемной области и исследованием знаний и умений специалиста. На этом пути и родились экспертные системы. Они представляют в ИИ школу символизма, базирующуюся на двух гипотезах ее основателей А. Ньюэлла (A. Newell) и Г. Саймона (H. Simon), утверждавших, что символьные системы имеют необходимые и достаточные средства для интеллектуальных действий и, что решения задач могут быть представлены в виде символьных структур. Символьные системы решают задачи с помощью поиска, т.е. генерируют потенциальные решения и постепенно модифицируют их, пока они не будут удовлетворять условиям решения [29]. Таким образом, экспертные системы моделируют высокоуровневые процессы познания и естественноязыковую обработку информации экспертами.

Значительный вклад в развитие методов рассуждений в экспертных системах внесли российские ученые Э.В. Попов, В.Ф. Хорошевский, Т.А. Гаврилова и др.

Экспертная система — система ИИ, в которой в частично формализованном виде накапливаются знания экспертов-специалистов из соответствующей предметной области и имеются правила использования этих знаний для решения конкретных задач [146]. Э. Фейгенбаум под-

черкивал, что экспертные системы должны быть сильны знаниями, даже если они слабы методами. Экспертные системы решают неформализованные задачи на уровне хорошего специалиста. Они способны пояснить, каким образом получены те или иные решения.

На рис. 3.5 изображены четыре основных элемента статической продукционной экспертной системы: база фактов, база знаний, интерпретатор и средства объяснений.

База фактов содержит символическое описание ресурсов предметной области, в которой работает эксперт. Объектно-ориентированный подход требует представления этих описаний с использованием классов, подклассов объектов, объектов, слот, в которых заданы количественные или качественные значения свойств объектов. Поскольку интерпретатор рассматривает базу фактов как объект своей работы, т.е. преобразования с использованием правил базы знаний, то базу фактов принято называть рабочей памятью, подчеркивая тем самым ее участие в процессе обработки информации.

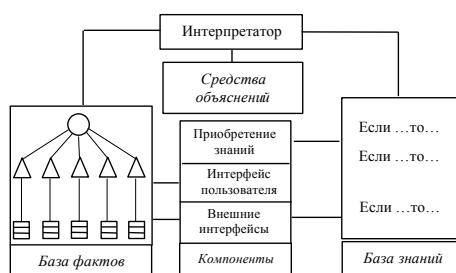


Рис. 3.5 — Структура статической, продукционной экспертной системы

База профессиональных знаний предназначена для хранения правил «Если ... то ...». Она характеризуется: размером, полнотой, непротиворечивостью, эффективностью обслуживания и другими параметрами. Размер БЗ лежит в пределах, примерно, от 50 до 1000 правил.

Интерпретатор (машина вывода) использует в качестве

исходных данных базу фактов, базу знаний и формирует такую последовательность правил, что, будучи примененной к базе фактов, она приводит к решению задачи. В некотором смысле он имитирует работу эксперта, рассуждающего над задачей.

Средства объяснений — еще один неотъемлемый элемент ЭС, предназначенный для увеличения у пользователя степени доверия к советам системы при решении задачи. Иными словами, ЭС может сделать свои рассуждения прозрачными, контролируруемыми и оцениваемыми по качеству пользователем. Компонента объяснений может предложить и другие возможности. Например, показать причину изменения значения свойства объекта (т.е. значения слота) или выдать любые комментарии, априори введенные инженером по знаниям в правила.

Прочие, приведенные на рис. 3.5, компоненты, обеспечивают взаимодействие с экспертом или (и) другими источниками знаний [98], ин-

терфейс пользователя и внешние интерфейсы [29, 148].

Рассмотрим рассуждения в экспертной системе при решении экспертом задачи в СППР транспортного узла типа «морской порт». Ее суть состоит в следующем. Судно находится близко от порта (на подходе) и просит выделить причал для обслуживания. Таких причалов — два. Один из них (причал 1) специализируется на обработке больших морозильных траулеров. Оба причала, т.е. причал 1 и причал 2, могут обслуживать и обычные траулеры. Если крупные суда требуют специализированного причала, а он оказывается занятым, то возможны перешвартовки. Эксперту необходимо запланировать причал для вновь прибывающего судна.

Пусть из эксперта извлечены профессиональные знания, которые включены в виде продукций (англ. productions) или правил (англ. rules) в базу знаний:

1. Если *состояние судна = на подходе* и *тип судна = большой морозильный траулер* и *состояние причала 1 = свободен* то *обработка на причале 1 = большой морозильный траулер* и *состояние судна = стоянка* (8).
2. Если *состояние судна = на подходе* и *тип судна = траулер* и *состояние причала 1 = свободен* то *обработка на причале 1 = траулер* и *состояние судна = стоянка* (7).
3. Если *состояние судна = на подходе* и *тип судна = траулер* и *состояние причала 1 = занят* и *состояние причала 2 = свободен* то *обработка на причале 2 = траулер* и *состояние судна = стоянка* (6).
4. Если *состояние судна = на подходе* и *тип судна = большой морозильный траулер* и *состояние причала 1 = занят* и *обработка на причале 1 = траулер* и *состояние причала 2 = свободен* то *перешвартовка = траулер* (4).
5. Если *состояние судна = на подходе* и *обработка на причале 1 = большой морозильный траулер* или *обработка на причале 1 = траулер* то *состояние причала 1 = занят* (5).
6. Если *перешвартовка = траулер* то *причал 1 = свободен* (3).
7. Если *перешвартовка = траулер* и *причал 1 = свободен* то *причал 2 = занят* и *обработка на причале 1 = нет* (2).
8. Если *перешвартовка = траулер* и *причал 1 = свободен* и *причал 1 = свободен* то *обработка на причале 2 = траулер* и *перешвартовка = нет* (1).

Как видно, таких правил восемь. Каждое из них — это пара «условие — действие», определяющее одну порцию информации, необходимой для решения задачи. В конце правила в круглых скобках указан приоритет. В «условии» перечисляется совокупность фактов (обстоятельств, свидетельств, посылок), важных для решения задачи, а «действие» часто трактуется как результат (цель, гипотеза, заключение). Условная, левая часть правила — это образец, шаблон, который определяет, когда данное правило может быть применено на каком-либо этапе решения задачи. Для этого образец сопоставляется с описанием текущей ситуации в системе S , хранящимся в рабочей памяти (англ. working memory). Если условие некоторого правила соответствует содержанию рабочей

памяти, то правило считается готовым к применению и помещается в специальную память, называемую «планом решения» (англ. agenda) или «списком правил» (англ. rule's list). Понятие плана решения имеет глубокую и обоснованную психологами и специалистами в области искусственного интеллекта природу, поскольку было установлено, что человек, решая задачи, обязательно разрабатывает подобный план. По сути, применительно к рассуждениям на основе профессиональных знаний это означает, что эксперт устанавливает некоторый порядок на множестве этапов решения задачи.

Поскольку в план решения может быть помещено несколько подходящих правил, то возможен конфликт правил за «право» быть использованным в дальнейших рассуждениях. Для разрешения конфликтов используются специальные методы [1, 29].

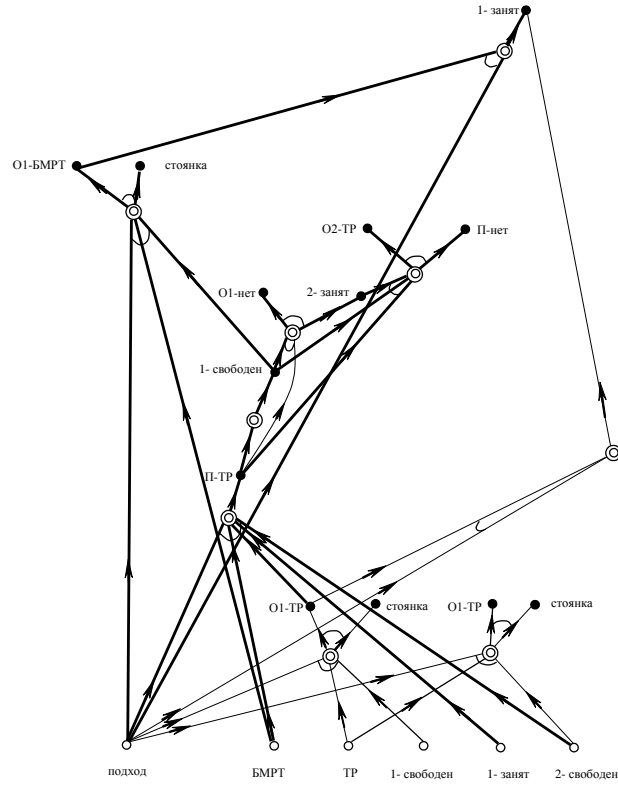
Предположим, что в нашей задаче эксперт для разрешения конфликта правил руководствуется приоритетом правил: преимущество получает правило с меньшим значением приоритета. Обратим внимание, что на практике установление приоритета правил — не простая задача и может привести к внесению «жесткого», алгоритмического управления в продукционную систему, что не желательно.

В работе [149] А. Ньюэлл и Г. Саймон приводят результаты моделирования посредством продукционной системы таких важных аспектов и навыков человека при решении задач, как фокусирование внимания, долгосрочная и краткосрочная память. Так, текущий фокус внимания был описан как текущее состояние системы S и правило, которое изменяет это состояние. Продукции соответствуют навыкам решения задач в долгосрочной памяти. Рабочая память продукционной системы соответствует краткосрочной памяти эксперта (текущей области его внимания). Содержание рабочей памяти после завершения решения задачи не сохраняется.

Рассуждения в экспертных системах организуются различными методами работы интерпретатора, использующими цикл шагов «распознавание — действие». В начале рассуждений в рабочую память заносится исходное состояние для решения задачи. Эта информация сопоставляется с образцами всех правил, имеющихся в базе знаний, либо со всеми образцами из некоторой части базы знаний, называемой «островком знаний» или подмножеством правил. В итоге сопоставления определяется конфликтное множество (англ. conflict set) правил. Эти правила готовы к применению (англ. fired rules), но только после разрешения конфликта. Правило, «победившее» в конфликте, активируется, т.е. выполняются действия из его правой части. При этом изменяется содержимое рабочей памяти. Условия завершения рассуждений могут быть самыми различными: достижение специально определяемой цели; несо-

ответствие содержимого рабочей памяти ни одной условной части правил из базы знаний (план решения пуст) и др.

Для моделирования рассуждений эксперта методами экспертных систем используется представление базы профессиональных знаний эксперта графом И/ИЛИ [29] — расширением понятия графа специфическими элементами (помимо вершин и ребер (дуг)) для отображения логических связей между фактами в левых частях правил (рис. 3.6).



Обозначения: ○ — исходный факт, ● — добавленный факт, ⊙ — левая часть правила, подход, стоянка — факты *состояние судна* = на подходе и *состояние судна* = стоянка; БМРТ, ТР — факты *тип судна* = БМРТ и *тип судна* = ТР; 1- свободен, 1- занят, 2- свободен, 2- занят — факты *состояние причала 1* = свободен, *состояние причала 1* = занят, *состояние причала 2* = свободен, *состояние причала 2* = занят; О1-БМРТ, О1-ТР, О2-ТР — факты *обработка на причале 1* = БМРТ, *обработка на причале 1* = ТР, *обработка на причале 2* = ТР; П-ТР, П-нет — факты *перешвартовка* = траулер, *перешвартовка* = нет

Рис. 3.6 — Граф И/ИЛИ рассуждений эксперта при решении задачи выделения места стоянки для прибывающего в порт судна

Кривая, соединяющая дуги графа (дужка), указывает, что в рассуждениях должны быть истинными несколько фактов для заходящих в вершину дуг или что гипотезы и факты в правых частях правил становятся истинными (для исходящих дуг). Отсутствие дужки, соединяющей дуги графа, указывает на альтернативность, наличие связки ИЛИ в рассуждениях эксперта.

Моделировать рассуждения эксперта на графе И/ИЛИ, т.е. выполнять поиск решающего подграфа [29], можно различными методами, однако чаще всего — это поиск в прямом направлении снизу-вверх (рассуждения на основе данных, прямой вывод, прямая цепочка — англ. *data-driven search, forward chaining*) и поиск в обратном направлении сверху—вниз (рассуждения на основе цели, поиск от цели, обратный вывод, обратная цепочка — англ. *goal-driven search, backward chaining*).

Рассуждения на основе данных. Эти рассуждения рекомендуются [18] использовать, когда: все или большинство исходных данных задано в постановке задачи; существует большое число гипотез-целей, но только незначительная часть комбинаций фактов позволяет сфокусировать рассуждения; формулировка целей затруднительна.

Рассмотрим механизм моделирования рассуждений на основе данных для планирования причала вновь прибывающему судну. Для этого обратимся к рис. 3.7, на котором рассуждения эксперта отображены как последовательность шагов цикла «распознавание — действие», в процессе которых интерпретатор выполняет сопоставление с образцом, разрешает конфликты и изменяет содержимое рабочей памяти.

Здесь введены следующие обозначения: жирным шрифтом в рабочей памяти обозначены факты, добавляемые в результате выполнения правила на предыдущем шаге; в плане решения через запятую перечислены номера правил и в скобках — приоритеты. Из рис. 3.7 видно, что в ходе рассуждений была найдена цепочка правил: *Правило 4* → *Правило 6* → *Правило 7* → *Правило 8* → *Правило 1* → *Правило 5*, изображенная жирными линиями на графе И/ИЛИ (рис. 3.6) в виде решающего подграфа.

Механизм рассуждений на основе данных обеспечивает возможность прозрачности результатов моделирования за счет предоставления пользователю объяснений. Обычно экспертные системы, использующие продукционные базы знаний, могут отвечать на два вопроса: «Почему?» и «Как?».

Рассуждения на основе цели. Эти рассуждения рекомендуются [137] использовать, когда: цель поиска (гипотеза) присутствует в постановке

задачи или легко формулируется; среди извлеченных из эксперта правил много таких правил, которые продуцируют возрастающее число целей (заклучений), что позволяет эффективно отсеивать ветви на графах И/ИЛИ; исходные данные не участвуют в постановке задачи, но должны быть известны эксперту, и он может использовать рассуждения от цели для правильной постановки задачи.

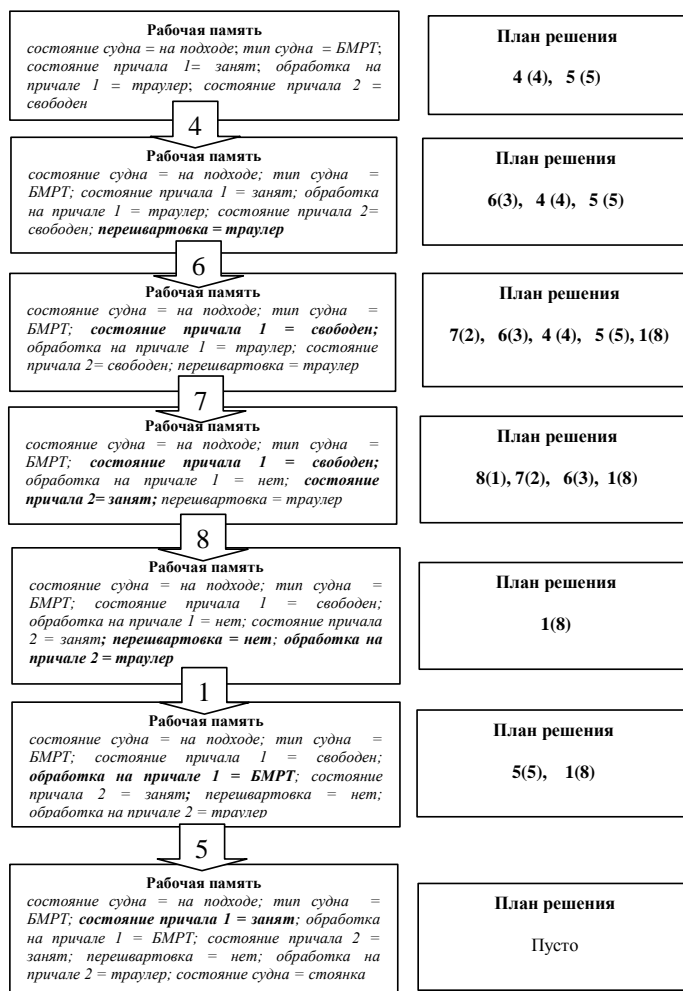


Рис. 3.7 — Рассуждения на основе данных как процесс преобразования информации в рабочей памяти

Для работы с базой знаний из восьми рассмотренных выше правил, правые части всех правил трактуются как цель, которая должна быть достигнута экспертом в ходе рассуждений при решении задачи. Предположим, что такая цель — «занять причал 1», т.е. предоставить его для стоянки вновь прибывающего в порт судна.

Рассуждения в обратном направлении начинаются с того, что исходная цель заносится в рабочую память (рис. 3.8).

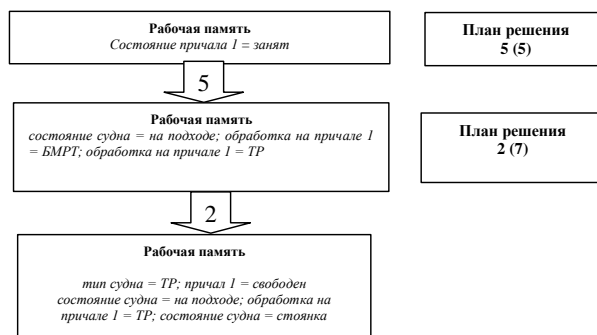


Рис. 3.8 — Рассуждения на основе цели как процесс преобразования информации в рабочей памяти

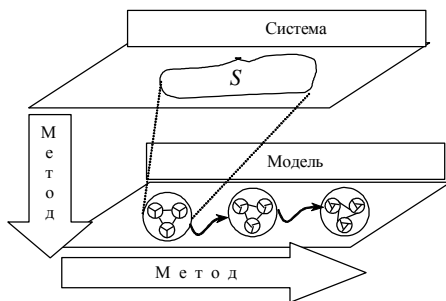
Теперь в качестве образца используются правые части правил из базы знаний, с которыми сопоставляется высказывание, находящееся в рабочей памяти. В нашем примере планирования причала вновь прибывающему судну с выражением из рабочей памяти успешно сопоставится одно правило — это правило 5. Тогда факты из его левой части помещаются в рабочую память. Тем самым экспертная система в ходе рассуждений выдвигает две гипотезы, следствием осуществления которых могло явиться занятие первого причала: 1) *состояние судна = на подходе* и *обработка на причале 1 = БМРТ* или 2) *состояние судна = на подходе* и *обработка на причале 1 = ТР*. Иными словами, к занятию причала 1 могло привести то, что какое-то судно — *на подходе* и начнется его обработка. При этом эксперту выдаются запросы, он должен оценить гипотезы и подтвердить или опровергнуть их истинность. Пусть он ввел в систему в качестве своего ответа: (*состояние судна = на подходе*) = ЛОЖЬ, (*обработка на причале 1 = БМРТ*) = ЛОЖЬ, (*обработка на причале 1 = ТР*) = ИСТИНА. В этом случае факт (*обработка на причале 1 = ТР*) получает статус цели, и система начинает рассуждения по возможности ее достижения, т.е. ищет в базе знаний правила, у которых правая часть соответствует цели (*обработка на причале 1 = ТР*). Это

будет одно правило — правило 2. Поскольку в правиле 2 в правой части две гипотезы, соединенные конъюнкцией, то автоматически, без запроса у пользователя, в рабочую память размещается и *состояние судна = стоянка*, причем (*состояние судна = стоянка*) = ИСТИНА. Поскольку истинность целей в правой части правила 2 подтвердилась, правило 2 активируется, в рабочую память размещаются гипотезы *тип судна = ТР*; *причал 1 = свободен*; *состояние судна = на подходе* и эксперту задаются вопросы об их истинности. Пусть все три гипотезы истинны. То есть причина того, что на первом причале стоит и обрабатывается судно — его запрос на обслуживание (*состояние судна = на подходе*), его тип — траулер и то, что причал 1 оказался свободен. Все гипотезы в рабочей памяти получили значение ИСТИНА, что является признаком окончания рассуждений в обратном направлении.

Из рассмотренного примера видно, что в рассуждениях от цели подходящие правила используются, чтобы исключить неперспективные ветви при поиске на графе И/ИЛИ, что значительно эффективнее, чем «нефокусированные» рассуждения на основе данных.

Обобщая сказанное о методах рассуждений с базами профессиональных знаний, отметим, что экспертные системы [1, 29, 98, 123, 124, 146, 150, 151] используют лингвистические (знаковые) представления знаний о системе S совместно с логическими методами преобразования знаковых систем. Чтобы подчеркнуть специфичность такого подхода, введены понятия логико-лингвистических [152] или символьных [153] моделей и методов.

На рис. 3.9 показано, что лингвистические представления отображают систему S в знаковую систему. Она показана окружностью с расположенными внутри тремя знаками, соединенными отношениями. Каждый знак-окружность разделен на три сектора, обозначающих синтаксис, семантику и прагматику. В ходе рассуждений над решаемой задачей знаковая система преобразуется из соответствующей постановке задачи в соответствующую решению задачи.



Основу понятийного аппарата экспертных систем составляют как термины лингвистических представлений (тезаурус, грамматика) и семиотических представлений (знак, знаковая система, знаковая ситуация), так и специфические понятия символа, символической и продукцион-

134 Рис. 3.9 — Логико-лингвистические методы

ной системы, базы знаний и др.

Экспертные системы решают неформализованные и плохо структурированные задачи в управлении и проектировании. Достоинства и недостатки экспертных систем даны в табл. 1.12 и [1].

Концептуальная модель для класса методов экспертных систем приведена ниже:

$$\begin{aligned} \dot{m}_e^a | T^e &= {}^1x_2^0 \text{ } {}^{11}r^n \text{ } {}^1x_1^0 \circ {}^1x_2^0 \text{ } {}^{12}r^n \text{ } {}^{22}x_{13}^0 \circ {}^1x_2^0 \text{ } {}^{12}r^n \text{ } {}^{22}x_{24}^0 \circ {}^1x_2^0 \text{ } {}^{12}r^n \text{ } {}^{22}x_{33}^0 \circ \\ &\circ {}^1x_2^0 \text{ } {}^{13}r^n \text{ } {}^3x_{12}^0 \circ {}^1x_2^0 \text{ } {}^{18}r^n \text{ } {}^8\pi_e^h \circ {}^1x_2^0 \text{ } {}^{12}r^n \text{ } {}^2O_e^m, \\ {}^8\pi_e^h &= A \circ {}^8x_1^0 \text{ } {}^{82}R^n \text{ } {}^2K_e^h \circ {}^8x_1^0 \text{ } {}^{82}R^n \text{ } {}^2O_e^h \circ B, \end{aligned} \quad (3.14)$$

где ${}^8\pi_e^h$ — схема ролевых концептуальных моделей задач, решаемых экспертными системами; ${}^2K_e^h$ — включает фазу — «контроль», «анализ», «прогнозирование», переменные — «детерминированные», «четкие лингвистические»; класс задач — (см. табл. 7.2 в [1]); переменные-отношения — «сравнения», «причинности»; ${}^2O_e^h$ — специфицируется в ходе разработки; A и B — фрагменты (3.14), уточняемые в ходе разработки; ${}^2O_e^m$ — спецификатор, включающий: знания о возможной погрешности решения в целом для класса методов — отсутствуют; знания о гибридных возможностях — (см. табл. 8.6 в [1]).

Модель вычислений в экспертных системах приведена ниже:

$$\dot{m}_e^a | C^a = \langle KB, FB, RB, I^e, R^e \rangle, \quad (3.15)$$

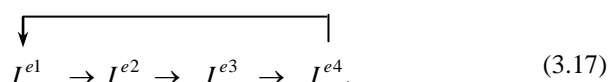
где KB — база знаний, как совокупность символьных правил-продукций, имитирующих минимально возможную конструкцию точных, детерминированных рассуждений человека в форме «условие-действие»; FB — база фактов, как совокупность связанных или несвязанных друг с другом символьных выражений, задающих декларативную, фактографическую информацию о системе S (ресурсах для решения задач и их свойствах, а также значениях этих свойств), необходимую и достаточную для решения задачи; RB — база выводов, формируемая I^e в ходе работы, содержащая информацию о причинах изменений в RB и комментарии, внесенные экспертом в KB для объяснений; I^e — интерпретатор; R^e — системообразующие отношения.

Интерпретатор I^e — это четверка процессов [29]:

$$I^e = \langle I^{e1}, I^{e2}, I^{e3}, I^{e4} \rangle, \quad (3.16)_{35}$$

где I^{e1} — процесс выбора из KB подмножества активных правил; I^{e2} — процесс сопоставления с образцом для активных правил; I^{e3} — процесс разрешения конфликтов правил; I^{e4} — процесс выполнения правила, т.е. использование его в рассуждениях.

Работу интерпретатора I^e можно представить как последовательное, циклическое выполнение рассмотренных выше процессов:



3.3.2 Методы вычислений в нечетких системах

Еще Аристотель утверждал, что если говорить о незнающих и знающих, то первые полагают, что так обстоит дело, а знающие и знают, что так обстоит дело [125]. Если выше шла речь о знании экспертов в СППР, то настало время поговорить и о «незнании».

Незнание следует отличать от неопределенности. Рассмотрим подбрасывание монеты. Мы знаем, что если монета подлинная, то вполне резонно считать, что вероятность выпадения орла равна 0,5. Если же известно, что монета поддельная и неизвестно, есть ли на ней два орла или две решки, то по-прежнему единственным разумным решением является принятие гипотезы о вероятности 0,5. Оба случая неодинаковы, но теория вероятностей, по-видимому, не позволяет провести между ними различие [67].

Рассуждения с использованием фактора уверенности. Для того, чтобы измерить незнание в рассуждениях экспертов, используется мера уверенности (или доверия). Это неформальная оценка, которую эксперт добавляет к своему заключению, например, «вероятно, это так», «почти наверняка», «это так», «это совершенно невероятно». Такая мера предложена в теории фактора уверенности [18, 137]. Она вносит простые предположения о мере доверия, предлагает методы «работы» с фактами в левых частях правил экспертных систем и показывает, как поступать, если два и более правил имеют одинаковые правые части.

Первое теоретическое допущение — разделение меры доверия и недоверия («за» и «против») для каждого отношения «причина-гипотеза», записанного правилом «Если ... то ...».

Пусть $M_y(b|a)$ — мера уверенности эксперта в гипотезе b при данном свидетельстве a , $M_n(b|a)$ — мера сомнения эксперта в гипотезе b при заданном свидетельстве a . Тогда: $1 > M_y(b|a) > 0$, если $M_n(b|a) = 0$, или $1 > M_n(b|a) > 0$, если $M_y(b|a) = 0$. Эти две меры накладывают ограничения друг на друга, так как заданной считается часть свидетельства в пользу данной гипотезы либо против нее.

В этом состоит различие между теорией фактора уверенности и теорией вероятности. Меры уверенности и сомнения эксперта можно объединить в формуле фактора уверенности (англ. certainty factor) следующим образом: $F_y(b|a) = M_y(b|a) - M_n(b|a)$. Приближение значения фактора уверенности к 1 означает, что у эксперта усиливается доверие к гипотезе, а приближение к (-1) — означает, что эксперт все больше сомневается в ее истинности. Близость значения $F_y(b|a)$ к нулю означает, что у эксперта мало доказательств, как в пользу гипотезы, так и против нее, либо есть баланс доказательств «за» и «против».

Когда эксперты принимают участие в разработке базы знаний, т.е. предоставляют для этого свои профессиональные знания, вербализованные и выраженные в виде правил «Если...то...», то они в большинстве случаев могут сопоставить с каждым правилом некоторое значение фактора уверенности $F_y(b|a)$. Рассмотрим это вопрос подробнее.

Пусть в базе знаний экспертной системы имеется правило: «Если (p_1 И p_2) ИЛИ p_3 то d_1 И d_2 », где p_1, p_2, p_3 — факты (свидетельства, условия), а d_1 и d_2 — действия (гипотезы).

Например, «Если (*лето жаркое* = ИСТИНА И *рост посевов слабый* = ИСТИНА) ИЛИ *мало влаги в почве* = ИСТИНА то *искусственный полив* = ИСТИНА И *выполнить подкормку растений* = ИСТИНА.

Фактор уверенности связывается в правиле как с фактами (свидетельствами в левой части), так и с действиями-гипотезами в правой части. Пусть эксперт в ходе разработки правила установил следующие значения фактора уверенности: «Если (p_1 И p_2) ИЛИ p_3 то d_1 (0,8) И d_2 (0,3)». Это означает, что он в большей степени уверен (0,8), что в условиях жаркого лета и слабого роста посевов необходимо применить искусственный полив и не совсем уверен (0,3), что в этих обстоятельствах нужно подкормить растения.

Возможна и другая ситуации при разработке базы профессиональных знаний и учетом фактора уверенности. В этом случае эксперт априори сообщает системе свою уверенность в истинности фактов в левой

части. То есть информация: «Если (p_1 (0,5) И p_2 (0,4) ИЛИ p_3 то d_1 (0,8) И d_2 (0,3)», означает его уверенность (0,5) в том, что лето будет жаркое и уверенность (0,4) в том, что рост посевов будет слабый.

Как отразится наличие такого правила в базе знаний на моделировании рассуждений с использованием теории фактора уверенности?

В этом случае выполняются следующие расчеты:

$$F_y(p_1 \text{ И } p_2) = \min(F_y(p_1), F_y(p_2)) = \min(0,5; 0,4) = 0,4,$$

$$F_y(p_1 \text{ ИЛИ } p_2) = \max(F_y(p_1), F_y(p_2)) = \max(0,5; 0,4) = 0,5,$$

и гипотезы добавляются в базу фактов со следующими значениями фактора уверенности: для d_1 — $(0,8 * 0,4) = 0,32$, т.е. d_1 (0,32), для d_2 — $(0,3 * 0,4) = 0,12$, т.е. d_2 (0,12).

Таким образом, рассмотрен механизм «действия» фактора уверенности в «пределах» одного правила из базы знаний.

А как имитировать рассуждения эксперта, когда два или более правил приводят к одному и тому же результату, т.е. гипотезе?

В этом случае используют приемы из теории вероятностей. Значения фактора уверенности при объединении (комбинировании) независимых свидетельств перемножаются. Многократно используя это обстоятельство, можно объединить результаты применения в рассуждениях любого количества правил из базы знаний, в правой части которых указан один и тот же результат. Если $F_y(d_1)$ — фактор уверенности в действии d , рассчитанный по одному правилу, а $F_y(d_2)$ — фактор уверенности в действии d , рассчитанный по второму правилу, то новое значение $F_y(d)$ вычисляется следующим образом: $F_y(d_1) + F_y(d_2) - (F_y(d_1) * F_y(d_2))$, если $F_y(d_1), F_y(d_2) > 0$; $F_y(d_1) + F_y(d_2) + (F_y(d_1) * F_y(d_2))$, если $F_y(d_1), F_y(d_2) < 0$; $(F_y(d_1) + F_y(d_2)) / (1 - \min(|F_y(d_1)|, |F_y(d_2)|))$, где $|x|$ — абсолютная величина x во всех остальных случаях.

Комбинированная мера $F_y(d)$ — монотонно возрастающая (убывающая) функция, что характерно для обобщенного свидетельства.

Мера уверенности в теории фактора уверенности описывает экспертную (субъективную) оценку причинной вероятностной меры. В отличие от байесовского подхода, где необходимо выделить и скомби-

нирывать все априорные и апостериорные вероятности, теория фактора уверенности позволяет эксперту описать все эти взаимосвязи одним фактором F_y доверия к правилу из базы профессиональных знаний. Считается, что такой подход лучше отражает способ мышления эксперта в ходе рассуждений [137].

Тем не менее, значение меры доверия не так строго обосновано, как в теории вероятностей, и носит эвристический характер.

Рассуждения с использованием лингвистической переменной в условиях лингвистической неопределенности. Как было отмечено в разд. 1.5, эксперты и ЛПР действуют в СППР в условиях мультязыковой среды, образованной комбинированием рассуждений на нескольких языках профессиональной деятельности. Лингвистическая языковая неопределенность, связанная с неточностью обычного человеческого языка, начала активно исследоваться с 1965 г., когда в журнале “Information and Control” вышла статья Л. Заде “Fuzzy sets” («нечеткие множества»). В предисловии к книге «Нечеткие множества в моделях управления и искусственного интеллекта» Д.А. Поспелов пишет, что основная идея Заде состояла в том, что человеческий способ рассуждений на естественном языке, не может быть описан в рамках математических формализмов. Этим формализмам присуща однозначность интерпретации, а все, что связано с использованием естественного языка, имеет многозначную интерпретацию.

Программа построения нечеткой математики быстро нашла широкий отклик у ученых и инженеров во всем мире. В нашей стране известны работы А. Н. Аверкина, И. З. Батыршина, Л.С. Берштейна, А.Н. Борисова, А.Н. Мелихова, А.С. Нариньяни, В. Б. Тарасова и других ученых. Из зарубежных ученых в области нечеткой математики следует отметить работы L. Zade, A. DeLuca, S. Termini, J. Fodor, E. Mamdani, M. Sugeno.

Исследования развивались в направлении расширения таких фундаментальных понятий математики, как функция, отношение, предикат, так и в прикладных областях по созданию нечетких алгоритмов. Нечеткое управление, впервые испытанное в начале 70-х годов Е. Mamdani, сотрудником Лондонского университета, в виде нечеткого алгоритма управления лабораторной паровой машиной [154], уже в начале 80-х годов было реализовано в Дании для управления цементной печью. В Японии исследования по нечеткому управлению начались примерно в 1985 г., и в начале 90-х годов имелись примеры их практического применения [154]. Именно в этот период трудами японских ученых [154] появилось понятие «нечеткая система». В [120] подчеркивается, что приоритет в нечетких системах принадлежит японским ученым.

Рассмотрим несколько существенных для дальнейшего изложения определений [67].

Нечеткое множество A — это пара объектов $\langle U, \mu_A \rangle$, где U — универсум; μ_A — функция принадлежности, т.е. подмножество декартова произведения универсума U и отрезка $[0,1]$, причем $A \subset U \otimes [0,1]$.

Нечеткая переменная — это кортеж $\langle \alpha, U, A \rangle$, где α — наименование или название нечеткой переменной; U — область ее определения, т.е. универсум; A — нечеткое множество на U , описывающее возможные значения, которые может принимать нечеткая переменная α .

Лингвистическая переменная — это кортеж $\langle \beta, T, U, P_{r_1}, P_{r_2} \rangle$, где β — наименование или название лингвистической переменной, T — множество ее значений (терм-множество), каждое из которых представляет собой наименование отдельной нечеткой переменной α ; U — область определения (универсум) нечетких переменных, которые входят в определение лингвистической переменной; P_{r_1} — синтаксическая процедура, которая описывает процесс образования или генерирования из элементов множества T новых термов; P_{r_2} — семантическая процедура, ставящая в соответствие каждому новому терму, образуемому процедурой P_{r_1} (значению лингвистической переменной), осмысленное содержание (нечеткую семантику) посредством формирования соответствующего нечеткого множества.

Для примера введем лингвистическую переменную «урожай». Многие эксперты-агрономы дают субъективную оценку результатов своего труда. Используем словосочетания типа «малый урожай», «средний урожай», «высокий урожай», относящиеся к диапазону от 0 до 80 ц с гектара. Тогда $\langle \beta, T, U, P_{r_1}, P_{r_2} \rangle$, где β_1 — «урожай»; $T = \{\text{«малый урожай»}, \text{«средний урожай»}, \text{«высокая скорость»}\}$; $U = [0,80]$; P_{r_1} — процедура образования новых термов с помощью логических связок И, ИЛИ, НЕ и модификаторов типа «очень», «слегка» и др., например, «малый или средний урожай», «очень высокий урожай»; P_{r_2} — процедура задания на $U = [0,80]$ нечетких переменных α_1 — «малый урожай», α_2 — «средний урожай», α_3 — «высокий урожай», а также соответствующих нечетких множеств для термов из $P_{r_1}(T)$ по правилам P_{r_1} .

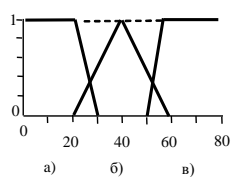


Рис. 3.10 — Графики функций принадлежности множеств A_1, A_2, A_3 , соответствующих нечетким переменным: а) — «малый урожай», б) — «средний урожай», в) — «высокий урожай»

Нечеткие множества A_1, A_2, A_3 , соответствующие нечетким переменным α_1 — «малый урожай», α_2 — «средний урожай», α_3 — «высокий урожай», задаются графически кусочно-линейными функциями принадлежности. Один из возможных вариантов нечетких множеств A_1, A_2, A_3 изображен на рис. 3.10.

Нечеткие лингвистические высказывания — высказывания следующих трех видов: 1) высказывание « β есть α »; 2) высказывание « β есть $\nabla \alpha$ », где ∇ — модификатор; 3) составные высказывания, образованные из выражений видов 1 и 2 и нечетких логических операций в форме связок И, ИЛИ, ЕСЛИ-ТО, НЕ.

Рассмотрим варианты записи правила нечеткой продукции, наиболее часто используемые для рассуждений в нечетких системах (рис. 3.11).

Простейший вариант (вариант 1) можно представить в следующем виде:

$$\text{Правило } \langle \text{№} \rangle: \text{Если } \langle \beta_1 \text{ есть } \alpha' \rangle \text{ то } \langle \beta_2 \text{ есть } \alpha'' \rangle, \quad (3.18)$$

где нечеткое высказывание « β_1 есть α' » — нечеткое условие, а нечеткое высказывание « β_2 есть α'' » — заключение данного правила ($\beta_1 \neq \beta_2$).

Вариант 2 — когда нечеткими логическими операциями соединены нечеткие высказывания, относящиеся к разным лингвистическим переменным, записанным в условной (левой) части правила. Этот вариант может быть записан в одной из двух форм:

$$\text{Правило } \langle \text{№} \rangle: \text{Если } \langle \beta_1 \text{ есть } \alpha' \rangle \text{ ИЛИ } \langle \beta_2 \text{ есть } \alpha'' \rangle \text{ то } \langle \beta_3 \text{ есть } \omega \rangle, \quad (3.19)$$

$$\text{Правило } \langle \text{№} \rangle: \text{Если } \langle \beta_1 \text{ есть } \alpha' \rangle \text{ И } \langle \beta_2 \text{ есть } \alpha'' \rangle \text{ то } \langle \beta_3 \text{ есть } \omega \rangle,$$

где $\beta_i \neq \beta_j \mid i, j = \{1, 2, 3\}, i \neq j$. В этом случае в рассуждениях необходимо использовать один из методов агрегирования условий.

Вариант 3, когда нечеткими логическими операциями соединены нечеткие высказывания, относящиеся к разным лингвистическим пере-

менным, записанным в заключении (правой части) правила. Этот вариант может быть записан в одной из двух форм:

Правило <№>: Если « β_1 есть α » то « β_2 есть ω' » И « β_3 есть ω'' », (3.20)

Правило <№>: Если « β_1 есть α » то « β_2 есть ω' » ИЛИ « β_3 есть ω'' »,

где $\beta_i \neq \beta_j \mid i, j = \{1, 2, 3\}, i \neq j$. В этом случае в рассуждениях необходимо использовать один из методов аккумуляции заключений.

На рис. 3.11 изображена архитектура классической нечеткой системы. Рассмотрим этапы ее функционирования: фазификацию, агрегирование, активацию, аккумуляцию и дефазификацию.

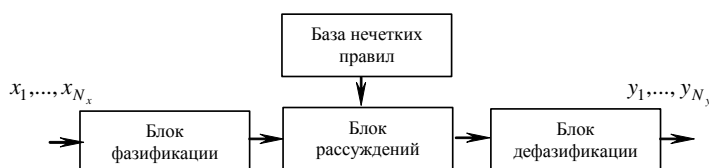


Рис. 3.11 — Архитектура классической нечеткой системы

При этом вместо единого обозначения наименования лингвистической переменной — β используется обозначение x — для входных и y — для выходных лингвистических переменных нечеткой системы.

Фазификация. Фазификация выполняется в блоке фазификации с целью установления соответствия между конкретными значениями $\dot{x}_1, \dots, \dot{x}_{N_x}$, где $\dot{x}_i \mid i = 1, \dots, N_x \in U_i$ — универсум лингвистической переменной x_i , входных лингвистических переменных x_1, \dots, x_{N_x} и значением функции принадлежности соответствующего термина лингвистической переменной. Предполагается, что $\dot{x}_1, \dots, \dot{x}_{N_x}$ известны от датчиков или эксперта. Далее рассматриваются нечеткие высказывания « x_i есть α' », где α' — терм (нечеткая переменная, как значение лингвистической переменной x_i) с известной функцией принадлежности $\mu(x_i)$ на универсуме U_i , например, если $i = 1$, то «урожай (x_1) есть малый (α_1)», «урожай (x_1) есть средний (α_2)», «урожай (x_1) есть высокий (α_3)» и находятся количественные значения $b_i' = \mu(\dot{x}_i)$.

Агрегирование. Выполняется в блоке рассуждений. До начала этого этапа известно множество значений истинности $B' = \{b'_1, b'_2, \dots, b'_{N_B}\}$ всех нечетких высказываний « x_i есть α' ». Далее рассматривается каждое из условий в левой части продукции и находится его степень истинности. Если условие в правиле — нечеткое высказывание (3.18), (3.19), т.е. вариантов 1 или 2, то степень его истинности равна соответствующему значению $b'_i \mid i = 1, 2, \dots, N_B$. Если условие в правиле — нечеткое высказывание (3.20), т.е. варианта 3 и лингвистические переменные в нечетких высказываниях не равны друг другу, то степень его истинности вычисляется из соответствующих значений b'_i , которые являются аргументами соответствующих логических операций: $T(b'_i \wedge b'_j) = \min(T(b'_i), T(b'_j))$, $T(b'_i \vee b'_j) = \max(T(b'_i), T(b'_j))$, $T(\neg b'_i) = 1 - T(b'_i)$. Этап считается законченным, когда для всех правил N_{KB} из рассматриваемой базы знаний найдено значение $b''_k \mid k = 1, 2, \dots, N_B$. Обозначим множество этих значений через $B'' = \{b''_1, \dots, b''_2, b''_{N_B}\}$.

Активизация. Выполняется в блоке рассуждений. До начала этого этапа известно множество значений истинности $B'' = \{b''_1, \dots, b''_2, b''_{N_B}\}$ всех условий всех правил из базы знаний, а также значения весовых коэффициентов уверенности (определенности) w_k из базы знаний, выражающих количественную оценку степени истинности или относительный вес нечеткой продукции. Далее рассматривается каждое из заключений правил и находится степень его истинности w_k . Если заключение правила — высказывание (3.18) или (3.19), то степень его истинности рассчитывается по формуле $c_k = b''_k \cdot w_k$. Если заключение правила — высказывание (3.20), причем лингвистические переменные в его высказываниях попарно не равны друг другу, то степень истинности каждого из высказываний равна $b''_k \cdot w_k$. Роль конъюнкции и дизъюнкции нечетких высказываний в правой части (заключении) не является определяющей в вычислениях (точнее — не влияет на вычисления) по сравнению с условной частью. Это объясняется тем, что термы в правой части интерпретируются как действия оператора или исполнительных механизмов, которые могут быть выполнены совместно (конъюнкция) или одно из них (дизъюнкция) по выбору эксперта. Далее

находятся все значения c_k степеней истинности для нечетких высказываний для каждого из правил построенного по (3.20). Это множество значений обозначим как $C = \{c_1, c_2, \dots, c_{N_C}\}$, где N_C — общее количество нечетких высказываний в правилах базы знаний, построенных по (3.20), N_C может быть равно N_{KB} . Весовой коэффициент w_k может быть задан индивидуально для отдельных нечетких высказываний, а процедура активизации остается прежней. После нахождения множества $C = \{c_1, c_2, \dots, c_{N_C}\}$ определяются функции принадлежности каждого из нечетких высказываний для выходных лингвистических переменных одним из методов, являющихся модификацией того или иного метода нечеткой композиции, например:

$$\begin{aligned} \text{min-активизация: } \mu'(y) &= \min(c_k, \mu(y)), \\ \text{prod-активизация: } \mu'(y) &= c_k \cdot \mu(y), \end{aligned} \quad (3.21)$$

где $\mu(y)$ — функция принадлежности термина (нечеткого высказывания), являющегося значением выходной переменной y_j , значения которой $\dot{y}_1, \dots, \dot{y}_{N_Y}$ задаются на универсуме $V_j \mid j = 1, 2, \dots, N_Y$. Этап активизации закончен, когда для каждой из выходных лингвистических переменных y_1, \dots, y_{N_Y} , входящих в отдельные нечеткие высказывания заключенных нечетких правил, определены функции принадлежности нечетких множеств их значений. Иными словами, определена совокупность нечетких множеств C_1, C_2, \dots, C_{N_C} . Кроме рассмотренных, существуют и другие методы активизации.

Аккумуляция. Выполняется в блоке рассуждений. До начала этапа известны значения истинности всех нечетких высказываний для каждого правила из базы знаний, в форме совокупности нечетких множеств C_1, C_2, \dots, C_{N_C} . Далее последовательно рассматривается каждая из выходных лингвистических переменных $y_j \in \mathbf{Y} = \{y_1, y_2, \dots, y_{N_Y}\} \mid j = 1, 2, \dots, N_Y$ и относящиеся к ней нечеткие множества $C_i \mid i = 1, 2, \dots, N_C$ из множества C_1, C_2, \dots, C_{N_C} . Такие выбранные множества обозначаются далее, как $C_l \mid l = 1, 2, \dots, N_Y$. Результат аккумуляции определяются одной из операций с нечеткими множествами: объединением, пересечением, суммой, а также другими. Этап аккумуляции закончен, когда для каждой лингвистической переменной определены итоговые функции принадлежности

нечетких их значений, т.е. совокупность нечетких множеств $C_1', C_2', \dots, C_{N_Y}'$.

Дефазификация. Дефазификация (или приведение к четкости) выполняется в блоке дефазификации для реализации результатов нечетких рассуждений исполнительными механизмами. Если результаты рассуждений будет реализовывать человек, то дефазификация может и не выполняться, если только этот человек не приводит в действие устройства, для которых параметры должны быть заданы в количественном виде точно. Формально дефазификация выполняется так. До начала этого этапа определены функции принадлежности всех выходных лингвистических переменных в форме нечетких множеств $C_1', C_2', \dots, C_{N_Y}'$. Затем последовательно рассматриваются выходные лингвистические переменные $y_j \in Y = \{y_1, y_2, \dots, y_{N_Y}\} | j = 1, 2, \dots, N_Y$ и относящиеся к ним нечеткие множества C_j' . Результат дефазификации переменной y_j — вещественное число \dot{y}_j , которое получается с использованием одного из методов дефазификации [67]. В [155] приведен результат сравнительного анализа 14 методов дефазификации.

Методы моделирования нечетких рассуждений эксперта. Пусть в итоге извлечения из эксперта правил разработана база знаний, состоящая из двух следующих правил:

Если (x_1 есть A_1 и x_2 есть B_1) то (y есть C_{01}),

Если (x_1 есть A_2 и x_2 есть B_2) то (y есть C_{02}),

где x_1, x_2 — имена входных лингвистических переменных; y — имя выходной лингвистической переменной; $A_1, A_2, B_1, B_2, C_{01}, C_{02}$ — некоторые заданные нечеткие множества. В ходе рассуждений необходимо на основе базы нечетких правил и четких значений \dot{x}_1, \dot{x}_2 переменных x_1 и x_2 соответственно, получить четкое значение \dot{y} переменной y .

Основная отличительная особенность нечетких рассуждений — все правила «срабатывают» параллельно. Отсутствуют сопоставление с образом, разрешение конфликтов, т.е. нет поиска продолжения рассуждений как такового, что значительно увеличивает производительность вычислений.

Размеры базы правил нечетких систем обычно на порядок меньше, чем если бы аналогичная задача решалась экспертной системой с детерминированными переменными и двоичной логикой рассуждений.

Рассмотренные этапы нечетких рассуждений идут в следующем порядке: фазификация, агрегирование, активизация, аккумуляция, дефазификация.

Фазификация выполняется обычно вычислением пересечения функции принадлежности входной лингвистической переменной и синглетона [1, 67], полученного из входного четкого значения переменной. Агрегирование выполняется разными способами, например, вычислением минимума или произведения, максимума или суммы, а активизация — одним из методов вычисления импликации [67]. Аккумуляция лингвистических переменных производится различными методами, например, логическим объединением [67]. Алгоритмы нечетких рассуждений обычно различаются по методам активизации и аккумуляции.

Моделирование нечетких рассуждений эксперта методом (алгоритмом) Мамдани (Mamdani) или *min-max* методом. Пример нечетких рассуждений на нашей базе знаний в соответствии с методом Мамдани показан на рис. 3.12, а порядок действий нечеткой системы, моделирующей эти рассуждения, приведен ниже:

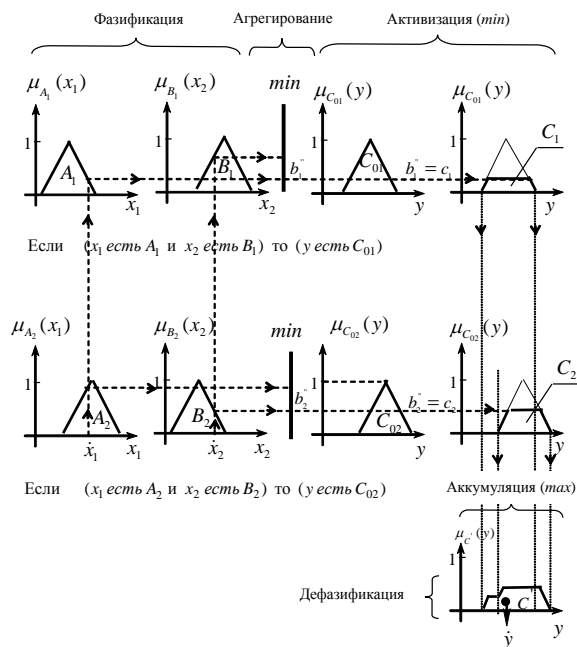


Рис. 3.12 — Нечеткие рассуждения методом Мамдани (*min-max*)

1. Фазификация четких исходных значений \dot{x}_1, \dot{x}_2 производится стандартным способом.
2. Агрегирование осуществляется *min*-конъюнкцией. Имеем c_j .
3. Активизация производится *min*-активизацией, которая обычно называется отсечением функции принадлежности нечеткого множества.
4. Аккумуляция выполняется по формуле логического объединения нечетких множеств. В итоге имеем нечеткое множество C' , которое затем дефазифицируется одним из методов в четкое значение \dot{y} .

Моделирование нечетких рассуждений эксперта методом (алгоритмом) Ларсена (Larsen), или *prod-max* методом. Пример нечетких рассуждений на нашей базе знаний в соответствии с методом Ларсена показан на рис. 3.13, а порядок действий нечеткой системы, моделирующей эти рассуждения, приведен ниже:

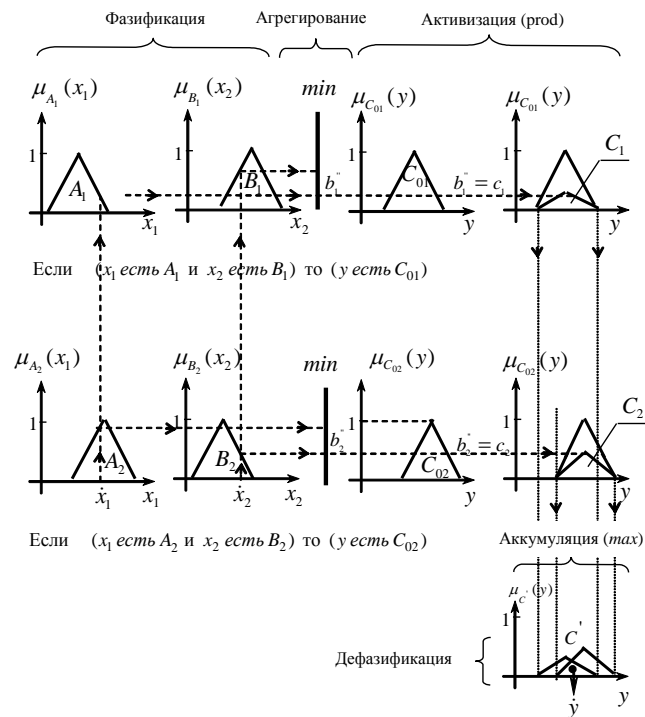


Рис. 3.13 — Нечеткие рассуждения методом Ларсена (*prod-max*)

1. Фазификация четких исходных значений \dot{x}_1, \dot{x}_2 производится стандартным способом.
2. Агрегирование осуществляется *min*-конъюнкцией. Имеем c_j .
3. Активизация производится *prod*-активизацией, которая обычно называется масштабированием функции принадлежности нечеткого множества.
4. Аккумуляция выполняется по формуле логического объединения нечетких множеств. В итоге имеем нечеткое множество C' , дефазифицируемое одним из методов в четкое значение \dot{y} .

Моделирование нечетких рассуждений эксперта методом (алгоритмом) Цукамото (Tsukamoto). Пример нечетких рассуждений на нашей базе знаний в соответствии с методом Цукамото показан на рис. 3.14.

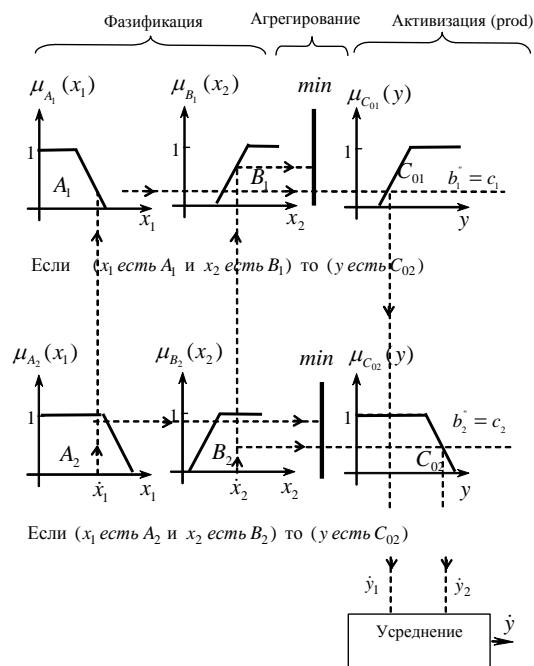


Рис. 3.14 — Нечеткие рассуждения методом Цукамото

Для сокращения объема вычислений в методах нечетких рассуждений используются различные упрощения. В частности, в алгоритме Цу-

камото лингвистические переменные имеют монотонные, неубывающие и невозрастающие функции принадлежности. Порядок действий нечеткой системы, моделирующей эти рассуждения, приведен ниже:

1. Фазификация четких исходных значений \dot{x}_1, \dot{x}_2 производится стандартным способом.
2. Агрегирование осуществляется *min*-конъюнкцией. Имеем c_j .
3. Нахождение четких значений всех выходных лингвистических переменных y_j из уравнений: $c_j = \mu(y_j) | j = 1, 2, \dots, N_Y$.
4. Аккумуляция отсутствует. В итоге имеем четкое значение \dot{y} , полученное из чисел c_j и \dot{y}_j каким-либо методом усреднения, например, взвешенным средним арифметическим чисел \dot{y}_j :

$$y = \left(\sum_{j=1}^{N_{KB}} c_j \cdot y_j \right) / \sum_{j=1}^{N_{KB}} c_j.$$

Моделирование нечетких рассуждений эксперта методом (алгоритмом) Сугено (Sugeno). Пример нечетких рассуждений на нашей базе знаний в соответствии с методом Сугено показан на рис. 3.15. В алгоритме Сугено применяются правила, в которых выходные переменные функционально зависят от входных: $y_j = f_j(x_1, x_2, \dots, x_{N_x}) | j = 1, 2, \dots, N_Y$, где f_j — некоторая функция. Порядок действий нечеткой системы, моделирующей эти рассуждения, приведен ниже:

1. Фазификация четких исходных значений \dot{x}_1, \dot{x}_2 производится стандартным способом.
2. Агрегирование осуществляется *min*-конъюнкцией. Имеем c_j .
3. Вместо активизации находятся четкие значения всех выходных лингвистических переменных y_j из уравнения $y_j = f_j(x_1, x_2)$.
4. Аккумуляция отсутствует. В итоге имеем четкое значение \dot{y} , полученное из чисел c_j и y_j каким-либо методом усреднения.

Упрощенный метод нечетких рассуждений. Пример нечетких рассуждений на нашей базе знаний в соответствии с упрощенным методом показан на рис. 3.16. В данном методе полагается, что $y_j = \varepsilon_j = \text{const} | j = 1, 2, \dots, N_Y$. Порядок действий нечеткой системы, моделирующей эти рассуждения, приведен ниже:

1. Фазификация четких исходных значений \dot{x}_1, \dot{x}_2 производится стандартным способом.
2. Агрегирование осуществляется \min -конъюнкцией. Имеем c_j .
3. Вместо активизации берутся четкие значения всех выходных лингвистических переменных $y_j: y_j = \varepsilon_j$.
4. Аккумуляция отсутствует. В итоге имеем четкое значение \dot{y} , полученное из чисел c_j и \dot{y} (т.е. ε_j) каким-либо методом усреднения.

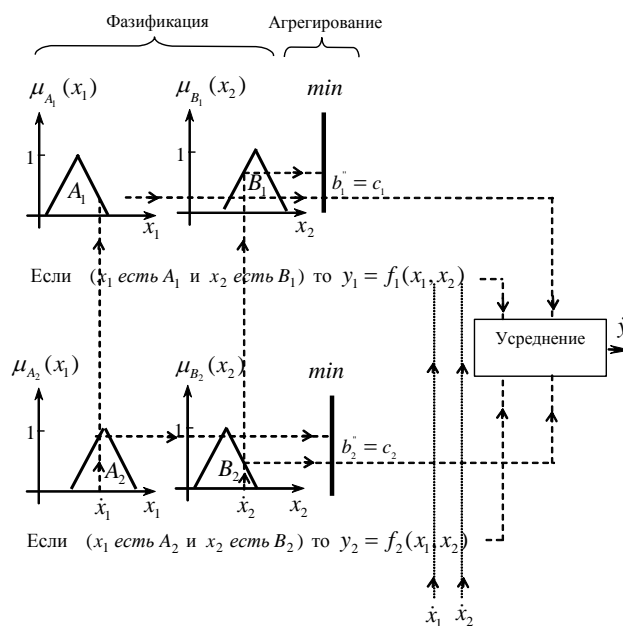


Рис. 3.15 — Нечеткие рассуждения методом Сугено

Обобщая сказанное о методах рассуждения в условиях лингвистической неопределенности, отметим, что нечеткие системы [1, 67, 154, 156 — 159] используют лингвистические (знаковые) представления, которые в языках профессиональной деятельности имеют неясный, размытый характер. Английское слово “fuzz”, от которого образовано прилагательное “fuzzy” («нечеткий»), в словосочетании “fuzzy systems” означает «ворс» — специальный термин, определяющий свойство тканей. Когда человек смотрит на рисунок из ворсистой ткани, он кажется ему размытым. Поэтому, говоря «нечеткий», имеют в виду «неясный»,

150

«размытый». Специфические для нечетких систем представления условно изображены на рис. 3.17.

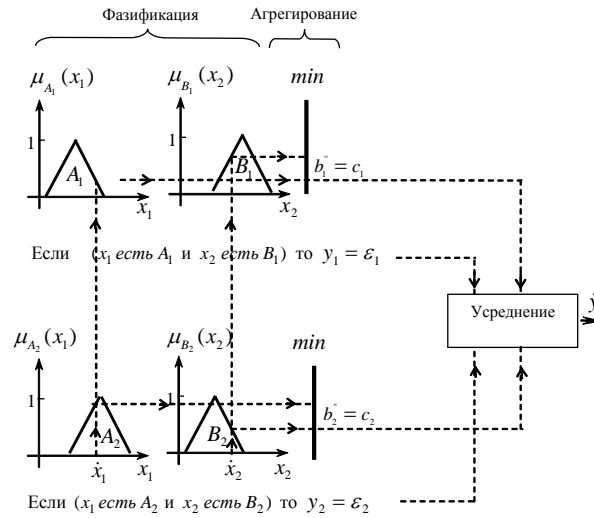


Рис. 3.16 — Нечеткие рассуждения упрощенным методом

В этих представлениях система S отображается в объект, который называют «серый ящик» [154]. Использование этого термина означает, что модель объекта управления превращается из кибернетического «черного ящика» в «серый ящик», т.е. более понятный и прозрачный для специалиста.

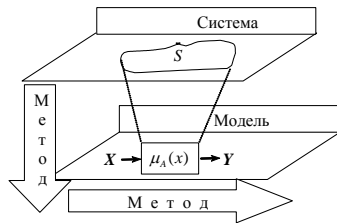


Рис. 3.17 — Методы нечетких систем

Входы (\mathbf{X}) и выходы (\mathbf{Y}) серого ящика могут представляться по-разному, однако главная отличительная особенность нечетких систем — применение при этом нечетких множеств A и многозначных функций принадлежности $\mu_A(x)$ элементов x этим множествам, заменяющих пропозициональные функции для обычных множеств в методах рассуждений для экспертных систем. С использованием таких представлений, а также определенных в теории нечетких множеств операций и строится механизм преобразования входа в выход, дающий решение поставленной задачи.

На базе нечетких представлений развиваются методы дискретной математики, математической лингвистики и теория интеллектуального управления [115, 120]. Наметилась тенденция применения нечетких представлений в гуманитарных науках и социологии.

Нечеткие представления помогают решать задачи управления роботами, техническими процессами, распознавания образов и автоматического перевода, хранения и поиска данных. В сфере управления — задачи принятия решения, консультаций, планирования, прогноза для сельского хозяйства, оценки качества и др. Достоинства и недостатки нечетких представлений и нечетких систем даны в табл. 1.12 и [1].

Концептуальная модель для методов нечетких систем дана ниже:

$$\begin{aligned} \hat{m}_f^a |^{Te} &= {}^1x_2^0 \circ {}^{11}r^n \circ {}^1x_1^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{13}^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{25}^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{33}^0 \circ \\ &\circ {}^1x_2^0 \circ {}^{13}r^n \circ {}^3x_{14}^0 \circ {}^1x_2^0 \circ {}^{18}r^n \circ {}^8\pi_f^h \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^2O_f^m, \\ {}^8\pi_f^h &= A \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2K_f^h \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2O_f^h \circ B, \end{aligned} \quad (3.22)$$

где ${}^8\pi_f^h$ — схема ролевых концептуальных моделей задач, решаемых нечеткими системами; ${}^2K_f^h$ — включает фазу — «контроль», «анализ», «прогнозирование», переменные — «детерминированные», «нечеткие лингвистические»; класс задач — (см. табл. 7.2 в [1]); переменные-отношения — «сравнения», «причинности»; ${}^2O_f^h$ — специфицируется в ходе разработки; A и B — фрагменты (3.22), уточняемые в ходе разработки; ${}^2O_f^m$ — спецификатор, включающий: знания о возможной погрешности решения в целом для класса методов — отсутствуют; знания о гибридных возможностях даны в табл. 8.6 [1].

Модель вычислений в нечетких системах приведена ниже:

$$\hat{m}_f^a |^{CA} = \langle \mathbf{X}, \mathbf{Y}, KB, F^\mu, F^{TS}, F^y, I^f \rangle, \quad (3.23)$$

где \mathbf{X}, \mathbf{Y} — пространства входных и выходных лингвистических переменных соответственно; $\mathbf{X} = \mathbf{X}_1 \times \dots \times \mathbf{X}_{N_x}$, $\mathbf{Y} = \mathbf{Y}_1 \times \dots \times \mathbf{Y}_{N_y}$, $\mathbf{x} \in \mathbf{X}$, $\mathbf{x} = [x_1, \dots, x_{N_x}]^{Tr}$, $\mathbf{y} \in \mathbf{Y}$, $\mathbf{y} = [y_1, \dots, y_{N_y}]^{Tr}$; KB — база знаний, как множество из $k = 1, \dots, N_{KB}$ правил иногда называемая лингвистической моделью; $F^\mu = \{F_x^\mu, F_y^\mu\}$ — множество функций принадлежности входов и выходов соответственно; F^{TS} — множество функциональных зависи-

мостей в качестве заключений в алгоритме Сугено; F^y — множество аналитических выражений и (или) алгоритмов дефазификации; Tr — операция транспонирования.

Интерпретатор I^f может быть представлен совокупностью пяти процессов: $I^f = \langle I^{f1}, I^{f2}, I^{f3}, I^{f4}, I^{f5} \rangle$, где I^{f1} — фазификация; I^{f2} — агрегирование; I^{f3} — активизация; I^{f4} — аккумуляция; I^{f5} — дефазификация.

Процессы I^{f1}, \dots, I^{f5} выполняются последовательно в следующем порядке:

$$I^{f1} \rightarrow I^{f2} \rightarrow I^{f3} \rightarrow I^{f4} \rightarrow I^{f5}. \quad (3.24)$$

3.3.3. Методы вычислений в искусственных нейронных сетях

Специфика предметной области, в которой работает эксперт, может способствовать развитию у него специфических интуитивно-образно-интуитивных представлений [160].

Логические рассуждения, использование которых для решения задач экспертами было рассмотрено в разд. 3.3.1, и интуитивно-образные представления могут сменять друг друга в процессе анализа проблем [160]. Это объясняется двойственностью восприятия и познания, представленных образным и аналитическим свойствами высших психических процессов. Можно сказать, что разум (логическое рассуждение) и чувственное восприятие действительности («отраженный свет») — два лика сфинкса, именуемого постижением мира.

Знания, в обязательном плане включающие отношения между объектами, могут быть представлены «мысленными образами» явлений внешнего мира, сформировавшимися в прошлом на основе фактов, но не являющимися их простым отражением. Появление новых образов, как и новых данных, может приводить к изменению направления мыслительного процесса эксперта. Например, интуиция в медицине способствует выбору рационального пути обследования больного, что оптимизирует диагностический процесс. Намек (деталь) как ориентир вызывает у врача или цепочку признаков, складывающихся в образ (симультанное объединение признаков), или «всплывание» образа («инсайт», момент мгновенного проникновения в суть вещей) [160].

Достижения в области нейрофизиологии дают начальное понимание механизма естественного мышления, где хранение информации происходит в виде сложных образов. Процесс хранения информации как образов, использование образов и решение поставленной задачи определяют новую область в обработке данных, которая, не используя традиционного программирования, обеспечивает создание параллельных сетей и их обучение — нейровычисления. В связи с этим И.Б. Фоминих отмечает, что, в принципе ... можно использовать для принятия решений и «образное мышление», свойственное нейросетям [161].

Продолжительный период эволюции придал мозгу человека много качеств, необходимых для решения сложных задач, отсутствующих в современных компьютерах с архитектурой фон Неймана. К ним относятся: распределенное представление информации и параллельные вычисления; способность к обучению и обобщению; адаптивность; толерантность к ошибкам; низкое энергопотребление.

Еще одно важное свойство процессов обработки информации в мозгу человека следует из различий естественно-языковой (высокоуровневой) и «стимул-реакция» (низкоуровневой) подсистем нервной системы человека. Моделирование высокоуровневых процессов в ИИ традиционно связывают с методами рассуждений в экспертных системах, а обработку «стимул-реакция», где огромную роль играют интуиция, опыт и, наверное, самое главное обстоятельство быстрота реакции — с нейровычислениями.

В этой связи в СППР можно услышать фразу: «Мы приняли неправильное решение, зато уверенно и быстро». Быстроте и уверенности в действиях экспертов и ЛПР отдается приоритет в некоторых специфических условиях, когда задержка с выработкой и принятием решения, по сути, означает отказ от принятия решения, что может привести к значительно более серьезным проблемным ситуациям, чем текущая. Быстро реагировать на происходящее — это не значит всегда ошибаться. Это значит быстро обрабатывать меньшие объемы информации, чем нужны для полного и всестороннего рассмотрения задачи. У быстро выработанного экспертного решения есть еще одно преимущество в СППР — оно может быть также быстро уточнено и оценено другими экспертами и ЛПР. Рекордное быстродействие развивают компьютеры, имитируя рассуждения в условиях нечеткости (см. разд. 3.3.2) и выполняя нейровычисления. Настоящий раздел открывает обсуждение методов моделирования интуитивно-образного мышления экспертов в СППР. Продолжен этот разговор будет в разд. 3.5.

Нейровычисления. Согласно взглядам современных коннекционистов, нейронные сети — это упрощенные модели мозга, состоящие из большого числа модулей (аналогов нейронов), которым приписыва-

ются веса, измеряющие силу соединений между модулями. Эти веса моделируют действия синапсов, обеспечивающих информационный обмен между нейронами.

Модули нейронной сети обычно делят на три класса: входные модули, которые получают необходимую для обработки информацию; выходные модули, где содержатся результаты обработки информации; модули, находящиеся между входными и выходными, получившие название скрытых модулей. Если нейронную сеть рассматривать как модель человеческого мозга, то входные модули аналогичны сенсорным нейронам, выходные — моторным нейронам, а скрытые модули — всем другим нейронам. Каждый входной модуль имеет величину возбуждения, репрезентирующую некоторое свойство, внешнее к сети. Сигнал от входных модулей распространяется по связям между нейронами через сеть и определяет величины возбуждения во всех скрытых и выходных модулях. Величина возбуждения для каждого получающего сигнал нейрона рассчитывается согласно функции возбуждения (активации). Поскольку допускается, что все модули вычисляют ту же самую простую функцию возбуждения, то успешное моделирование человеческих интеллектуальных действий зависит, прежде всего, от параметров настройки весов между нейронами. Поэтому нахождение правильного набора значений, необходимых для выполнения данной задачи, — главная цель в исследованиях коннекционистов. Для этого изобретены соответствующие алгоритмы, вычисляющие правильные значения весов, необходимые для решения многих задач.

Успешное применение коннекционистских методов зависит от весьма тонкой корректировки таких алгоритмов и используемых для обучения наборов чисел.

Некоторые из искусственных нейронных сетей обладают способностью извлекать сущность из входных сигналов. Например, сеть может быть обучена на последовательности искаженных версий буквы «А». После соответствующего обучения предъявление такого искаженного примера приведет к тому, что сеть породит букву совершенной формы. В некотором смысле она научится порождать то, что никогда не «видела». Способность извлекать идеальные прототипы — весьма ценное качество экспертов.

Отклик сети после обучения может быть до некоторой степени нечувствителен к небольшим изменениям входных сигналов. Эта внутренне присущая способность видеть образ сквозь шум и искажения жизненно важна в решении задач и распознавании образов. Она позволяет преодолеть требование строгой точности, предъявляемое обычным компьютером, и открывает путь к системе, которая может иметь дело с несовершенным миром экспертных оценок в СППР. Важно отметить,

что искусственная нейронная сеть делает обобщения автоматически, благодаря своей структуре, а не с помощью использования «человеческого интеллекта» в форме специально написанных компьютерных программ.

Виды нейровычислений. Для понимания способностей искусственных нейронных сетей имитировать рассуждения специалистов, решающих задачи, нейровычисления разделяют на два вида: нейровычисления с целью обучения и нейровычисления в обученной решать задачи нейросети. Оба вида вычислений зависят от архитектуры (топологии) нейросети, принятой разработчиком. На сегодняшний день, в зависимости от того, какие классификационные признаки вводятся, известно от трех до полутора десятков классов искусственных нейронных сетей.

Нейровычисления с целью обучения. Способность к обучению — основное свойство мозга. Для искусственных нейронных сетей под обучением понимается процесс настройки архитектуры сети (структуры связей между нейронами) и весов синаптических связей — коэффициентов, влияющих на значения сигналов в нейровычислениях, решающих поставленную задачу. Обычно обучение нейронной сети осуществляется на выборке чисел-примеров. По мере процесса обучения, который происходит по специальному алгоритму, сеть должна все лучше и лучше (правильнее) реагировать на входные сигналы.

Выделяют три парадигмы обучения: с учителем, самообучение и смешанную. В первом способе известны правильные ответы к каждому входному примеру, а веса подстраиваются так, чтобы минимизировать ошибку. Обучение без учителя позволяет распределить образцы по категориям за счет раскрытия внутренней структуры и природы данных. При смешанном обучении комбинируются два вышеизложенных подхода. Существует большое число алгоритмов обучения, ориентированных на решение разных задач.

Нейровычисления в обученной решать задачи нейросети обратного распространения погрешности. Нейросеть обратного распространения погрешности (англ. back propagation neural network) с архитектурой feedforward backpropagation была разработана в начале 1970-х годов несколькими независимыми авторами: П.Дж. Вербос (P. Werbos); Д. Паркер (D. Parker); Д. Румельгарт (D. Rumelhart), Г. Хинтон (G. Hinton) и Р. Вильямс (R. Williams).

Типичная сеть обратного распространения ошибки (рис. 3.18) имеет один слой скрытых нейронов и один выходной слой. Теоретически ограничений относительно числа скрытых слоев не существует, но практически применяют один или два. Нейроны организованы в полойную структуру с прямой передачей сигнала. Каждый нейрон сети

продуцирует взвешенную сумму своих входов, пропускает эту величину через передаточную функцию и выдает выходное значение. Определенные числа промежуточных слоев и числа нейронов в них — важное обстоятельство при моделировании сети. Большинство исследователей и инженеров, выбирая архитектуру нейросети для моделирования решения задач управления и обработки информации, использует общие правила, в частности: 1) количество входов и выходов сети определяются количеством входных и выходных параметров исследуемого объекта, явления, процесса, и т.п., т.е. системы S (2.1); 2) число нейронов скрытого слоя выбирается эмпирическим путем. Если сложность в отношении между полученными и желаемыми данными на выходе увеличивается, количество нейронов скрытого слоя должно также увеличиться; 3) если моделируемый процесс может разделяться на много этапов, нужен дополнительный скрытый слой (слои).

В процессе обучения нейросети одним из известных алгоритмов настраиваются (рассчитываются) матрицы синаптических весов ${}^I\mathbf{W}^{1,1}$, ${}^L\mathbf{W}^{2,1}$ и матрицы смещений \mathbf{b}^1 , \mathbf{b}^2 , что устраняет неопределенность в алгоритме нейровычислений (здесь и ниже сохранены обозначения, используемые в математическом пакете MATLAB [162]). После обучения могут быть выполнены нейровычисления с целью решения задачи, стоящей перед экспертом.

Эти вычисления выполняются по формулам, приведенным на рис. 3.18. Характер этих вычислений — арифметические, матричные операции и прямые вычисления по формулам.

С учетом того, что современные компьютеры имеют значительные объемы оперативной памяти и специальные процессоры, нейровычисления выполняются с высокой производительностью.

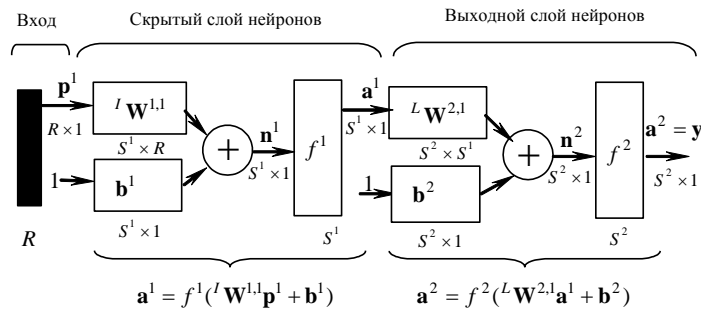
Сеть может моделировать (аппроксимировать) функцию практически любой сложности, причем число слоев и число нейронов в каждом слое определяют сложность функции. С ее помощью моделируется решение экспертами задач распознавания образов, классификации и прогнозирования.

Нейровычисления в обученной решать задачи нейросети Кохонена. Сеть разработана Тойво Кохоненом (Т. Kohonen) в начале 1980-х годов и принципиально отличается от рассмотренной выше сети обратного распространения погрешности, поскольку использует самообучение, и обучающее множество состоит лишь из входных векторов.

Сеть распознает кластеры в больших объемах данных и распределяет данные по соответствующим кластерам (образам). Если дальше сеть встречается с набором данных, не похожим ни на один из известных

образцов, она относит его к нового кластеру. Если в данных содержатся метки классов, то сеть способна решать задачи классификации.

Сеть Кохонена — имеет один слой специальным образом организованных нейронов. Эта организация называется топологией. Она может быть прямоугольной (как на рис. 3.19), где у нейрона четыре нейрона-соседа, гексоидальной, где у нейрона шесть соседей, или другой. До начала обучения эта топология имеет «регулярный» вид, т.е. расстояния между нейроном и его ближайшими соседями, смежными вершинами графа, одинаковы для всех участков карты.



Обозначения: \mathbf{p}^1 — матрица размерности $R \times 1$ входного вектора из R компонент (целые или действительные числа), правый верхний индекс обозначает номер слоя; $l\mathbf{W}^{1,1}$, $L\mathbf{W}^{2,1}$ — матрицы размерности $S^1 \times R$ и $S^2 \times S^1$ синаптических весов первого (скрытого) слоя из S^1 нейронов и второго (выходного) слоя из S^2 нейронов соответственно; \mathbf{b}^1 , \mathbf{b}^2 — матрицы размерности $S^1 \times 1$ и $S^2 \times 1$ смещений для нейронов первого и второго слоя соответственно; 1 — постоянный входной сигнал подаваемый на фиктивный вход; + — сумматор; \mathbf{n}^1 , \mathbf{n}^2 — матрицы размерности $S^1 \times 1$ и $S^2 \times 1$ линейных выходов для S^1 нейронов первого слоя и S^2 нейронов второго слоя соответственно; f^1 , f^2 — функции активации для S^1 нейронов первого слоя и S^2 нейронов второго слоя соответственно; \mathbf{a}^1 , $\mathbf{a}^2 = \mathbf{y}$ — матрицы размерности $S^1 \times 1$ и $S^2 \times 1$ нелинейных выходов (действительные из интервала, заданного функцией активации нейронов) S^1 нейронов первого слоя и $S^2 \times 1$ выходов S^2 нейронов второго слоя (выхода нейросети) соответственно

Рис. 3.18 — Нейровычисления в обученной сети обратного распространения ошибки

В ходе самоорганизации вектора из обучающей последовательности подаются на вход сети и вектора весов нейронов смещаются в области с более высокой плотностью входных векторов. Такие нейроны-победители и их ближайшее соседство, т.е. нейроны, которые в некото-

158

ром смысле близки к победителю, и представляют кластеры во входном пространстве после обучения. Визуально для двумерной карты в результате обучения регулярность топологии нарушается, нейроны группируются в некоторых областях, расстояние между нейроном и его

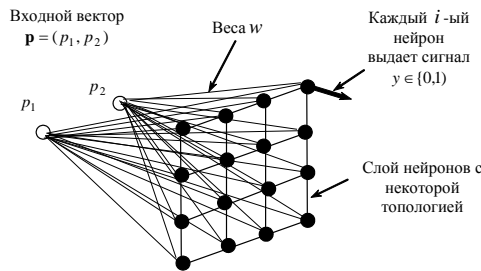


Рис. 3.19 — Самоорганизующаяся карта Кохонена с двумя входами

ближайшими соседями различно для различных участков карты.

Нейровычисления в обученной карте Кохонена выполняются в соответствии с рис. 3.20.

Цель этих вычислений — определить, к какому кластеру относится вектор, посту-

пивший на вход карты. При этом конкретный входной вектор возбуждает нейрон-победитель. Значение на его выходе становится равным единице, в то время как другие нейроны «молчат», что идентифицируется на их выходах нулевыми значениями.

Сеть может использоваться для моделирования решения экспертом задач кластерного анализа, распознавания образов и классификации.

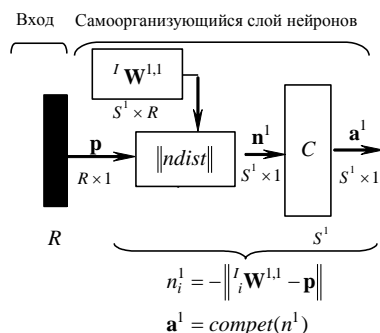
Нейровычисления в обученной вероятностной нейронной сети.

Вероятностная нейронная сеть (англ. probabilistic neural network — PNN) была разработана Дональдом Спехтом (D. Specht) и относится к классу радиальных сетей. Эта сетевая архитектура была впервые представлена в двух статьях: «Вероятностные нейронные сети для классификации, отображения и ассоциативной памяти» (англ. Probabilistic Neural Networks for Classification, Mapping or Associative Memory) [163] и «Вероятностные нейронные сети» [164].

Выходы сети можно интерпретировать, как оценки вероятности принадлежности элементу к определенному классу.

PNN реализуют один из подходов математической статистики к задаче оценки плотности вероятности по имеющимся данным. Этот подход основан на ядерных оценках [165]. Предположим, что наблюдения за значениями случайной величины размещаются в соответствующих им точках пространства. Тогда в этих точках имеется некоторая плотность вероятности. Кластеры из близко расположенных в пространстве точек указывают на места с большей плотностью вероятности. Суть метода ядерных оценок состоит в том, что в каждой точке пространства размещается, чаще всего, функция Гаусса. Далее все эти функции складываются, что и дает оценку общей плотности вероятности. Если

наблюдений (обучающих примеров) достаточно много, то ядерные оценки дают хорошее приближение к истинной плотности вероятности.



Обозначения: \mathbf{p} — матрица размерности $R \times 1$ входного вектора из R компонентов (целые или действительные числа); $I \mathbf{W}^{1,1}$ — матрица размерности $S^1 \times R$ синаптических весов единственного слоя (карты) из S^1 нейронов; $ndist$ — блок вычисления отрицательного евклидова расстояния между вектором входа с матрицей \mathbf{p} и строками $I_i \mathbf{W}^{1,1}$ матрицы весов $I \mathbf{W}^{1,1}$; \mathbf{n}^1 — матрица размерности $S^1 \times 1$, элементы которой n_i^1 — отрицательные расстояния между входным вектором и вектором $I_i \mathbf{W}^{1,1}$; C — конкурирующая функция $compet$ (“Competition”) возвращает 1 для элемента выхода \mathbf{a}^1 , соответствующего победившему нейрону (все другие элементы вектора \mathbf{a}^1 равны 0)

Рис. 3.20 — Нейровычисления в обученной карте Кохонена

Таким образом, в отличие от многослойных искусственных нейронных сетей, реализующих при отображении входного множества в выходное глобальную аппроксимацию, в BNN применяется локальная аппроксимация. Она заключается в преобразовании путем адаптации нескольких аппроксимирующих функций к ожидаемым значениям, причем адаптация проводится только в ограниченной области многомерного пространства. На рис. 3.21 проиллюстрированы два способа разделения пространства данных: нейроном многослойных нейросетей и нейроном радиальных, в частности, BNN сетей.

Вероятностная сеть (рис. 3.22) имеет три слоя: входной, радиальный и выходной. Радиальные нейроны берутся по одному на каждый пример. Каждый из них имеет гауссовскую функцию с центром в этом примере. Каждому классу отвечает один выходной нейрон. Выходной эле-

мент соединен лишь с радиальными элементами, относящимися к его классу, и подытоживает выходы всех элементов, принадлежащих к его классу. Значения выходных сигналов получаются пропорционально ядерным оценкам вероятности принадлежности соответствующим классам. Важно отметить, что веса нейронов в суммирующем слое на данном этапе не рассчитываются, а происходит это уже на этапе нейровычислений с целью решения задачи классификации.

Вероятностная сеть имеет единственный управляющий параметр обучения, значение которого должно выбираться пользователем, — степень сглаживания (или отклонение функции Гаусса).

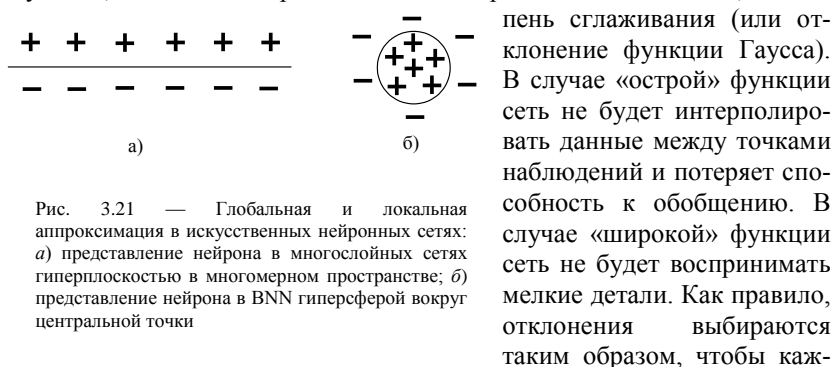


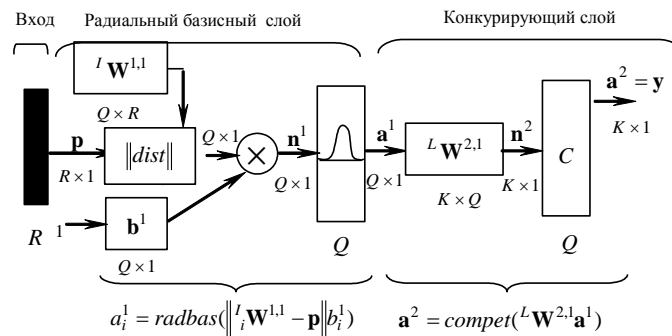
Рис. 3.21 — Глобальная и локальная аппроксимация в искусственных нейронных сетях: а) представление нейрона в многослойных сетях гиперплоскостью в многомерном пространстве; б) представление нейрона в BNN гиперсферой вокруг центральной точки

Суть этапа обучения, а это обучение с учителем, состоит в создании для каждого обучающего примера-образца нового нейрона, чей вес равняется значениям компонент образца и связывании этого нейрона с нейронами из радиального слоя. Обучающий пример состоит из вектора и идентификатора класса, к которому он относится.

Важная отличительная особенность BNN — минимальное количество расчетов на этапе обучения, поскольку веса нейронов в радиальном слое на данном этапе не рассчитываются, а происходит это уже на этапе нейровычислений с целью решения задачи классификации. Поэтому обучение идет очень быстро, не тратится время и машинные ресурсы на расчёт весов нейронов. Однако при этом расходуются значительные ресурсы на хранение данных всей обучающей выборки.

Нейровычисления в обученной вероятностной сети выполняются в соответствии с рис. 3.22.

Вероятностные нейронные сети применяются для моделирования решения экспертами задач классификации. Если задано достаточно большое обучающее множество, то результаты нейровычислений сходятся к решениям, соответствующим правилу Байеса.



Обозначения: \mathbf{p} — матрица размерности $R \times 1$ входного вектора из R компонентов (целые или действительные числа); Q — количество пар векторов вход/цели и количество нейронов в слое 1 (радиальном базисном слое); K — количество элементов вектора цели (число классов входных данных) и количество нейронов во втором слое (вектор цели преобразуется в матрицу связности \mathbf{T} размера $Q \times R$, строки которой обозначают классы, а столбцы — входные векторы; элемент $t(i, j) = 1$, если j — q -й входной вектор принадлежит i -му классу); ${}^I\mathbf{W}^{1,1}$ — матрица размерности $Q \times R$ синаптических весов первого слоя из Q нейронов; $dist$ — блок вычисления близости нового i -го входного вектора к векторам обучающего множества (векторов-строк ${}^I\mathbf{W}^{1,1}$) и формирования матрицы размерности $Q \times 1$; \otimes — умножение вычисленных в блоке $dist$ расстояний на смещения; $radbas$ — функция активации $f(x) = e^{-x^2}$; \mathbf{a}^1 — выходной вектор первого слоя (если входной вектор близок к вектору обучающего множества, то он представляется числом, близким к 1 как значение элемента вектора \mathbf{a}^1 ; если входной вектор близок к нескольким векторам некоторого класса, то он представляется несколькими элементами вектора \mathbf{a}^1 , значения которых будут близки к 1); ${}^L\mathbf{W}^{2,1}$ — матрица весов второго слоя, соответствующая матрице \mathbf{T} , построенной для данной обучающей последовательности; $\mathbf{n}^2 = \mathbf{T}\mathbf{a}^1$ — вектор, определяющий соответствие элементов вектора \mathbf{a}^1 каждому из K классов; C — конкурирующая функция $compet$ ("Competition") возвращает 1 для самого большого по значению элемента вектора \mathbf{n}^2 и 0 в остальных случаях

Рис. 3.22 — Нейровычисления в обученной вероятностной сети

Обобщая сказанное о методах интуитивно-образного рассуждения с использованием искусственных нейронных сетей, отметим, что коннекционистские (нейросетевые) [1, 157, 159, 162, 165 — 167] (рис. 3.23) представления отображают систему S в кибернетический «черный

ящик» со входом (X), перерабатывающий информацию и выдающий выходной сигнал (Y). Сетевая модель, состоящая из математических моделей биологических нейронов, способна обучаться на примерах, адаптируя свои параметры, извлекая и фиксируя знания из примеров. После обучения она способна, как «хороший» ученик, точно повторять действия учителя по преобразованию входа в выход. Использование термина «черный ящик» означает, что знания, извлекаемые алгоритмами обучения из элементарных количественных примеров функционирования системы S , становятся скрытыми, невидимыми (неконтрастными) для эксперта-управленца и специалиста-разработчика.

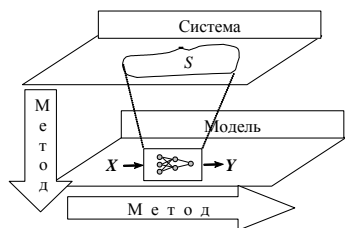


Рис. 3.23 — Нейросетевые методы

На базе нейросетевых представлений развиваются методы дискретной математики, нейроуправления, методы искусственного интеллекта.

Нейропредставления позволяют решать задачи идентификации, классификации, оптимизации, распознавания образов, прогнозирования и многие другие. Достоинства и недостатки нейросетей даны в табл. 1.12 и

[1].

Концептуальная модель для класса методов искусственных нейросетей с прямым распространением сигнала приведена ниже:

$$\begin{aligned} \dot{m}_n^a | T^e &= {}^1x_2^0 \circ {}^{11}r^n \circ {}^1x_1^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{14}^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{21}^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{34}^0 \circ \\ &\circ {}^1x_2^0 \circ {}^{13}r^n \circ {}^3x_{13}^0 \circ {}^1x_2^0 \circ {}^{13}r^n \circ {}^3x_{21}^0 \circ {}^1x_2^0 \circ {}^{18}r^n \circ {}^8\pi_n^h \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^2O_n^m, \end{aligned} \quad (3.24)$$

$${}^8\pi_n^h = A \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2K_n^h \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2O_n^h \circ B,$$

где ${}^8\pi_n^h$ — схема ролевых концептуальных моделей задач, решаемых нейросетями; ${}^2K_n^h$ — включает фазу: «контроль», «анализ», «прогнозирование», «регулирование», переменные — «детерминированные», «стохастические»; класс задач — (см. табл. 7.2 в [1]); переменные-отношения — «сравнения»; ${}^2O_n^h$ — специфицируется в ходе разработки; A и B — фрагменты (3.24), уточняемые в ходе разработки; ${}^2O_n^m$ — спецификатор, включающий: знания о возможной погрешности решения в целом для класса методов — в пределах ошибки обучения и тестирования; знания о гибридных возможностях даны в табл. 8.6 [1].

Модель нейровычислений в сети прямого распространения сигнала приведена ниже:

$$\dot{m}_n^a \Big|^{Ca} = \langle Ar, X, Y, M^n, Ed, I^{n1}, I^{n2} \rangle, \quad (3.25)$$

где Ar — архитектура (топология) нейросети; X, Y — множества входов и выходов нейросети соответственно; M^n — множество моделей искусственных нейронов; Ed — обучающая и тестирующая последовательности; $I^n = \langle I^{n1}, I^{n2} \rangle$ — интерпретаторы обучения и нейровычислений соответственно.

Модель искусственного нейрона $\dot{m}^n \in M^n$ может быть представлена как $\dot{m}^n = \langle X^n, y \in Y, Sw, f^\Sigma(x), f^A(x) \rangle$, где $X^n \subseteq X$ — множество входов нейрона; y — выходной сигнал; Sw — матрица-столбец си-напсических весов; $f^\Sigma(x)$ — функция суммирования и $f^A(x)$ — функ-ция активации.

Архитектуру сети представим графом Бержа $Ar = (\Delta, \Psi)$, где $\Delta = \{\Delta_1, \Delta_2, \Delta_3\}$ — множество вершин, а $\Psi = \{\Psi_6, \dots, \Psi_{12}\}$ — множество взаимнооднозначных соответствий: $\Psi_6: X \rightarrow \Delta_1$, $\Psi_7: M^n \rightarrow \Delta_2$, $\Psi_8: Y \rightarrow \Delta_3$, $\Psi_9: X \rightarrow X^n$, а также отображений: $\Psi_{10}: \Delta_1 \rightarrow \Delta_2$, $\Psi_{11}: \Delta_2 \rightarrow \Delta_2$, $\Psi_{12}: \Delta_2 \rightarrow \Delta_3$. Отображения Ψ_{10}, Ψ_{12} можно задать взвешенными матрицами смежности графа Ar , что позволяет учесть си-напсические веса нейронов. Отображение Ψ_{11} задает слоистую струк-туру сети, когда отсутствуют связи между нейронами разных слоев.

Элемент Ed в (3.25) включает обучающую и тестирующую после-довательности, используемые в процессах I^{n1} и I^{n2} соответственно.

Процессы I^{n1} и I^{n2} выполняются в следующей последовательности:

$$I^{n1} \rightarrow I_{\uparrow}^{n2}, \quad (3.26)$$

где символ « \uparrow » обозначает, что после обучения для нейровычислений достаточно использовать только I^{n2} .

Модель вычислений в картах Кохонена рассмотрена в [1].

3.3.4 Методы эволюционных вычислений

Использование интуитивного мышления экспертов и ЛПР в СППР для решения задач неразрывно связано с умением правильно применять генетические знания. Творческие процессы в СППР, как отмечалось в разд. 3.1, имеют «генетический» компонент.

Генетические знания людей и знания, накопленные в течение их индивидуальной жизни, передаются потомкам. Введение результатов индивидуальной адаптации в генетические знания ещё недостаточно изучено, но оно, несомненно, существует, возможно, посредством эпигенетических изменений работы генов [140].

Вся эта информация передается также посредством хорошо известных первой (сигналы, изотропные действия) и второй (сигналы, не-изотропные действия) сигнальных систем, а также малоизвестной третьей сигнальной системы, действующей посредством сигналов электромагнитных биополей.

Роль отбора сильнейших представителей популяции ярко проявляется на примере эволюции видов, а также в социальных процессах, приводящих к изменению и формированию культуры. Эти процессы формализованы с помощью теории клеточных автоматов, генетических алгоритмов, генетического программирования, искусственной жизни и других форм эмерджентных вычислений.

Эмерджентные модели (англ. emergent models) обучения имитируют наиболее мощную форму адаптации — эволюцию форм жизни растительного и животного мира. Как отмечал Чарльз Дарвин, нет границ этой энергии, которая медленно, но верно адаптирует каждое создание к самым сложным жизненным отношениям.

Генетические алгоритмы и другие эмерджентные аналоги обуславливают более точное решение задачи за счет операций над популяциями кандидатов на роль решения.

Идея генетических алгоритмов заимствована у живой природы и состоит в организации эволюционного процесса, конечная цель которого — получение оптимального решения в сложной комбинаторной задаче. Разработчик генетических алгоритмов выступает в данном случае как «создатель», который должен правильно установить законы эволюции, чтобы достичь желаемой цели как можно быстрее.

Идею генетических алгоритмов предложил Дж. Холланд (J. Holland) на рубеже шестидесятых и семидесятых годов. В 1975 г. им была опубликована книга “Adaptation in Natural and Artificial Systems” [168]. Описание ранней истории ГА дано в книге Д. Голдберга “Genetic Algorithms in Search, Optimization and Machine Learning” [169]. ГА вместе с эволюционными стратегиями (англ. evolution strategies, Германия) и эволюционным программированием (англ. evolutionary programming, США) —

одно из трех главных направлений развития так называемой имитированной эволюции (англ. simulated evolution).

Общую схему эволюционных вычислений с помощью генетических алгоритмов проще всего понять, рассматривая задачи безусловной оптимизации: $\max\{f(i) \mid i \in \{0,1\}^n\}$. Примерами служат задачи размещения, стандартизации, выполнимости и др.

Классический генетический алгоритм начинает свою работу с формирования начальной популяции $I_0 = \{i_1, i_2, \dots, i_{N_t}\}$ — конечного набора допустимых решений задачи. Эти решения могут быть выбраны случайным образом или получены с помощью вероятностных жадных алгоритмов. Выбор начальной популяции не имеет значения для сходимости процесса, однако формирование «хорошей» начальной популяции, например, из множества локальных оптимумов, может сократить время достижения глобального оптимума.

На каждом шаге эволюции с помощью вероятностного оператора селекции выбираются два решения — родители: i_1, i_2 . Оператор скрещивания по решениям i_1, i_2 строит новое решение i' , которое затем подвергается небольшим случайным модификациям, которые принято называть мутациями. Затем решение добавляется в популяцию, а решение с наименьшим значением целевой функции удаляется из популяции. Общая схема такого алгоритма может быть записана следующим образом.

- 1) Выбрать начальную популяцию I_0 и положить $f^* = \max\{f(i) \mid i \in I_0\}$, $k := 0$.
- 2) Пока не выполнено условие останова, делать следующее:
 - 2.1) Выбрать родителей i_1, i_2 из популяции I_k (селекция).
 - 2.2) Построить i' по i_1, i_2 (скрещивание).
 - 2.3) Модифицировать i' (мутация).
 - 2.4) Если $f^* < f(i')$, то $f^* := f(i')$.
 - 2.5) Обновить популяцию и положить $k := k+1$.

Остановимся подробнее на основных операторах этого алгоритма: селекции, скрещивании и мутации.

Среди операторов селекции наиболее распространены методы пропорциональной и турнирной селекции. При пропорциональной селекции вероятность на k -м шаге выбрать решение i в качестве одного из родителей задается формулой: $P(i|_{\text{выбрано}}) = f(i) / \sum_{j \in I_k} f(j)$, $i \in I_k$, в пред-

положении, что $f(i) > 0$ для всех $i \in I$. При турнирной селекции формируется случайное подмножество из элементов популяции и среди них выбирается один элемент с наибольшим значением целевой функции. Турнирная селекция имеет определенные преимущества перед пропор-

циональной, так как не теряет своей избирательности, когда в ходе эволюции все элементы популяции становятся примерно равными по значению целевой функции. Операторы селекции строятся таким образом, чтобы с ненулевой вероятностью любой элемент популяции мог бы быть выбран в качестве одного из родителей. Более того, допускается ситуация, когда оба родителя представлены одним и тем же элементом популяции.

Как только два решения задачи выбраны, к ним применяется вероятностный оператор скрещивания (англ. crossover) — п. 2.2 алгоритма. Существует много различных версий этого оператора [169], среди которых простейший — однородный оператор. По решениям i_1, i_2 он строит решение i' , присваивая каждой координате этого вектора с вероятностью 0,5 соответствующее значение одного из родителей. Если вектора i_1, i_2 совпадали, скажем, по первой координате, то вектор i' «унаследует» это значение. Можно сказать, что оператор скрещивания старается выбрать новое решение i' где-то между i_1, i_2 , полагаясь на удачу. Наряду с однородным оператором скрещивания применяются и другие операторы, использующие не два, а произвольное число решений из популяции.

Оператор мутации, применяемый к решению i' в п. 2.3 генетического алгоритма, с заданной вероятностью $p_m \in (0, 1)$, изменяет значения всех координат на противоположное. Например, вероятность того, что $i' = (0, 0, 0, 0, 0)$ в ходе мутации перейдет в $j' = (1, 1, 1, 0, 0)$, равна $p_m \times p_m \times p_m \times (1 - p_m) \times (1 - p_m) > 0$. Таким образом, с ненулевой вероятностью p_m решение i' может быть преобразовано в любое другое решение.

Таким образом, генетический алгоритм выполняет перестройку решения задачи (операторами скрещивания и мутации) и «сосредотачивается» на поиске глобального экстремума среди локальных, так как экспериментальные исследования распределения локальных оптимумов свидетельствуют о высокой концентрации их в непосредственной близости от глобального оптимума [170]. Это наблюдение известно как гипотеза о существовании «большой долины» для задач на минимум или «центрального горного массива» для задач на максимум.

В ходе функционирования классического генетического алгоритма значение средней приспособленности (в непрерывном случае) популяции решений растет и с некоторого шага остается приблизительно постоянным. Это значение и принимается за искомое решение задачи.

Эволюционные вычисления в модификациях классического генетического алгоритма. Если генетический алгоритм использует бинарное представление индивидуумов-решений, селекцию методом рулетки, скрещивание с одной точкой разрыва кода и полное обновление популяции после цикла «оценка—селекция—скрещивание—мутация»,

он называется классическим, а поскольку каждый такой цикл рассматривается как генерация (порождение) популяции, то его все чаще называют генерирующим генетическим алгоритмом.

К настоящему времени разработано много модификаций классических, генерирующих генетических алгоритмов. Такие модификации имеют целью ликвидировать такое отрицательное явление, как преждевременная сходимость, ускорить расчеты за счет параллельных вычислений функций оценки решений, расширить класс задач, например, охватить многокритериальную оптимизацию, скорректировать модель наследования за счет учета взаимной эволюции между популяциями, наконец, гибридизировать генетические алгоритмы с другими алгоритмами поиска. Ниже приведен краткий обзор возможных вариантов модификации.

Генетические алгоритмы с элитарной стратегией репродукции. Элитаризм — один из способов преодоления преждевременной сходимости. Он основан на защите наилучших решений в очередных итерациях. В генерирующих генетических алгоритмах наилучшие индивидуумы-решения не всегда переходили в новую популяцию $P(n+1)$. В элитарных алгоритмах наилучшее решение всегда (т.е. принудительно) включается в новую популяцию. Но даже это не может полностью исключить потерю полезных индивидуумов. Поэтому были предложены алгоритмы с установившимся режимом.

Генетические алгоритмы с установившимся режимом. В этих алгоритмах эволюционных вычислений имеется параметр n , определяющий количество наихудших решений, замещаемых в новой популяции. Эти алгоритмы особенно эффективны, если нужно сохранить скорректированные полезные копии решений.

Генетические алгоритмы с турнирной селекцией. В настоящее время разработано много методов селекции: рулетки, Бейкера (Baker), рейтинговый, турнирный и др. Особую популярность имеет метод турниров (англ. tournament selection). В турнирной селекции индивидуумы-решения популяции делятся на подгруппы. Далее в каждой из них выбирается индивидуум с наилучшей приспособленностью. Возможны два способа такого выбора: детерминированный и вероятностный. В детерминированном случае выбор выполняется с вероятностью 1, а в случае стохастического — с вероятностью, меньшей 1. Подгруппа решений может быть произвольного размера, однако наиболее часто она делит популяцию на подгруппы, состоящие из двух — трех индивидуумов. Турнирный метод применяется как для задач максимизации, так и минимизации функции. Кроме того, он может быть легко расширен на задачи, касающиеся многокритериальной оптимизации.

Генетические алгоритмы с многоточечным скрещиванием. В классическом алгоритме используется так называемое однотоочечное скрещивание. Его рекомендуется применять для задач, в которых функция оценки характеризует пространственную, дискретную структуру, например, для задач раскраски графов. Оно рассмотрено выше.

Однако есть и другие виды скрещивания: двухточечное, многоточечное, равномерное [157, 159]. При двухточечном скрещивании (англ. two point crossover) решения-потомки наследуют фрагменты родительских решений, назначаемых через две разыгранные точки скрещивания (рис. 3.24, а).

Многоточечное скрещивание (англ. multi-point crossover) — это обобщение предыдущей операции, характеризующееся большим количеством точек (в частности — трех) скрещивания (рис. 3.24, б). Аналогично происходит скрещивание для пяти, а также большего нечетного числа точек скрещивания. Однотоочечное скрещивание — частный случай такого скрещивания.

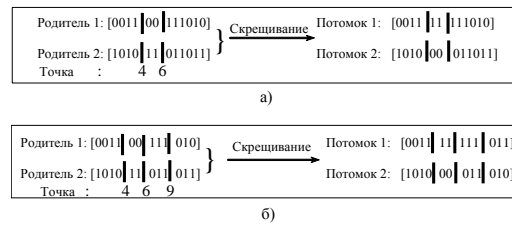


Рис. 3.24 — Пример скрещивания: а) — двухточечного и б) — многоточечного

Генетические алгоритмы с нишами. В задачах с целевыми функциями, имеющих несколько критериев, иногда требуется найти не только глобальный оптимум, но и локальные экстремумы. В классическом генетическом алгоритме это сделать трудно из-за «генетического дрейфа», когда индивидуумы-решения имеют тенденцию концентрироваться в наивысших пиках эволюционного процесса (рис. 3.25, а). Точнее, этот феномен возникает вследствие схождения конечной популяции, когда давление селекции снижается.

Предложено несколько генетических алгоритмов для оптимизации многомодальных функций. Большинство таких генетических алгоритмов формируют и поддерживают стабильные подпопуляции вокруг различных экстремумов по аналогии с биологическим феноменом ниш (рис. 3.25, б).

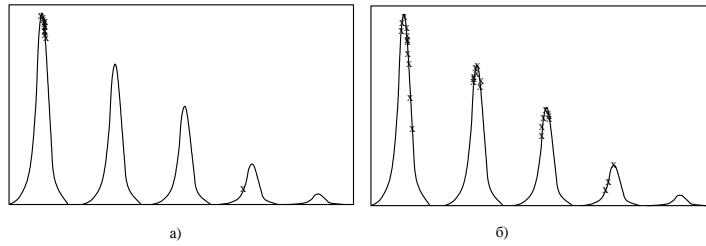


Рис. 3.25 — Работа генетического алгоритма классического — а) и с нишами — б)

Применяемый на практике метод создания ниш в генетических алгоритмах использует так называемую функцию участия (англ. sharing function). Эта функция определяет близость и степень участия для каждого решения в популяции. Функция участия обозначается $h(\gamma_{ij})$, где λ_{ij} — мера близости между индивидуумами i_l и i_j . Например, эта близость может быть определена по следующей формуле:

$$\gamma_{ij} = \sqrt{\sum_{k=1}^{N_{\pi}} (x_{k,l} - x_{k,j}) / (x_{k,\max} - x_{k,\min})^2},$$

где N_{π} — число параметров задачи; $x_{k,\min}, x_{k,\max}$ — обозначают, соответственно, минимальное и максимальное значения k -го параметра; $x_{k,l}, x_{k,j}$ — обозначают, соответственно, k -й параметр l -го и j -го решения.

Функция участия $h(\gamma_{ij})$ должна иметь следующие свойства: 1) $0 \leq h(\gamma_{ij}) \leq 1$ для $\forall \gamma_{ij}$; 2) $h(0) = 1$; 3) $\lim_{\gamma_{ij} \rightarrow \infty} h(\gamma_{ij}) = 0$. Одна из многих функций, удовлетворяющих этим условиям, приведена ниже:

$$h(\gamma_{ij}) = \begin{cases} 1 - (\gamma_{ij} / \sigma_h)^\omega, & \text{если } \gamma < \sigma_h, \\ 0, & \text{иначе,} \end{cases}$$

где σ_h и ω — константы. Например, $\sigma_h = 0,5 \cdot q^{-1/N_{\pi}}$, где q — предполагаемое число пиков оптимизируемой функции; ω можно принять равным единице, что дает равную степень участия между соседними индивидуумами-решениями.

Тогда функция оценки индивидуума i_l может быть задана формулой:

$$F_h(i_l) = F(i_l) / \sum_{j=1}^{N_l} h(\gamma_{ij}), \quad (3.27)$$

где N_l — число индивидуумов-решений в популяции.

Если решение i_l только одно в своей нише, то $F_h(i_l) = F(i_l)$. В противном случае значение функции оценки убывает пропорционально количеству индивидуумов в нише, а также близости соседних решений. Из формулы (3.27) следует, что рост числа похожих кодов, т.е. входящих в одну и ту же нишу, ограничен, так как этот рост вызывает уменьшение значения функции оценки.

Существуют также другие модификации описанного выше метода ниш [157, 159].

Генетические алгоритмы со взаимной эволюцией. В обычной эволюции одиночная популяция развивается под давлением селекции на основе заданной функции оценки, играющей роль среды. Однако в природе среда данной популяции реально охватывает физическую среду, которая обычно меняется очень медленно, а также и другие биологические популяции. Взаимодействие между популяциями вездесуще. Тогда имеет смысл рассматривать эволюцию как процесс взаимной эволюции, т.е. когда изменения в одной популяции влияют на другую популяцию. В этом случае говорят, что среда изменяется. Преимущество таких алгоритмов на основе взаимной эволюции в том, что они не нуждаются в обязательном определении глобальной функции оценки, посредством которой члены популяции ранжируются. Достаточно только относительных оценок. Это может быть полезным, когда получение адекватной функции оценки для данной задачи затруднено или даже невозможно. Что характерно, например, для сложных игр.

Гибридные генетические алгоритмы. В сложных задачах СППР очень часты случаи, когда эффективные методы поиска, использующие проблемно-ориентированные представления, уже существуют. Гибридные алгоритмы используют встроенные быстрые и эффективные проблемно-ориентированные поисковые процедуры. Сила гибридных эволюционных вычислений лежит в комбинации двух различных комплементарных поисковых принципов: эволюционная часть выполняет широкий поиск в пространстве решений, а локальный метод, используя знания о решаемой задаче, выполняет поиск в перспективной, более узкой области в глубину.

Параллельные эволюционные вычисления. Есть два главных резона разработки и применения параллельных генетических алгоритмов. Один из них — экономия машинного времени для выполнения расчетов. Второй — извлечь пользу из естественной параллельной эволюции пространственно распределенных популяций. Для многих задач СППР расчет значений функции оценки — наиболее медлительный шаг вычислений. В параллельных генетических алгоритмах расчет функции

оценки отдельного индивидуума может быть передан одному из процессоров многопроцессорной системы. Если в популяции число членов больше, чем процессоров, то значение функции может вычисляться первым освободившимся процессором. В других подходах одновременно и независимо работает N копий алгоритма на N доступных процессорах. Лучшая из многих независимых попыток — требуемый результат.

Сформировались два класса моделей параллельных генетических алгоритмов — модель с островками [171] и сеточная (рис. 3.26) или мелкозернистая модель [172]. Так например, в сеточной модели индивидуумы-решения размещены на одно- или двухтороидальной сетке. Один индивидуум находится на одной позиции (рис. 3.26).

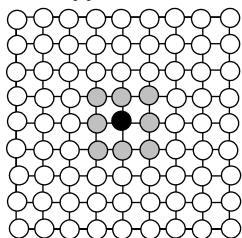


Рис. 3.26 — Сеточная модель в параллельных генетических алгоритмах

Расчет функции оценки выполняется одновременно для всех решений, а селекция, скрещивание и мутация имеют место в пределах небольшого соседства (на рис. 3.26 оно выделено серым цветом). Через некоторое время полуизолированные ниши генетически однородных индивидуумов перемешиваются по сетке как результат медленной диффузии индивидуумов-решений. Этот феномен назван «изоляцией расстоянием» (англ. *isolation by distance*) и обязан тому факту, что вероятность взаимодействия двух

решений зависит от их расстояния.

Отметим, что гибридные модели также возможны и в парадигме параллельных генетических алгоритмов. Например, можно рассматривать модель островков, в которой каждый остров — сетка взаимодействующих локально индивидуумов.

Генетические микроалгоритмы. Генетический микроалгоритм — это модификация классического генетического алгоритма, предназначенная для задач СППР, не требующих больших популяций и длинных кодов-решений. Такие алгоритмы применяются в случаях, когда решение должно быть найдено быстро, но оно не есть глобальный оптимум. Микроалгоритмы работают немного хуже, зато экономят машинное время. В них размер популяции очень маленький и заранее определенный. Применяется элитарная стратегия, что предотвращает потерю хорошего индивидуума. Поскольку популяция — маленькая, то селекция осуществляется в соответствии с детерминированным способом. Скрещивание выполняется с вероятностью 1. В мутации нет необходимости, поскольку достаточная разнородность вводится выбором новой популяции в каждом «рестарте» алгоритма.

Генетические алгоритмы для многокритериальной оптимизации — один из бурно развивающихся, особенно в последние 10 лет, классов эволюционных вычислений. Главное место здесь занимает многокритериальная оптимизация по методу Парето. В многокритериальной оптимизации отыскивается не единственный индивидуум, как закодированный вид оптимального решения в обычном смысле, а как множество оптимальных кодов Парето.

На рис. 3.27 [173] приведена популяция решений до (размытая область) и после (вытянутое в линию множество индивидуумов в нижней части размытой области) формирования фронта Парето. Эксперт, получив такое множество, может уже выбрать из него наилучшее, с его точки зрения, решение.

Применение генетических алгоритмов для поиска оптимальных Парето-решений должно выглядеть, казалось бы, просто. На одной и той же

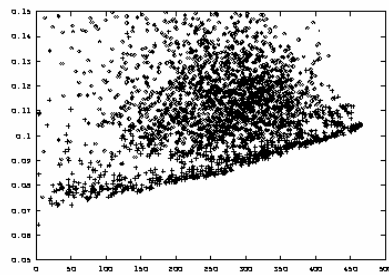


Рис. 3.27 — Исходная популяция и оптимальный фронт Парето после работа генетического алгоритма

исходной популяции обрабатывает несколько раз классический генетический алгоритм, но с разными функциями оценки. В этом случае получаем множество оптимальных по определенным критериям популяций. Затем эти популяции перемешиваются в одну, в которой с использованием приведенных выше соотношений обычной вычислительной процедурой отыскивается фронт несдоминированных решений.

Однако на этом простом пути,

как показали наши исследования и, например, [173] много проблем, рассмотрение которых выходит за пределы этой книги. В работе [173] предложен специальный класс алгоритмов с нишами для многокритериальной оптимизации по методу Парето. Его специфика состоит в особенностях реализации селекции. В этих алгоритмах применяется особый вид селекции «турнирная селекция с доминированием Парето», которая на турнире размером два индивидуума задает частичный порядок, определяемый вышерассмотренными отношениями доминирования Парето. Для более тщательного анализа фронта Парето применяются ниши.

Обобщая сказанное о методах эволюционного моделирования решения задач экспертами, отметим, что можно говорить о еще одном бурно

развивающемся в настоящее время классе представлений — эволюционных представлениях, которые условно изображены на рис. 3.28.

Особенность таких методов — абстрагирование от многих свойств системы S и отображение ее инициализируемой популяцией индивидуумов-кодов, чаще — бинарных. Решение задачи рассматривается как адаптивный процесс селекции, применения генетических операторов скрещивания, мутации и инверсии, что повторяется до нахождения лучших индивидуумов-решений.

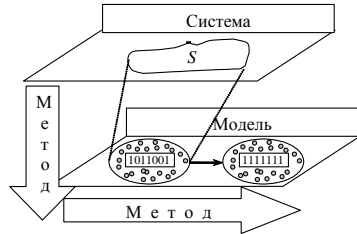


Рис. 3.28 — Эволюционные методы

На базе эволюционных представлений развиваются методы дискретной математики, исследования операций в технических, а также совершенствования в био-производственных и экосистемах, методы искусственного интеллекта. Эволюционные представления перспективны и применяются в системах автоматизированного проектирования для решения задач компоновки, размещения, трассировки соединений, распознавания образов и классификации. Генетические алгоритмы имеют хороший потенциал для многокритериальной оптимизации, в частности, маршрутизации, распределения ресурсов, составления расписаний. Эволюционные представления эффективно используются во многих комбинаторно-логических задачах на графах. Достоинства и недостатки ГА даны в табл. 1.12 и [1].

Концептуальная модель для класса классических ГА приведена ниже:

$$\begin{aligned} \dot{m}_g^a | Te = & \overset{1}{x_2} \overset{0}{11} \overset{n}{r} \overset{1}{x_1} \overset{0}{x_2} \circ \overset{1}{x_2} \overset{12}{r} \overset{n}{22} \overset{0}{x_{15}} \circ \overset{1}{x_2} \overset{12}{r} \overset{n}{22} \overset{0}{x_{26}} \circ \overset{1}{x_2} \overset{12}{r} \overset{n}{22} \overset{0}{x_{32}} \circ \\ & \circ \overset{1}{x_2} \overset{13}{r} \overset{n}{3} \overset{0}{x_{16}} \circ \overset{1}{x_2} \overset{18}{r} \overset{n}{8} \overset{h}{\pi}_g \circ \overset{1}{x_2} \overset{12}{r} \overset{n}{2} \overset{m}{O}_g, \\ & \overset{8}{\pi}_g^h = A \circ \overset{8}{x_1} \overset{0}{82} \overset{n}{R} \overset{2}{K}_g^h \circ \overset{8}{x_1} \overset{0}{82} \overset{n}{R} \overset{2}{O}_g^h \circ B, \end{aligned} \quad (3.28)$$

где $\overset{8}{\pi}_g^h$ — схема ролевых концептуальных моделей задач, решаемых генетическими алгоритмами; $\overset{2}{K}_g^h$ — включает фазу — «анализ», «прогнозирование», «организация», переменные — «генетические», «стохастические», класс задач — (см. табл. 7.2 в [1]); переменные-отношения — «сравнения»; $\overset{2}{O}_g^h$ — специфицируется в ходе разработки; A и B — фрагменты (3.28), уточняемые в ходе разработки; $\overset{2}{O}_g^m$ — спецификатор,

включающий: знания о возможной погрешности решения в целом для класса методов — в пределах ошибки обучения и тестирования; знания о гибридных возможностях даны в табл. 8.6 [1].

Модель вычислений с использованием классических генетических алгоритмов имеет следующий вид:

$$\dot{m}_g^a |^{Ca} = \langle KB, N, f(i^{10}), I^g \rangle, \quad (3.29)$$

где $KB = \{i_1, \dots, i_{N_{KB}}\}$ — база знаний [133, 174] системы, работающей как генетический алгоритм, представляющая собой эволюционирующую во времени (от итерации к итерации) популяцию индивидуумов; $I_j = g_1 \circ g_2 \circ \dots \circ g_{N_I} | j = 1, \dots, N_{KB}$ — бинарный код — микроуровневое представление индивидуума, часто называемое [174] генетической переменной, где $g_q | q = 1, \dots, N_I$ — ген, принимающий значения 1 или 0; N_I — число генов в решении; N — число итераций алгоритма; $f(i^{10})$ — функция оценки приспособленности индивидуумов, где i^{10} — макроуровневое (десятичными числами) представление индивидуумов из KB .

Интерпретатор I^g , используя $f(i^{10})$, обрабатывает KB и есть совокупность процессов: $I^g = \langle I^{g1}, \dots, I^{g7} \rangle$, где I^{g1} — инициализация, когда из пространства поиска случайным образом выбирается начальная популяция индивидуумов; I^{g2} — кодирование индивидуумов, т.е. перевод из макро- в микроуровневое представление; I^{g3} — оценка индивидуумов в соответствии с функцией $f(i^{10})$; I^{g4} — селекция индивидуумов, т.е. отбор родителей для скрещивания; I^{g5} — формирование родительской пулы; I^{g6} — рекомбинация бинарных кодов операциями скрещивания, мутации, инверсии [1] и формирования новой текущей популяции; I^{g7} — проверяет условие окончания работы алгоритма и отображает наилучшего за N итераций индивидуума-решения.

Процессы I^{g1}, \dots, I^{g7} выполняются в следующей последовательности:

$$I^{g1} \rightarrow I^{g2} \rightarrow I^{g3} \rightarrow I^{g4} \rightarrow I^{g5} \rightarrow I^{g6} \rightarrow I^{g7} \rightarrow \quad (3.30)$$

3.4. Моделирование профессионального опыта экспертов

Формирование типовых вариантов решений в «семантической каше» человеческого сознания и использование типовых приемов выбора решений — главные условия ускорения мыслительного процесса экспертов в СППР. «Типовой» не значит «известный». «Типовое» — это айсберг сознания, а «известное» — верхушка этого айсберга. Граница «известного» подвижна и является важнейшей отличительной характеристикой индивида. В понятие «типовое» решение вкладывается принципиальная возможность осознания экспертом посылок и дальнейшего многократного использования этого решения.

Описание посылок-условий принято называть дескриптивными моделями, описывающими объект управления, а использование решений для соответствующих действий — прескриптивными моделями, предписывающими те или иные действия. Дескриптивные модели описывают структуры данных и сами данные, к которым применяются прескриптивные модели. Применение прескриптивной модели к дескриптивной модели порождает рассуждения так же, как в школьном учебнике применение типового способа решения задачи к описанию задачи порождает решение задачи. Дескриптивные модели рождаются в подсознании индивида, но используются исключительно осознанной прескриптивной мыслительной деятельностью. Напротив, некоторые прескриптивные модели остаются навсегда под «ватерлинией айсберга», они недоступны для сознания человека, и направляют его неосознанные, интуитивные действия.

Любой осознаваемый стереотип может быть представлен отношениями между дескриптивной и прескриптивной моделями и в этом его коренное отличие от неосознаваемого стереотипа.

В силу свойств человеческого мозга, рассуждения эксперта, включая управленческие решения, — типовые и базируются на осознаваемых или неосознаваемых моделях. Источником стереотипов могут служить внутренние и (или) внешние носители стереотипов, используемые для поддержки принятия решений.

«Библиотека» стереотипов — это множество сопоставленных друг другу пар дескриптивных и прескриптивных моделей, это сложно организованное хранилище коллективного опыта СППР, формируемого в процессе коллективного обучения. Конечно, индивид способен принимать участие в организации библиотеки стереотипов лишь в той ее части, где стереотипы осознаны им.

Осознанный и неосознанный эвристический поиск соответствующих наилучшим образом друг другу дескриптивных и прескриптивных моделей и совместное их использование, по сути, и представляют собой

рассуждения, приводящие к получению решений задачи. Это идеальная технократическая модель эксперта на основе опыта решения задач.

Подобная схема стереотипных рассуждений позволяет эксперту напрямую использовать исторический опыт, а разработчику отказаться от всестороннего анализа знаний о данной области и трудоемкого и во многом противоречивого процесса извлечения знаний из экспертов. Появляется возможность сокращения рассуждений. Если известен аналогичный случай, новые задачи решаются гораздо быстрее, чем применением рассуждений на основе профессиональных знаний или теоретических моделей.

Рассуждения на основе прецедентов характерны для таких предметных областей, как: история, юриспруденция и программирование для ЭВМ. Архитекторы также используют свои знания об эстетической привлекательности и полезности существующих сооружений для разработки новых зданий, которые будут восприняты современниками как красивые и удобные. Еще одна область рассуждений на основе опыта — проектирование судов, где успешно выполненный проект может использоваться в новой ситуации, когда верфи заказывается новое судно. Аналогии используются в диагностике, где на ум, прежде всего, приходят ошибки прошлого, управлении сложными установками и финансами. Так, например, мастер бурения, имеющий опыт двух ситуаций-взрывов, быстро вспоминает одну или обе ситуации, когда совокупность критических параметров установки становится похожей на ситуации-взрывы. Заведующая отделом кредитования в банке, решающая сложную задачу выделения клиенту кредита, вспоминает предыдущий случай, который вовлек ее финансовую организацию в похожие хлопоты, и рекомендует отказаться от выдачи ссуды.

Как показывают рассмотренные выше примеры, рассуждения на основе опыта — мощный и часто используемый путь решения задач людьми. Эта претензия обоснована и по результатам когнитивных психологических исследований, т.е. имеет психологическое правдоподобие. Различные исследователи приводили эмпирические доказательства о доминирующей роли специфических, имевших место в прошлом ситуаций, — опыта (названного прецедентами, кейсами) в решении задач человеком. Р. Шенк (R. Shank) [175] разработал теорию обучения и напоминания, основанную на сохранении в динамике жизненного опыта путем развития структуры памяти. Дж. Андерсон (J. Anderson) показал, что люди применяют прошлый опыт как модели, когда обучаются решать задачи, особенно в молодости. Другие исследования [137] показывают, что использование прошлого опыта — доминирующий метод решения задач у экспертов.

Подходы к таким рассуждениям позволяют КСППР учиться на своем опыте. Вновь найденное решение задачи может быть сохранено, и им можно воспользоваться в новой ситуации. Таким образом, рассуждения эксперта на основе опыта — мощный дидактический инструмент.

Для моделирования рассуждений эксперта и обучения на основе опыта используется механизм, названный рассуждениями на основе прецедентов (англ. case-based reasoning — CBR). Программные системы, в которых такой механизм реализован, называются CBR-системами.

CBR — это парадигма решения задач, которая фундаментально отличается от других подходов искусственного интеллекта к моделированию рассуждений экспертов. Вместо исключительного доверия профессиональным знаниям о предметной области, сформулированным в виде «условие—действие», «описание задачи — заключение», прецедент-системы способны использовать специфические знания, предварительно примененные в конкретных ситуациях решения задач. Новая задача решается путем нахождения прецедента в библиотеке прецедентов и его повторного использования в новой проблемной ситуации. Вторая отличительная особенность состоит в том, что прецедент-системы — это подход к непараметрическому обучению, позволяющему увеличивать сложность результата решения задачи (гипотезы) по мере роста объема данных. Чем больше данных поступает в распоряжение эксперта, тем более развитой может становиться гипотеза.

Итак, если задать себе вопрос: «Что такое рассуждения на основе прецедентов в прецедент-системах?», то можно ответить так — это решение задачи с помощью эвристического поиска запомненных ранее похожих ситуаций и посредством повторного использования информации и знаний об этих ситуациях.

Очень часто рассуждения на основе прецедентов и рассуждения по аналогии считают синонимичными понятиями. Однако в работе [137] показано, что есть и отличия. Рассуждения на основе опыта следует рассматривать как рассуждения по аналогии в пределах специфики конкретной предметной области, а рассуждения по аналогии — это рассуждения эксперта «крест-накрест», т.е. с использованием информации о нескольких предметных областях.

Терминология прецедент-систем основывается на понятии прецедента, аналога, кейса — проблемной ситуации. Кейсы различаются как прошлый, предыдущий, сохраненный, удерживаемый в памяти опыт (англ. past, previous, stored, retained case) и как новый, нерешенный случай (англ. new, unsolved case).

Истоки данного подхода следует искать в работах Р. Шенка [176] по динамической памяти и центральной роли запоминания прошлых ситуаций и ситуационных образов (сценариев) в принятии решений и обуче-

нии. Другие истоки — это исследования рассуждений по аналогии, теории формирования понятий, принятия решений и обучения [137].

Первая система, которая может быть отнесена к CBR-системам, — это вопросно-ответная система CYRUS, созданная Ж. Колоднер (J. Kolodner) в 1983 г. [177, 178], для работы со знаниями о командировках и встречах государственного секретаря США. Среди первых результатов работ по моделированию рассуждений на основе опыта находятся системы MEDIATOR [179], CHEF [180], PERSUADER [181]. В Европе разработки начались несколько позже, чем в США. Здесь первыми были системы PATDEX [182] и REFINER [183].

Методы моделирования рассуждений экспертов на основе опыта в CBR-системах. Функционально все CBR-системы идентифицируют текущую проблемную ситуацию, ищут прошлый прецедент, похожий на новый, исправляют этот старый прецедент, чтобы принять решение по текущей задаче, оценивают предполагаемое решение, обновляют базу (библиотеку) прецедентов путем обучения. Для выполнения этих действий используется несколько основных методов.

Метод рассуждений на основе экземпляров (англ. exemplar-based reasoning). Этот термин — следствие различия во взглядах на проблему формирования понятия. Таких подходов несколько: классический подход, вероятностный подход и «экземплярный» подход. В последнем случае понятие определяется расширенно (экстенционально), как множество экземпляров-объектов. Например, понятие агрофирма определяется только через свой объем, т.е. множество объектов, экземпляров реального мира, обозначаемых в сознании эксперта понятием: «агрофирма» = {агрофирма 1, агрофирма 2, ..., агрофирма N}. При использовании этого метода рассуждений решить задачу — это значит решить задачу классификации, т.е. сопоставить понятие классу понятий. Таким образом, данный метод использует термины — классы прецедентов, понятия-экземпляры прецедентов и отношения «класс-экземпляр». Множество классов понятий трактуется как множество возможных решений. Как только распознается класс, к которому относится прецедент-экземпляр, — это однозначно идентифицирует решение. При использовании данного подхода изменить решение нельзя, т.е. исправить понятие-класс по результатам рассуждений невозможно.

Метод рассуждений на основе сущностей (англ. instance-based reasoning). Данный метод в целом похож на предыдущий метод, однако в значительно большей мере может рассматриваться как синтаксический подход. Чтобы компенсировать отсутствие в рассуждениях профессиональных знаний, сущности упорядочиваются для вычисления их схожести на основе определения понятия. Представление сущностей простое — векторное. Это возможность автоматического обучения, без

участия эксперта. В этом методе нет обобщения нескольких прецедентов в ходе обучения. Оно выполняется классическими индуктивными методами машинного обучения.

Метод рассуждений на основе воспоминаний, записей в памяти (англ. *memory-based reasoning*). В этом подходе используется большая совокупность прецедентов, организованная в виде памяти. Рассуждения рассматриваются как процесс доступа и поиска в этой памяти. Организация памяти и доступ к ней — это центральный вопрос методов, объединенный в данный класс. Это и отличает данный подход от других. Методы доступа к памяти и запоминания используют чисто синтаксические критерии или могут применять профессиональные знания о предметной области для рассуждений. Они могут также работать с массивом параллельно функционирующих областей памяти.

Метод рассуждений на основе прецедентов (англ. *case-based reasoning*). Этот класс методов — родовое понятие. Однако типичные, характерные рассуждения на основе прецедентов имеют и специфические особенности. Во-первых, прецедент, если касаться его организации, обычно должен иметь совокупность различных степеней выразительности, прагматичности содержащейся в нем информации и несколько степеней сложности. Приписывание прецеденту совокупности векторов и одновременно отнесение прецедента к некоторому классу — не характерно для данного класса методов. Здесь важно другое свойство: они могут модифицировать, адаптировать решения, в зависимости от контекста решаемой задачи. Эти методы также могут использовать обобщенные знания — благодаря степеням выразительности, уровням явного представления. При этом можно варьировать процесс рассуждений. Ядро таких методов заимствовано из когнитивных психологических теорий.

Метод рассуждений на основе аналогий (англ. *analogy-based reasoning*). Этот термин иногда используется как синоним рассуждений на основе прецедентов. Однако он также применяется, чтобы выделить методы, которые решают новые задачи, основываясь на прецедентах из другой предметной области, в то время как типичные рассуждения на основе прецедентов концентрируются на индексации и стратегиях сопоставления для однодоменных прецедентов. Методы данной группы акцентируются на механизмах выявления и применения аналогий между несколькими предметными областями и в большей степени характерны для рассуждений ЛППР, чем экспертов. Главный вопрос таких методов — это способы повторного использования прошлого прецедента, что называется отображением задачи. Отображение задачи — это, по сути, отыскание способа передачи или сопоставления решения путем обнаружен-

ной аналогии (оно называется исходным или базовым) с настоящей задачей, называемой целью.

Жизненный цикл рассуждений на основе опыта. Для всех рассмотренных выше методов рассуждений на основе опыта можно построить логически увязанную и специфическую последовательность этапов компьютерного моделирования, обычно называемую жизненным циклом (рис. 3.29). Этот цикл включает обнаружение (поиск) подходящих случаев в памяти (англ. retrieve), адаптацию (приспособление) информации из найденного прецедента для решения текущей (возникшей) задачи (англ. reuse), пересмотр предложенного решения (англ. revise), сохранение успешного или неудачного решения для дальнейшего использования (англ. retain).

Таким образом, новая, возникающая перед экспертом задача решается поиском одного или более прецедентов из предыдущего опыта, адаптацией прецедента тем или иным способом, использованием решения, принятого в прошлом для выработки настоящего решения, и сохранением нового опыта посредством его включения в существующую базу прецедентов.

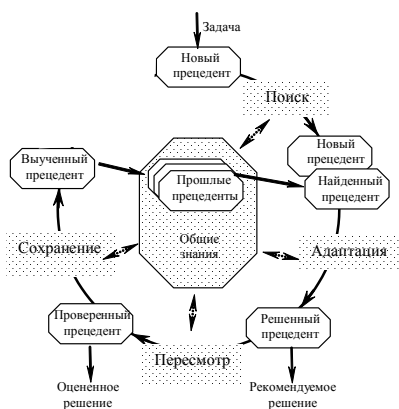


Рис. 3.29 — Жизненный цикл рассуждений на основе опыта

В итоге начальное описание (постановка) задачи определяет новый прецедент (кейс). Этот новый прецедент используется для поиска прошлых прецедентов в базе прецедентов. Найденный кейс комбинируется с новым случаем на этапе адаптации в «решенный кейс» (англ. solved case), т.е. в случай, для которого найдено и рекомендуется некоторое решение. На этапе пересмотра рекомендуемое решение проверяется на успех использованием в объекте управления, либо оно может быть оценено экспертом-учителем и исправлено, как ошибочное.

Во время этапа сохранения полезный опыт запоминается для будущего использования и база прецедентов обновляется либо изученным прецедентом (англ. learned case), т.е. путем его запоминания в базе прецедентов, либо путем изменения некоторых уже существующих в базе прецедентов.

Как показано на рис. 3.29, знания о предметной области играют определенную роль в жизненном цикле. Они поддерживают рассуждения. Эта поддержка может быть разной, от очень сильной до относительной.

тельно слабой в зависимости от применяемого метода рассуждений. Под знаниями понимаются предметно-ориентированные знания конкретной предметной области, в противоположность специфическим проблемно-ориентированным знаниям, включенным непосредственно в прецеденты. Например, в качестве предметно-ориентированных знаний могут выступать знания по биологии растений, а в качестве проблемно-ориентированных — знания об особенностях растениеводства на конкретных участках, конкретного сельскохозяйственного поля. Эти знания могут быть записаны, например, с помощью правил «Если...то...».

Для моделирования рассуждения на основе опыта требуется использование широкого спектра методов: представления знаний, поиска, адаптации, пересмотра и сохранения.

Представление прецедентов. Существуют различные методы представления прецедентов, в частности — модель динамической памяти [176 — 178] и модель «категория-экземпляр» [184].

Память в модели динамической памяти — иерархия, элементы которой — эпизоды или обобщенные эпизоды (рис. 3.30). Основная идея организации такой памяти — включение частных, конкретных прецедентов, обладающих общими признаками, т.е. похожих друг на друга, в некоторую обобщенную структуру, называемую эпизодом.

Обобщенный эпизод может включать три различных типа элементов: нормы, прецеденты и индексы. Нормы — это общие признаки для всех включенных, проиндексированных в эпизод прецедентов. Индексы — это признаки, отличающие один прецедент от другого. Индекс задает связь эпизода с другими более специфическими эпизодами или непосредственно с прецедентами. Индекс включает имя и значение.

Когда на входе системы появляется новое описание прецедента и когда один или более признаков из нового прецедента совпадут с одним или более признаками в обобщенном эпизоде, то в дальнейшем совпадение определяется по структуре индексов и их значений. Индексы, расположенные в иерархии ниже данного эпизода, пересекаются (т.е. выполняется операция пересечения множеств), чтобы найти прецедент с большим количеством совпадений признаков по отношению к исходной задаче.

В итоге будет найден прошлый прецедент, совпадающий по большинству признаков с новым.

Во время запоминания прецедента, в случае совпадения признаков прошлых прецедентов и нового, в памяти создается новый обобщенный эпизод. Для этого общая часть двух прецедентов записывается в раздел «нормы», а отличительные признаки с их значениями формируют индексную структуру на следующих по отношению к норме уровнях.

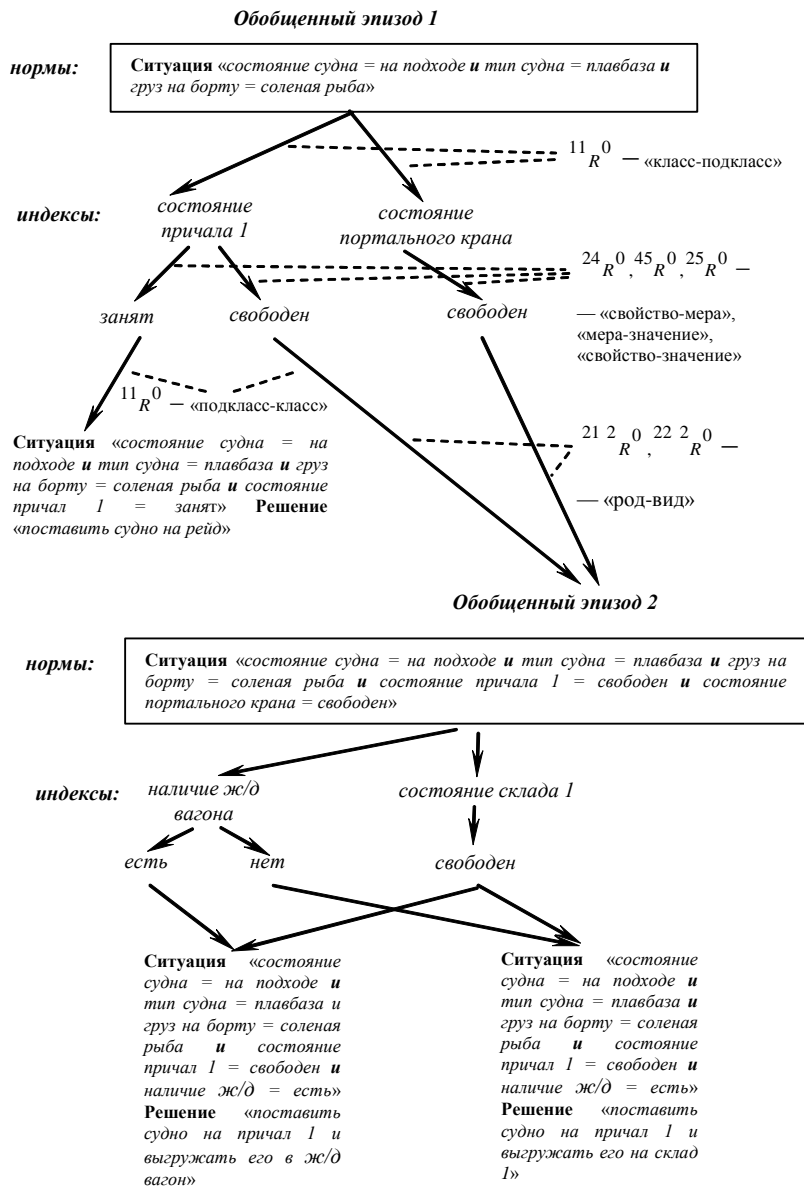


Рис. 3.30 — Структура организации информации в модели динамической памяти для задачи планирования причала для судна и схемы его обработки

Если в результате формирования индексной структуры некоторые прецеденты будут иметь одинаково индексированные «пути» к ним, то автоматически создается новый обобщающий эпизод. В этом смысле модель памяти и называется динамической.

Поскольку модель динамической памяти может приводить к взрывоопасному разрастанию структуры индексов, то при разработке СБР-систем вводят различные ограничения на размер словаря индексов.

Обнаружение похожих прецедентов. Получая на вход описание задачи, алгоритм обнаружения, используя индексы в памяти прецедентов, должен обнаружить наиболее похожие на текущую ситуацию прошлые прецеденты.

Рассуждения на основе прецедентов могут применяться для решения задач большой размерности только тогда, когда алгоритм обнаружения сможет работать эффективно на памяти с тысячами прецедентов. Подобно алгоритмам поиска записей в базах данных, который направляется специфическими значениями в записях, обнаружение прецедентов в базе прецедентов должно быть оснащено эвристиками, обрабатывающими частичные совпадения, пока не найдется прецедент, точно совпадающий с новым прецедентом.

Известны несколько методов обнаружения прецедентов: ближайшего соседа; индукция; индукция, управляемая знаниями; обнаружение по шаблону. Эти методы могут использоваться в отдельности или комбинироваться в гибридные стратегии. В частности, метод ближайшего соседа предполагает оценку схожести между прошлыми прецедентами и новым прецедентом, основываясь на подгонке взвешенной суммы признаков. Наибольшая проблема здесь — это определить веса признаков. Ограничение этого метода состоит в балансе между нахождением корректного решения и временем, затраченным на поиск. Время поиска линейно возрастает при увеличении числа прецедентов в памяти.

Типовой алгоритм базируется на вычислениях по формуле:

$$\left(\sum_{i=1}^n (w_i \times \text{sim}(x_i^I, x_i^R)) \right) / \sum_{i=1}^n w_i, \quad (3.31)$$

где w — весовое значение значимости (важности) признака; sim — функция схожести; x^I, x^R — значения i -го признака в текущем и прошлом прецеденте соответственно.

Такая задача оценки схожести, на первый взгляд, может показаться очень простой, пока не становится очевидной необходимость определения понятия «окрестность». Если текущая окрестность будет слишком мала, то в нее не попадет ни одного соседа. Если же взять такую окрестность излишне большой, то все оценки схожести могут оказать-

ся одинаковыми. На практике хорошие результаты для небольших размерностей пространств признаков дают значения n между 5 и 10.

Для выяснения соседних точек, ближайших к точке значения признака из текущего прецедента, применяется метрика расстояния $sim(\mathbf{x}_i^I, \mathbf{x}_i^R)$. В двухмерном пространстве используется евклидово расстояние, а для дискретных признаков — расстояние Хемминга, где $sim(\mathbf{x}_i^I, \mathbf{x}_i^R)$ — число признаков, отличающих точки $\mathbf{x}_i^I, \mathbf{x}_i^R$.

Обобщая сказанное о методах моделирования опыта решения задач экспертами, отметим, что помимо рассмотренных выше, еще один класс развивающихся в настоящее время представлений — представления специфического вида информации — опыта экспертов (рис. 3.31).

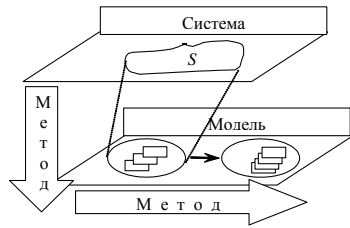


Рис. 3.31 — Методы рассуждений на основе опыта эксперта

Особенность таких методов — абстрагирование от многих свойств системы S и отображение ее базой единиц субъективного прошлого опыта, формулируемого в виде прецедентов «ситуация — решение», а также единиц знаний, уточняющих применение прецедентов для решения задач. Решение задачи рассматривается как процесс поиска подходящего прецедента в памяти, адаптации его к текущей ситуации, выработки и принятия решения на его основе и сохранения в памяти новых успешных или неудачных примеров решения задач.

На базе представлений опыта эксперта развиваются методы дискретной математики, теории управления, поддержки принятия решений, анализа данных и методы искусственного интеллекта.

Представления опыта экспертов перспективны в автоматизированном проектировании, программировании, истории, судебной практике, юриспруденции, медицине, диагностике технических установок, поиске неисправностей и обучении. Достоинства и недостатки методов рассуждений на основе опыта даны в табл. 3.3.

Концептуальная модель для класса методов рассуждений на основе опыта приведена ниже:

$$\begin{aligned} \dot{m}_p^a | Te &= {}^1x_2^0 \circ {}^{11}r^n \circ {}^1x_1^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{13}^0 \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{27}^0 \circ \\ &\circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^{22}x_{32}^0 \circ {}^1x_2^0 \circ {}^{13}r^n \circ {}^3x_{16}^0 \circ {}^1x_2^0 \circ {}^{13}r^n \circ {}^3x_{23}^0 \circ {}^1x_2^0 \circ {}^{18}r^n \circ {}^8\pi_p^h \circ {}^1x_2^0 \circ {}^{12}r^n \circ {}^2O_p^m, \quad (3.32) \\ {}^8\pi_p^h &= A \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2K_p^h \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2O_p^h \circ B, \end{aligned}$$

где ${}^8\pi_p^h$ — схема ролевых концептуальных моделей задач, решаемых методами рассуждений на основе опыта; ${}^2K_p^h$ — включает фазу — «анализ», «нормирование», «прогнозирование», «регулирование», «организация», переменные — «детерминированные», «четкие лингвистические», класс задач — (не определен); переменные-отношения — «сравнения», «включения», «определения», «причина-следствие»; ${}^2O_g^h$ — специфицируется в ходе разработки; A и B — фрагменты (3.32), уточняемые в ходе разработки; ${}^2O_g^m$ — спецификатор, включающий: знания о возможной погрешности решения в целом для класса методов — в пределах ошибки обучения и тестирования; знания о гибридных возможностях не установлены.

Модель вычислений с использованием методов рассуждений на основе опыта имеет следующий вид:

$$\dot{m}_p^a \Big|^{Ca} = \langle KB_1, KB_2, A(p_j, p_k), I^p \rangle, \quad (3.33)$$

где KB_1, KB_2 — базы единиц опыта (прецедентов) и единиц общих знаний о предметной области; $KB_1 = (\{p_1, \dots, p_{N_p}\}, \{i x^0 | i \in \{1, 2, 4, 5, 6\}\}; \{{}^{11}R^0, {}^{21}{}^2R^0, {}^{22}{}^2R^0, {}^{24}R^0, {}^{45}R^0, {}^{25}R^0\})$, где $p_j | j = 1, \dots, N_p$ — прецеденты, $i x^0$ — индексы; ${}^{11}R^0$ — отношения «класс-подкласс», «подкласс-класс»; ${}^{21}{}^2R^0, {}^{22}{}^2R^0$ — «род-вид»; ${}^{24}R^0, {}^{45}R^0, {}^{25}R^0$ — отношения «свойство-мера», «мера-значение», «свойство-значение»; $A(p_j, p_k)$ — алгоритм определения похожих прецедентов.

Интерпретатор I^p , используя $A(p_j, p_k)$ и KB_2 , обрабатывает информацию, хранящуюся в базе прецедентов KB_1 , и представляет собой совокупность процессов: $I^p = \langle I^{p1}, \dots, I^{p4} \rangle$, где I^{p1} — обнаружение; I^{p2} — адаптация; I^{p3} — пересмотр; I^{p4} — сохранение.

Процессы I^{p1}, \dots, I^{p4} , выполняются в следующей последовательности:

$$I^{p1} \rightarrow I^{p2} \rightarrow I^{p3} \rightarrow I^{p4} . \quad (3.34)$$

Таблица 3.3 — Преимущества и недостатки СВР-систем

Преимущества (${}^2O_3^{mp}$)	Недостатки (${}^2O_4^{mp}$)
1) Возможность напрямую использовать исторический опыт. Во многих областях такую информацию можно получить без извлечения знаний из эксперта [137] ЭКОПР, НАИНЖ. 2) Возможность сокращения рассуждений. Если известен аналогичный случай, новые задачи решаются гораздо быстрее, чем в ЭС или на моделях [137] ИЗИНВ. 3) Не требуется всестороннего анализа знаний о предметной области, когда инженер по знаниям должен описать взаимодействие правил. Вместо этого используется более простая аддитивная модель приобретения знаний [137] НАИНЖ. 4) Рассуждения на основе опыта позволяют избежать прошлых ошибок и использовать прошлые успехи [137] АДЕРЗ. 5) Стратегии индексирования повышают эффективность решения задач. Возможно выяснение отличительных особенностей решаемой задачи от прошлых случаев [137] ИЗИНВ, ЭКОПР	1) Описания прецедентов обычно не учитывают более глубокие знания о предметной области. Это снижает качество объяснения и во многих случаях приводит к ошибочному применению опыта, а значит, и низкому качеству советов [137] АДЕРЗ. 2) Большой размер базы прецедентов может привести к снижению производительности системы [137] ИЗИНВ. 3) Трудно определить хороший критерий для индексирования и сравнения прецедентов. Словари поиска и алгоритмы определения подобия необходимо тщательно отлаживать вручную. Это может нейтрализовать многие из преимуществ рассуждений на основе опыта [137] НАИНЖ, ЭКОПР

Обозначения: ПРОПС — предметная область, проблемная среда; НАИНЖ — наука, инженерия; ЭКОПР — экономика, практика; ИЗИНВ — измерения, информация, вычисления; АДЕРЗ — адекватность, релевантность, результаты

3.5. Моделирование невербальных рассуждений экспертов

В разд. 3.3.3 были затронуты вопросы моделирования интуитивно-образных рассуждений, отличающихся от логико-лингвистических представлений и вычислений (разд. 3.3.1, 3.3.2). Это различие принято объяснять наличием у человека двух подсистем обработки информации: образной и символично-логической. «Образная система — по-видимому, врожденная, а символично-логическая возникает в процессе обучения» [185]. Альберт Эйнштейн как-то заметил: «Слова, так, как они пишутся или произносятся, по-видимому, не играют существенной роли в механизме моего мышления. В качестве элементов мышления выступают более или менее отчетливые образы...» [22, с. 91].

Воспринимаемая человеком в ходе решения задачи сенсорная информация очень короткое время хранится в том виде, в котором она поступила. При передаче ее в основную память она проходит интерпретацию (категоризацию, кодирование). В психологии понятия «категоризация» и «рассуждения» имеют одинаковый смысл индуктивного обобщения, построения гипотезы. В ходе подобных рассуждений человек устанавливает смысл синтаксической информации соотношением ее с информацией, хранящейся в памяти.

Поскольку содержание памяти во многом определяется индивидуальным опытом, то «восприятие очень индивидуально и рассматривается как сложный процесс построения внутреннего представления (образа). Это объясняет такие эффекты в СППР, как разное «видение» одного и того же, противоречивые показания очевидцев, «испорченный телефон» и др. [185].

Внутренний образ — это не копия сенсорной картинки, это «картинка», прошедшая личностную категоризацию. Это внутреннее представление человека, по-видимому, принципиально защищенное от какой-либо формализации [138]. «Люди неспособны непосредственно обмениваться образами» [185, с. 120]. Для такого обмена они создают и используют внешние представления: слова, текст, речь, рисунки, жесты и др. При этом они отчуждают, объективизируют внутренний образ, кодируя его в терминах внешних представлений с потерей, искажением информации. Это объясняет такие эффекты в СППР, как «невозможность выразить словами все, что у тебя внутри», «риск быть понятым не так (или совсем не так), как ты хотел» и показывает невозможность полного перевода информации из образной подсистемы в символично-логическую [185].

Изучение образной (невербальной) подсистемы рассуждений специалиста актуально, далеко от завершения и научного осмысления. Известны некоторые отличительные свойства образов [185]: обрабатываются быстро; целостность; равновесие; распределенность в памяти; псевдооптические признаки — яркость, смутность («яркая мысль», «смутная догадка»), четкость, нечеткость, фокусировка и т.д.

Поскольку все эти свойства плохо и неадекватно моделируются логико-лингвистическими методами, О.П. Кузнецов [185] предлагает использовать для этого модели нейронного типа, в частности, псевдооптические нейронные сети. В таких сетях моделируются оптические, главным образом, голографические эффекты на информационном уровне: сигналы в сети распространяются по оптическим законам, а на входах нейронов происходит интерференция сигналов.

В основе вседооптических сетей, в отличие от модели Мак-Каллоха и Питса [1], лежит модель интерферирующего нейрона [185]. Он характеризуется порогом P_N и потенциалом $U_N(t)$, изменяющимся от 0 до P_N под действием входов. Нейроны соединяются волокнами с некоторой длиной и скоростью прохождения сигналов. Сигнал Sg_i — непериодическая функция с интенсивностью In_i , частотой ν , одинаковой для всех сигналов, и длительностью τ_i . Сигналы Sg_i и Sg_j поступают

на вход нейрона в моменты времени t_{1i} и t_{1j} с разностью фаз $\varphi_{ij} = 2\pi\nu(t_{1j} - t_{1i})$. Интенсивность на входе нейрона рассчитывается по закону интерференции: $In(t) = \sum_{i=1}^m In_i(t) + 2 \sum_{i,j \leq m} \sqrt{In_i(t) In_j(t)} \cos \varphi_{ij}$, где m — число входов нейрона. При постоянных значениях входных сигналов потенциал нейрона $U_N(t)$ растет пропорционально времени t и интенсивности $In(t)$, пока не достигнет порога P_N , после чего на выходе нейрона генерируется сигнал интенсивности In_N и длительности $\tau_N = P_N / (In_N \nu)$. Для интервала $[t, t']$, если на вход нейрона пришло m сигналов и потенциал U_N не превысил порога P_N , то: $U_N(t+t') = U_N(t) + \nu \left(\sum_{i \leq m} \sum_{j \leq m} \sqrt{In_i In_j} (\cos \varphi_{ij}) i_{ij} \right)$, где i_{ij} — время сосуществования сигналов Sg_i и Sg_j .

Псевдооптические сети состоят из четырех слоев нейронов: слоя A — источника (аналог источника опорной волны в голографии); слоя B — исходных образов, представляемых распределением потенциалов; слоя C — голограммы (образ B), как результат интерференции сигналов от слоев A и B ; слоя D — распределения потенциалов (восстановленный образ) в результате интерференции сигналов от нейронов слоя C , «освещенного» источником A .

Несколько иной подход к контрастированию образного мышления экспертов [22] направлен на практическое применение в деятельности СППР. Этот метод называется методом коллажей и фантазии.

Метод коллажей и фантазии. Это метод визуального восприятия, создания иллюзий на основе полученной информации, и он применяется, если проблемы и мнения экспертов трудно выразить словами. Это способ манипулирования в ходе рассуждений экспертов не словами, а образами. Графическое воплощение идеи дополнит вербальное выражение уже существующей идеи и может помочь экспертам в оформлении новой. Этот метод может использоваться индивидуально, каждым экспертом и коллективно в составе СППР.

Индивидуальное применение предполагает следующие шаги.

1. Постараться исследовать задачу всесторонне. Записать задачу на листе и поразмышлять над ней в течение нескольких минут: «Что в нее не вписывается?», «Каковы главные преграды?», «Что неизвестно?», «Что хочется понять?», «Больше всего беспокоит, что ...».
2. Расслабиться, чтобы интуитивное сознание более свободно использовало образы и символы.

3. Интуиция начнет создавать образы, сценарии, символы, отображающие сложившуюся ситуацию. Нет необходимости узнавать рисунок прежде, чем он будет нарисован.
4. Дать задаче форму, обозначив ее границы, любые по размерам очертаниям. Их можно рисовать тщательно или небрежно. Очерченные границы задачи дадут рисунку особый характер и глубину, воссоздадут цельное восприятие. Отделить «нужное» от «ненужного».
5. Рисовать по желанию интуиции, добиваясь, чтобы аналитическое мышление не участвовало в создании образов. Некоторые эксперты для этого рисуют левой рукой (для правой). Случайность и хаотичность придадут каракулям глубину, указывают на существование несфокусированной системы линий — «секретного послания подсознания».
6. Если результат не нравится, делается больше рисунков.
7. Исследование рисунка, сначала в целом, потом — по частям. Он — зримое отражение мыслей. Искать в каракулях символы неожиданных подсказок или новой информации.
8. Записать первое пришедшее на ум слово, обозначающее каждый образ, символ, закорючку, линию или структуры.
9. Связать слова воедино, написав небольшой текст. Дать волю ассоциациям, записать мысли, приходящие в голову. Сравнить текст и рисунок. Перерабатывая текст, добиться, чтобы рисунок и слова передавали одни и те же мысли на двух разных языках: образном и вербальном.
10. Соотнесение текста с поставленной задачей. Как изменилась точка зрения на задачу? Есть ли новые идеи? Посетило ли озарение? Каковы сюрпризы подсознания? Какие части рисунка озадачили? Что кажется неуместным?

Если в рассмотренном выше методе у СППР были трудности, то рекомендуется применять модифицированный метод коллажей и фантазий [22].

4. МЕТОДЫ МОДЕЛИРОВАНИЯ РАССУЖДЕНИЙ ЛИЦА, ПРИНИМАЮЩЕГО РЕШЕНИЯ, В СЛОЖНЫХ ЗАДАЧАХ СИСТЕМ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

4.1. Модели интеграции знаний экспертов

В силу свойств человеческого мозга все выводы, включая управленческие решения, совершаемые индивидом, в частности ЛПР, — типовые и базируются, как отмечалось выше, на осознаваемых или неосознаваемых моделях. Источники стереотипов — внутренние и (или) внешние носители стереотипов (знания экспертов), используемые для поддержки принятия решений. Управление процессом принятия решений в СППР — процесс управления стереотипами. Рождение «новых» типовых решений происходит благодаря, как правило, неосознанному процессу «смещения фреймов» или способности ЛПР находить прескриптивные модели, наилучшим образом соответствующие дескриптивным моделям экспертов. Это неразрывно связано с интеграцией знаний экспертов.

Способность интегрировать (комбинировать) мнения экспертов — характерная черта работы ЛПР в условиях коллективного принятия решений, такая же, как и способность к редукции сложной задачи и «раздаче» подзадач экспертам. Однако если редукция важна на первых шагах работы СППР, то интеграция — как субъективное явление — идет по ходу работы и актуальна в конце коллективного обсуждения.

Рассмотрим наиболее часто используемые подходы к моделированию эффекта интеграции знаний, применительно к СППР.

Многоагентная модель интеграции знаний экспертов. Интерпретируем основные элементы СППР (рис. 2.2) в терминах многоагентных систем и интеллектуальных организаций [13].

Агент — это лицо, уполномоченное кем или чем-нибудь для выполнения деловых служебных поручений, т.е. тот, кто действует в чьих-нибудь интересах, служит чьим-нибудь интересам. Из этого определения следует роль агентов в науке, как промежуточного звена между субъектом и объектом управления (рис. 4.1).

Если двигаться по стрелке на рис. 4.1 от субъекта управления к агенту, то возникает делегирование функций, задач и полномочий, а если перемещаться от объекта управления, возникают приобретение профессионального опыта и знаний (интеллектуализация) и активизация, т.е. способность взаимодействовать с внешней средой, общаться с другими агентами. Таким образом, агент, преследуя свои собственные цели, воспринимает информацию из внешней среды, обрабатывает ее, используя

свои ресурсы и знания, вступает в отношения с другими агентами и воздействует на среду в течение некоторого времени.

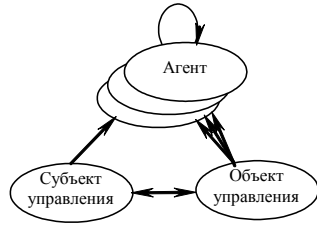


Рис. 4.1 – Агент в системах поддержки принятия решений

В многоагентных системах предполагается, что агент (эксперт) имеет ограниченное, частичное представление о задаче, т.е. «плохо видит» задачу в целом, но обязан «хорошо видеть» часть такой задачи — подзадачу. Решая ее, он тем самым высказывает свое профессиональное мнение. Поэтому для решения сложной задачи, как задачи-системы (см. разд. 1.3), требуется взаимодействие агентов-экспертов.

В многоагентных системах задачи распределены между агентами — членами группы или организации. Распределение задач предполагает,

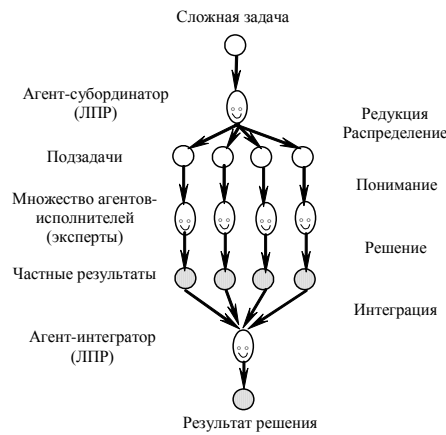


Рис. 4.2 — Распределенное решение задачи в многоагентной модели интеграции знаний по В.Б. Тарасову

что извне или от ЛПР назначаются роли каждому из агентов, определяется их ответственность, выдвигаются требования к их знаниям и опыту. Результаты решения подзадач экспертами оцениваются для принятия последующих решений о составе агентов. Все это похоже на распределенный искусственный интеллект, предлагающий следующую схему распределенного решения задач (рис. 4.2) [13].

ЛПР (агент-субкоординатор) редуцирует сложную задачу на отдельные части. Эти подзадачи распределяются им же между агентами-исполнителями. Каждый агент-исполнитель решает свою задачу, подчас также разделяя ее на части. Для получения общего результата агентом-интегратором проводится интеграция частных результатов. Чаще всего интегратор — это тот же субкоординатор, одно и то же ЛПР. Таким образом интеграция знаний экспертов в многоагентных системах обеспечивается функционированием специального агента-интегратора.

В отличие от теории систем, опирающейся на выделение элементов и рассмотрение их взаимосвязей, многоагентные системы используют синергетические идеи И. Пригожина и Г. Хакена [186, 187], а также кенетические идеи Ж. Фебре [13], где центральное место занимают исследование структур взаимодействия между агентами, выявление закономерностей самоорганизации и спонтанного образования упорядоченных пространственных, временных и функциональных структур, а также воздействие агентов на внешнюю среду. Образование и эволюция таких структур связываются с открытостью, нелинейностью, неравновесностью и неоднородностью систем.

В простейшем случае многоагентная система — это совокупность четырех объектов [13]:

$$MAS = (A, E, R, ORG), \quad (4.1)$$

где A — множество агентов; $E = \{e\}$ — среда, в которой находится данная многоагентная система; R — множество взаимодействий между агентами; ORG — множество базовых организационных структур, соответствующих конкретным функциям, ролям агентов и их установившимся отношениям.

В.Б. Тарасов в работе [13] посвящает целый раздел взаимодействию между агентами, что означает установление двухсторонних и многосторонних динамических отношений между ними. Взаимодействие он рассматривает не только как следствие одновременных действий агентов, но и как необходимое условие формирования виртуальных коллективов искусственных сообществ. Один из лозунгов синергетического искусственного интеллекта гласит: «нет знаний без взаимодействия». Под взаимодействием здесь понимается не только связь, взаимообусловленность между агентами, но и предпосылка для взаимных превращений. Например, агент-исполнитель может стать агентом-интегратором, а сотрудничество перейти в конфликт за ресурсы.

В многоагентных системах используются такие виды взаимодействия, как кооперация и конкуренция. Среди кооперативных отношений выделяются союзничество (аддитивный эффект объединения знаний агентов) и партнерство (мультипликативный эффект объединения знаний агентов). Конкуренция дает отрицательный эффект объединения агентов, а конфликт чреват приближением к гомеостатическим границам агентов.

Взаимодействия между агентами избирательны, динамичны, имеют некоторую силу и при формировании многоагентной системы

«общаются» только агенты, каким-либо образом соответствующие друг другу и поставленной задаче. Агенты вступают во взаимодействие из-за общих целей, нехватки индивидуальных ресурсов, знаний, опыта, неспособности решать задачу в одиночку, наличия обязательств.

В многоагентных системах возможны следующие стратегии взаимодействия агентов: кооперация (сотрудничество), компромисс, содействие, конкуренция (конфронтация), конформизм (подобие), приспособление, солидарность и уклонение. В частности, разработаны модели кооперации агентов: модели контрактных сетей [188], модели аукциона [189], модели теории полезности с минимальными уступками [190], модели социальных зависимостей [191]. Процессы взаимодействия между агентами описаны с помощью аппарата нечетких отношений [13].

В качестве примера рассмотрим четыре способа формирования различных архитектур многоагентных систем, как представления интегрированных знаний в процессе взаимодействия агентов: ассоциацию, комбинацию, интеграцию и слияние [13].

Ассоциация (рис. 4.3,а) — это архитектура (процесс ее формирования заключен в жирный прямоугольник), образуемая на основе симметричных положительных связей (R_+) между исходными автономными агентами a и b , когда $R_+(a,b) = R_+(b,a)$. Оба агента на паритетных началах участвуют в создании системы, вкладывая, например, одинаковое количество ресурсов. Автономность агентов означает слабую зависимость процесса создания ассоциации от внешней среды (входы и выходы обозначены пунктирными стрелками). Если автономные агенты однородны, то ассоциация называется коалицией.

Комбинация (рис. 4.3,б) — это архитектура, образуемая на основе несимметричных положительных связей между агентами a и b , когда $R_+(a,b) > R_+(b,a)$. Процесс формирования становится иерархическим, подчиненным, причем первая система — агент a имеет приоритет над второй системой — агентом b . В пределе первая система может блокировать вклад второй в создание системы-комбинации.

Интеграция (рис. 4.3, в) — это архитектура, образуемая из компонентов двух систем агентов a и b , наиболее приспособленных по отношению к внешней среде. Интегратор — внешние воздействия R_{es} .

Слияние (рис. 4.3, г) — это архитектура, когда исходные агенты a и b (например, «сильных» сторон a и b) теряют свою автономность, самостоятельность и из них формируется новая система с общим выходом R_{es} — требуемым поведением.

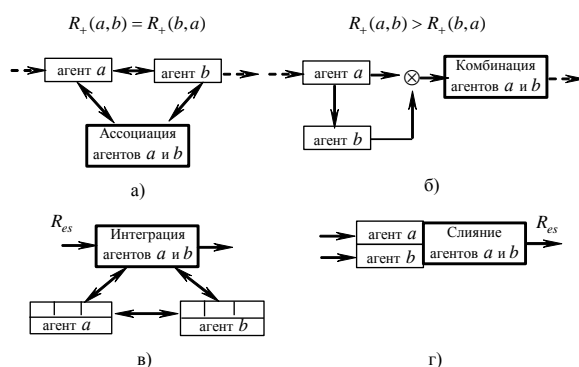


Рис. 4.3 — Способы формирования архитектур многоагентных систем:
 а) — ассоциация агентов, б) — комбинация агентов, в) — интеграция агентов,
 г) — слияние агентов

Модель интеграции знаний экспертов «доска объявлений». На протяжении последней четверти двадцатого века ученые постоянно пытались применить подход, используемый в колаборативном (направленном на обеспечение совместной деятельности) программном обеспечении, для решения сложных проблем. Системы с доской объявлений (англ. blackboard) — первая попытка организации взаимодействия между отдельными программными модулями в искусственном интеллекте, отображающая гибкий, основанный на принципах «мозгового штурма» процесс решения группой специалистов сложных, недоступных для решения одним экспертом задач.

В основе организации систем с доской объявлений лежит следующая идея: 1) группа экспертов находится возле классной доски (или большой доски объявлений) и пытается решить проблему; 2) каждый эксперт — специалист в определенной области, имеющей отношение к решению задачи; 3) формулировка проблемы и исходные данные записаны на доске; 4) эксперты пытливо вглядываются в то, что написано на доске, и каждый из них думает над тем, чем он может помочь в решении проблемы; 5) если кто-либо из экспертов чувствует, что ему есть что сказать по этому поводу, он выполняет соответствующие рассуждения, вычисления и записывает результат на той же доске; новый результат может позволить и другим экспертам внести определенный вклад в решение задачи; 6) процесс прекращается, когда проблема будет решена.

Приведенная выше метафора как нельзя лучше характеризует следующие восемь наиболее важных свойств систем с доской объявлений.

1. Независимость опыта. Эксперты никогда не учились работать вместе, в составе конкретной группы. Данные специалисты могут обладать разным опытом в совершенно разных областях. Кроме того, каждый специалист — эксперт (источник знаний) в каком-нибудь одном аспекте проблемы и может продвигаться к решению независимо от других специалистов.

Ни один источник знаний не требует вклада в решение проблемы от другого источника. Если он находит необходимую ему информацию на доске объявлений, то сразу приступает к решению задачи.

Могут быть добавлены другие источники знаний, что никак не отразится на уже задействованных в решении задачи, а неэффективные источники знаний могут быть безболезненно удалены из системы.

2. Разнообразие методов решения задач. Существует огромное разнообразие подходов специалистов к решению одной и той же задачи. Однако это разнообразие, в большинстве случаев, не мешает специалистам в рамках одной группы прийти к единому решению.

В системах с доской объявления процесс рассмотрения задачи тем или иным источником знаний закрыт от прямого доступа других источников знаний. Система рассматривает источник знаний в виде модели «черного ящика», где внутренняя работа и вычисления недоступны для просмотра извне.

3. Гибкое представление информации в рамках доски объявлений. При записи результатов вычислений эксперты могут использовать различные способы обозначения: формулы, диаграммы, предложения и т.д. Гибкость представления также важна и в системах с доской объявлений. Система не накладывает ограничений на вид информации, помещаемой на доску.

4. Общий язык интерфейса. Вся информация, выдаваемая одними источниками знаний, должна быть доступна для понимания другими. На практике существует проблема понимания представленных данных всеми источниками знаний.

5. Учет полученных результатов. По мере решения задачи количество записей на доске многократно увеличивается. Для специалиста не должна возникать проблема поиска результата, записанного другим специалистом. Один из способов рационального хранения информации — разграничение доски на области, каждая из которых предназначена для определенного класса информации: расположенной в алфавитном, нумерологическом порядке либо по уровню релевантности.

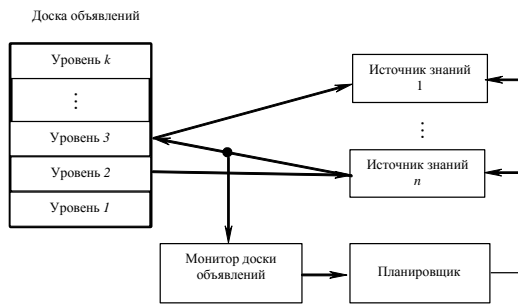
6. Действие на основе событий. Эксперты не взаимодействуют непосредственно друг с другом. Каждый из них смотрит на доску, ожидая возможности продолжить решение задачи. Возможность появляется тогда, когда информация появляется либо удаляется с доски.

Источники знаний в системах с доской объявлений приводятся в действие внутренними или внешними событиями. Помимо того, что источник сканирует доску в поисках нужных данных, каждый источник знаний оповещает доску о типе событий, в которых он заинтересован. При возникновении такого события система оповещает источник.

7. Необходимость контроля. Для организации решения задачи необходимо наличие протокола очередности «выступлений», регламента, вступающего в силу в ситуации, когда несколько экспертов берутся за мел и направляются к доске. Это обеспечивается контроллером, не зависимым от источника. Данный механизм отвечает за порядок решения проблемы и рассматривается как специалист в области управления решением задачи. Управление осуществляется посредством оценки ценности и качества вклада активируемого источника знаний в решение проблемы. После завершения работы какого-либо источника для активации выбирается другой, наиболее подходящий.

8. Последовательный вывод. Решение задачи зачастую возникает последовательным добавлением экспертами результатов применения своих знаний на доску. Ни один специалист не может единолично найти решение задачи. Специалисты «очищают» и уточняют (углубляют) рассуждения других экспертов, постепенно приходя к верному решению.

Если рассматривать систему с доской объявлений с точки зрения организации вычислений, то можно выделить в ней следующие компоненты (рис. 4.4): 1) источники знаний; 2) доска объявлений; 3) монитор.



Обозначения: стрелки, направленные от уровней доски объявлений к источникам знаний, указывают, данные какого уровня изменяют выполнение условий, специфизированных в источнике знаний; стрелки в обратном направлении указывают, на какой уровень помещает данные тот или иной источник знаний; отвлечение от стрелки «действия» источника знаний к монитору означает, что изменение данных, выполненное одним источником знаний, фиксируется в мониторе и затем используется планировщиком для активации другого источника знаний.

Рис. 4.4 — Архитектура доски объявлений

Знания о предметной области разделены между независимыми источниками знаний (англ. knowledge sources), которые работают под управлением планировщика (англ. scheduler).

Решение формируется в некоторой глобально доступной структуре, называемой доской объявлений. Таким образом, в этой системе все знания «как поступить»

будут представлены не в виде единственного набора правил, а в виде

набора программ. Каждый из компонентов этого набора может располагать собственным набором правил либо смесью правил и процедур.

Функции доски объявлений во многом сходны с функциями рабочей памяти в продукционных системах. Как правило, доска объявлений разделяется на уровни описания, причем каждый уровень соответствует определенной степени детализации. Данные в пределах отдельных уровней доски объявлений представляют иерархии объектов или графы, т.е. структуры более сложные, чем векторы, которые использовались в рабочей памяти продукционных систем. Досок объявлений может быть несколько.

Монитор используется для контроля всего процесса решения задачи. Он сортирует задачи из плана решения, списка и эвристическим путем выбирает следующую задачу для выполнения. Монитор осуществляет проверку доски объявлений и области сообщений после работы каждого модуля знаний, анализируя слоты объявлений, связанные с гипотезами, разделяющими текущее состояние процесса. На основе механизмов рассуждений в прямом (от данных) и в обратном (от цели) направлениях монитор определяет, какие модули знаний соответствуют текущему состоянию процесса решения задачи. По сути, механизм вывода состоит из agenda и монитора.

Источники знаний формируют объекты на доске объявлений посредством планировщика. Обычно записи активизации источников знаний (англ. knowledge source activation records) помещаются в план решения, откуда их извлекает планировщик. Источники знаний общаются между собой только через доску объявлений и не могут непосредственно передавать данные друг другу или запускать выполнение каких-либо процедур. Здесь есть определенная аналогия с организацией работы экспертных систем, в которых правила также не могут непосредственно активизировать друг друга: все должно проходить через рабочую память.

Для генерации, комбинирования и развития гипотез интерпретации в системе с архитектурой доска объявлений используется несколько источников знаний. Созданные гипотезы (интерпретации) разного уровня абстракции сохраняются на доске объявлений.

Источники знаний связываются с уровнями доски объявлений следующим образом. Условия, специфицированные в источнике знаний, будут удовлетворяться в результате обновления данных на определенном уровне доски объявлений. Источник знаний также может записывать данные в определенный уровень, причем не обязательно в тот же, который влияет на выполнение условий.

Главное отличие архитектуры с доской объявлений заключается в том, что такая система не диктует разработчику определенный режим управления знаниями в системе, например нисходящую или восходя-

щую стратегию построения рассуждений. Какой источник знаний будет активизирован, определяется монитором и планировщиком, а это решение можно сделать или независимым от предметной области, т.е. от соответствующих источников знаний, или зависимым от них.

Архитектура «доска объявлений» используется как самостоятельно, так и, например, в многоагентных системах для реализации механизма коммуникации агентов [13].

Гибридная модель интеграции знаний экспертов. Начало гибридов (греч. *hybris, hybridos* — организм, возникающий в результате гибридизации [141]), можно искать в древних легендах о происхождении рода человеческого или в более поздних работах ученых-генетиков, однако и в том и другом случаях имеют дело с производными от родителей сущностями, играющими в современной науке и эволюции цивилизации огромное и все возрастающее значение. Это значение определяется, прежде всего, привлекательностью уже доказанной учеными-генетиками для живой природы возможностью целенаправленного конструирования объектов на уровне генотипа [192] (микромира), что во многом определяет свойства индивидуумов на уровне фенотипа (макромира). Такое конструирование — «тонкая», долгая и трудоемкая работа — называется гибридизацией, т.е. соединением в одном организме разнородных наследственностей.

Естественная гибридизация на Земле наблюдалась давно, а история искусственной насчитывает около 300 лет. Возможность искусственного получения гибридов впервые предложил немецкий ученый Р. Каме-рариус (1694), а искусственную гибридизацию растений впервые осуществил англичанин Т. Фэргайлд (1694). Чешский естествоиспытатель Г. Мендель (1865) сформулировал законы доминирования, расщепления и независимого комбинирования в генетике как науке и обосновал необходимость гибридологического анализа для изучения механизма передачи наследственных задатков, генов.

Со середины 60-х годов двадцатого столетия «генетическая парадигма» начала активно использоваться и другими науками. В геологии изучались гибридные горные породы, обладающие неоднородными текстурами и структурами. В лингвистике исследовались гибридные языки с генетической неоднородностью лексики, морфологических и синтаксических моделей. В технических науках, в частности, вычислительной технике, разработаны гибридные, аналогово-цифровые вычислительные системы и гибридные микросхемы.

В любом случае переход к генетической парадигме означал отказ от взгляда на объект исследования как простую, однородную сущность и принятие мировоззрения сложного, составного, неоднородного объекта. Это всегда позволяло подниматься на качественно новый уровень

анализа известных и синтеза новых объектов с полезными для человека свойствами.

В науке управления генетическая, гибридная парадигма зародилась также в середине 60-х годов и проявляется в разработке интегрированных моделей и методов. Возможные формы такой интеграции — гибридные системы (см. разд. 4.2.1) и гибридные интеллектуальные системы, предмет настоящей книги.

Семантика этих уже устоявшихся в мировой практике терминов раскрывается через слова «гибрид» и «система». Само по себе слово «гибрид» уже содержит признаки системы — состав, структуру, порядок, эмерджентность и др. Поэтому сочетание слов «гибрид» и «система» в одном термине может только подчеркивать, что родовый смысл этого понятия выражается системой, а видовое отличие от других систем раскрыто в понятии гибрида. Это видовое отличие, прежде всего, состоит в методе получения — через гибридизацию и подчеркивает неоднородность, многоаспектность отображения в одной сущности-модели различных признаков одного или нескольких оригиналов. Например, таких признаков, как «дискретность и непрерывность», «абстрактность и конкретность», «интеллект и рутина», «эвристика и закон», «алгоритм и формула», «слабость и сила» и др.

Если, например, в качестве признаков-компонентов использовать различные виды моделей, методов, знаний, естественные интеллекты, то можно ввести понятия гибридных моделей, гибридных методов, гибридных знаний.

Обобщающий эти понятия термин «гибридный интеллект» ввел в 1975 г. советский психолог В.Ф. Венда [9], чтобы выдвинуть контрпонятие искусственному интеллекту и подчеркнуть непреходящее эволюционно-историческое значение взаимодействия естественных интеллектов в природе, обществе и технике. Было также подчеркнуто, что искусственный и гибридный интеллекты диалектически взаимосвязаны и переходят один в другой.

В.Ф. Венда сформулировал три закона гибридного интеллекта: взаимной адаптации, дискретных рядов структур и закон трансформаций.

Закон взаимной адаптации утверждает, что необходимое и достаточное условие возникновения и развития гибрида — наличие процессов внутренней, между компонентами гибрида, и внешней, гибрида с внешней средой, взаимной адаптации. Структура гибрида — отображение определенной закономерности взаимной адаптации его внутренних компонентов. Эта структура устойчива, если взаимная адаптация гибрида и внешней среды характеризуется состоянием, при котором эффективность решения задач гибридом лежит в заданном интервале.

Закон дискретных рядов структур утверждает, что любой гибрид может быть реализован посредством одной структуры из дискретного ряда его возможных структур. При этом существует некоторый метод получения одной структуры, входящей в дискретный ряд, из другой структуры этого ряда. Кроме этого, в таком ряду должны существовать целевые структуры, что позволит поставить задачу качества гибрида, т.е. сделать гибридизацию целенаправленной.

Закон трансформации утверждает, что трансформация одной структуры гибрида в другую может происходить только через общие для обеих структур знания. Этот закон описывает образование новых знаний, их интерференцию между собой, построение умозаключений, а также возникновение и роль ассоциаций в трансформации и связывании образов и мыслей. Любые новые знания не могут быть сформулированы на пустом месте, а могут быть получены только путем перехода от одного знания к новому знанию, взаимосвязанному, ассоциированному с предыдущим.

В соответствии с законом трансформации новая структура не может быть порождена как таковая и возникает только на базе предыдущей структуры. При этом сохраняется достигнутая при старой структуре взаимная адаптация части компонентов, соответствующей новой структуре. Закон трансформации показывает роль фундаментальных, формализованных знаний, которые, постепенно развиваясь, играют роль того общего, что связывает вновь появляющееся эвристическое знание.

Эти три закона можно положить в основу гибридизации в теории систем как процесса создания гибридов-методов, гибридов-моделей, гибридов-алгоритмов и гибридов-программ.

В соответствии с вышесказанным, понятие гибрида в информатике проиллюстрировано на рис. 4.5. Общее, что объединяет обе части рис. 4.5 — это представление сущности на различных уровнях детализации.

Для рис. 4.5,а такая сущность — сложная, неоднородная задача, для решения которой организована СППР. Понятие гибридного интеллекта возникает в этом случае тогда, когда ЛПР редуцирует единое целое, т.е. исходную задачу на составные части, и поручает решение отдельных частей целого, т.е. подзадач, экспертам. Таким образом, в системе возникает диалектическое противоречие «целое-часть», когда две сущности: общая задача-система и ее декомпозиция (результат редукции) начинают сосуществовать вместе. Такое противоречие, скорее всего, и является источником процессов самоорганизации в СППР. Выскажем гипотезу о том, что эти процессы идут следующим образом. После получения некоторой части задачи эксперт должен, прежде всего, понять ее. Один из значимых этапов понимания — осознание того, какой метод рассуждений лучше всего подходит для решения данной задачи из арсе-

нала методов, которые он обычно применяет в своей профессиональной деятельности. Чтобы сделать выбор, эксперт должен знать плюсы и минусы различных методов рассуждений. Выбрав метод (путем рассуждений или подсознательно), он строит модель для решения подзадачи, используя положительные стороны принятого метода рассуждений. Это не прямой путь, здесь возможны итерации проб и ошибок, выбор других методов и повторные рассуждения.

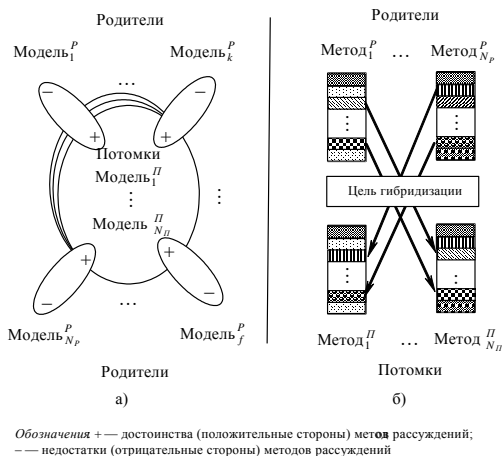


Рис. 4.5 — Два подхода к понятию гибрида в информатике

модель $1^P, \dots, \text{модель } k^P, \dots, \text{модель } f^P, \dots, \text{модель } N_p^P$. Многообразие используемых в СППР структур совместной работы, т.е. порядка, в котором применяются модели-родители для организации совместных рассуждений, и определяет, с нашей точки зрения, многообразие моделей-потомков: модель $1^П, \dots, \text{модель } m^П$. Эти модели, в данном случае, соотносятся с целым, т.е. общей задачей. Модели-потомки — это гибриды, полученные путем комбинирования по функциональному признаку и наследующие положительные стороны методов рассуждений, примененных отдельными экспертами для решения составных частей исходной задачи.

На рис. 4.5,б отобразена несколько другая картина. Она возможна не только в СППР, но и в других ситуациях принятия решений. Общий признак таких ситуаций — ощущение эксперта, что для решения стоящей перед ним задачи ни один из известных ему и успешно применявшихся ранее методов не подходит. Это ощущение возникает тогда, ко-

В итоге, по числу экспертов, решающих отдельные части общей задачи, и по характеру их совместной работы (последовательная работа над частями задачи, когда эксперты решают подзадачи в определенном порядке, параллельная, одновременная работа, формирование подгрупп (коалиций) и др.) создаются родительские модели (рис. 4.5): мо-

гда его не устраивают какие-либо отдельные аспекты методов. Например, один из них приводит к моделям с недопустимой погрешностью, но с хорошими вычислительными возможностями. Применение другого метода может давать прямо противоположную картину. Таких ситуаций много, при этом вольно или невольно эксперт начинает рассматривать инструмент решения задачи на разных уровнях детализации.

На макроуровне он оценивает возможности метода, как единого целого, применительно к решению некоторой задачи, а на микроуровне он представляет метод как сущность, состоящую из отдельных частей. Макро- и микроуровневые аспекты метода взаимообусловлены, здесь существуют причинно-следственные связи. Очень важно знать, какие модификации и каких частей метода на микроуровне могут привести к желаемым свойствам метода на макроуровне. И, наоборот, комбинировать, модифицировать отдельные части метода, чтобы получить полезные для решения задачи свойства.

В работах [1, 193] была предложена и применена для гибридизации классификация методов, в основе которой положены причинно-следственные связи макроуровневых свойств методов моделирования и особенностей их микроуровневого построения. Получить эту классификацию позволило рассмотрение таких составных частей любого метода, как «модель», «язык описания», «процедура получения решения». Модель — это те представления, например, аналитические, статистические (см. разд. 3), в пределах которых метод «работает». Это наименее подверженная изменениям часть метода. Смена представлений — это качественный «скачок» свойств метода из одного класса в другой. Язык описания — это альтернативное средство для записи модели. Это форма ее существования. Так, например, для записи аналитических моделей могут использоваться язык дифференциальных уравнений, алгебра, матрицы и др. Процедура — это упорядоченная совокупность действий (вычислений) для поиска решений на модели (3.3), (3.12), (3.15), (3.23), (3.25), (3.29) и (3.33). В разнообразии методов выполнения этих действий чаще всего и скрыто разнообразие их свойств на макроуровне.

Такой подход к представлению метода позволил ввести модель «мир методов моделирования» (рис. 7.3) и как один из центральных процессов в этом мире рассмотреть эволюцию методов моделирования [1]. Методы из некоторого класса, определяемого в большинстве случаев моделью и языком описания, формируют и занимают в этом мире некоторую нишу. В пределах этой ниши существует многообразие, определяемое в основном многообразием процедур поиска решений. Смена модели ведет к изменению ниши, а появление новой модели (представления) — к появлению новой ниши. Значительно чаще происходит мед-

ленное накопление количественных изменений в процедурной части методов и их «дрейф» в пределах одной и той же ниши.

Теперь можно вернуться к рис. 4.5,б. На нем показана некоторая совокупность методов-родителей из одной или нескольких различных ниш: метод P_1 , ..., метод P_{N_p} , которые известны эксперту и свойства на макроуровне ни одного из которых его не устраивает. Методы изображены на микроуровне, что показано многообразием составляющих их частей с различной штриховкой. Имея цель достигнуть желаемых макроуровневых свойств (цель гибридизации), можно скомбинировать эти части и получить новые методы-потомки: метод II_1 , ..., метод $II_{N_{II}}$. Можно сочетать признаки и случайным образом, т.е. сделать гибридизацию не целенаправленной. Однако такой путь не следует считать эффективным, хотя он часто и используется для генерации новых идей [22].

Методы-потомки — это гибриды, полученные путем целенаправленного комбинирования по инструментальному признаку и наследующие положительные стороны методов рассуждений, используемых отдельными экспертами. Это часто применяемый в информатике способ комбинирования, достаточно обратиться к алгоритму Сугено (см. разд. 3.3.2), в котором «работают» правила нечеткой импликации с лингвистическими переменными в левой части и аналитическими выражениями в виде полиномов в правой части. Такой же комбинацией будет экспертная система, в которой сопоставление с образцом (3.17) выполняется искусственной нейронной сетью (3.25).

Интегрированная модель знаний экспертов. Все возрастающая сложность архитектур организационно-технических систем приводит к объединению в едином программном продукте таких разнородных компонентов, как экспертные системы, базы данных, пакеты прикладных расчетных, моделирующих программ, машинной графики, обучающие системы, системы имитационного моделирования т.д. [64].

Сочетание методов искусственного интеллекта, математической статистики, имитационного моделирования, технологий программирования и других привело к интеграции моделей, методов и технологий традиционного программирования с экспертными системами. Такие сочетания стали называть системами, основанными на знаниях [64].

Разрабатывать архитектуры таких программных систем «слепым» переносом методологии экспертных систем и их инструментариев, например оболочек, малоэффективно, трудоемко и дорого.

В этой связи Г.В. Рыбиной создан теоретический, методологический базис, методы и программные инструментарии, в частности, комплекс

АТ-ТЕХНОЛОГИЯ, для поддержки жизненного цикла автоматизированного построения интегрированных экспертных систем.

В основе этой методологии — многоуровневая модель процесса интеграции в ИЭС, моделирование конкретных типов задач, релевантных технологии традиционных экспертных систем, а также методы и способы построения моделей и программных компонентов на каждом уровне интеграции. Это позволило систематизированно и наглядно «от архитектуры» проанализировать основные типы интегрированных интеллектуальных систем.

С точки зрения верхнего уровня интеграции, модель архитектуры интегрированной экспертной системы имеет следующий вид:

$$M_{ИЭС} = \langle F_{ЭС}, f_N, Q_{ЭСН} \rangle, \quad (4.2)$$

где $F_{ЭС}$ — множество функций F_i простой экспертной системы; $F_{ЭС} = \{ F_i \} | i = 1, \dots, k$; f_N — множество функций, не свойственных экспертным системам; $f_N = \{ f_j \} | j = 1, \dots, m$; $Q_{ЭСН}$ — отношения, описывающие взаимосвязи F_i и f_j между собой (в качестве f_j могут выступать любые «расширители» простой экспертной системы, выполняющие функции: обучения, создания и интерпретации гипертекстовых структур, интерфейса с расчетными, графическими и др. ППП).

На среднем уровне интеграции имеем:

$$M_{ИЭС} = \langle M_{РИЛ}, M_{ВЗ}, O_{МЕХ} \rangle, \quad (4.3)$$

где $M_{РИЛ}$ — расширенная информационно-логическая модель интегрированной экспертной системы; $M_{ВЗ}$ — модель взаимодействия элементов (модулей) разных типов; $O_{МЕХ}$ — отношение, описывающее механизмы отображения $M_{РИЛ}$ и $M_{ВЗ}$.

На уровне программной интеграции:

$$M_{ИНСТР} = \langle M_{БАЗ}, M_{ДОП}, M_{ТИП}, O_{СОВ} \rangle, \quad (4.4)$$

где $M_{БАЗ}$ — описание базовых программных средств; $M_{ДОП}$ — описание дополнительных средств; $M_{ТИП}$ — типы представления данных; $O_{СОВ}$ — отношение совместимости программных средств и типов данных.

Таким образом, интегрированная модель предполагает ведущую роль узкопрофессиональных знаний экспертов, которые для решения

сложных задач расширяются за счет теоретических знаний и вычислительных моделей, баз данных и скрытых в них знаний о предметной области, а также информационно-поисковых систем, систем автоматизированного обучения, знаний о поведении систем в имитационных моделях и др.

4.2. Методы интеграции знаний в многоагентных системах

В многоагентных системах предполагается, что агент может иметь только частичное представление о задаче и способен решить ее некоторую часть — подзадачу. Поэтому для решения сложной проблемы требуются организация и координация со стороны агента-ЛПР взаимодействия агентов-экспертов.

Первыми советскими системами, реализующими концепцию автономных агентов, стали разработанные школой Н.М. Амосова система ТАИР (транспортный автономный исследовательский робот) и программа «Животное», созданная под руководством М.М. Бонгарда. Первыми зарубежными практическими разработками в области многоагентных систем считаются результаты В. Лессера, К. Хьюитта и Д. Лената [194 —196]. С результатами работ в этой области можно познакомиться, например, в трудах В.Б. Тарасова, В.Ф. Хорошевского и В.И. Городецкого [13, 189, 197].

Классификация методов многоагентных систем В.Б. Тарасова приведена на рис. 4.6.



Рис. 4.6 — Классификация многоагентных систем В.Б. Тарасова

В многоагентных системах, построенных на идеях распределенного искусственного интеллекта, моделируются взаимодействие и кооперация небольшого числа интеллектуальных агентов, в частности, классических интеллектуальных систем с базами знаний и решателями. Центральная проблема здесь — разработка интеллектуальных групп и организаций для решения задач методами рассуждений на логико-лингвистических (символьных) представлениях (см. разд. 3.3.1). Методы интеграции при этом отличаются способами: распределения задач между агентами; распределения властных полномочий;

коммуникации агентов. Здесь предполагается наличие единого органа управления, например, ЛПР, принятия решений в конфликтных ситуациях. При этом для решения сложной задачи формируется группа агентов, строится общая концептуальная модель и вводятся глобальные критерии достижения цели.

В многоагентных системах, построенных на идеях децентрализованного искусственного интеллекта, моделируется не распределенное решение общей задачи, а деятельность автономного агента в динамическом мире и координация этой деятельности. При этом, наряду с распределенными знаниями и ресурсами, описываются локальные задачи агентов, решаемые на локальных концептуальных моделях и локальных критериях.

В многоагентных системах, построенных на идеях искусственной жизни (англ. artificial life), моделируется интеллектуальное поведение в контексте выживания, адаптации и самоорганизации в динамичной, враждебной среде [13]. Здесь поведение многоагентной системы рассматривается как результат локальных взаимодействий большого числа простых и не обязательно интеллектуальных агентов (см. табл. 1.9). Основное допущение здесь состоит в том, что сущность жизни определяется не столько свойствами этого материального субстрата, а организацией компонентов знаний и процессов в целостную систему. Поэтому считаются уместными термины «коллективный интеллект» (англ. collective intelligence), «интеллект роя» (англ. swarm intelligence).

Методы искусственной жизни основываются на следующих положениях [13]: 1) многоагентная система — популяция простых и не зависимых друг от друга агентов; 2) агент самостоятельно определяет свои реакции на события в локальной среде и по взаимодействию с другими агентами; 3) связи агентов — горизонтальные, координация их деятельности отсутствует; 4) нет точных правил, определяющих глобальное поведение агентов; 5) поведение, свойства и структура на коллективном уровне порождаются только локальными взаимодействиями агентов. Методы искусственной жизни используются для моделирования коллективного поведения в экобиосистемах. Такое поведение, в частности, демонстрируют колонии муравьев, пчелиные ульи, стаи перелетных птиц, рыб и др.

Из трех рассмотренных подходов к созданию многоагентных систем искусственная жизнь, в широком смысле этого слова, являясь научно-практическим, междисциплинарным комплексом [13], интегрирует знания математики, физики, химии, биологии, синергетики, кибернетики, робототехники и искусственного интеллекта.

Истоки этого направления восходят к работам Дж. фон Неймана по теории самовоспроизводящихся автоматов, А.Н. Колмогорова по теории

сложности и других ученых, а сам термин «искусственная жизнь» связан с публикациями Дж. Конвея конца 1960-х годов по использованию клеточных автоматов в модели «Игра жизни».

В искусственной жизни получены интересные результаты: 1) модель эволюции и самоорганизации популяций искусственных агентов, «живущих» в сети Интернет, В.Г. Редько и Б.Н. Гертгейзеля; 2) модель динамики жизнеспособных структур в клеточных автоматах К. Лэнгтона; 3) модель эволюции простых агентов ЕСНО Дж. Холланда, в которой агенты взаимодействуют между собой путем скрещивания, борьбы и переговоров; 4) модель искусственного муравейника MANTA А. Дрогуля; 5) модель эволюции самовоспроизводящихся программ Tierra Т. Рэя; и др. Кратко модели 1, 4, 5 рассмотрены в [13].

4.3. Методы интеграции знаний в гибридных системах

Один из ранних результатов (1966 г.) по искусственной гибридизации в детерминированных динамических системах с континуально-дискретным поведением был получен W. Witsenhausen [198], предложившим следующую H_W -модель гибридной системы:

$$H_W = \langle M, \Omega, f, d, \omega, J \rangle, \quad (4.5)$$

где M, Ω – конечные множества целых чисел (дискретные пространства состояний и выходов соответственно); $f: M \times \mathcal{R}^n \rightarrow \mathcal{R}^n$ — непрерывная функция перехода от двух аргументов; \mathcal{R} — множество действительных чисел; \mathcal{R}^n — пространство непрерывных состояний размерности n ; $d: M \times J \rightarrow M$ — дискретная функция перехода, вычисляющая очередное значение дискретной компоненты состояния, когда значение непрерывной компоненты состояния достигло одного из переходных множеств; $\omega: M \rightarrow \Omega$ — функция выхода и $J \in P(\mathcal{R}^n)$ — множество переходных множеств.

Гибридное состояние определяется, как вектор $(m, x) \in M \times \mathcal{R}^n$. Переход из дискретного состояния m_i в дискретное состояние m_j срабатывает, когда непрерывное состояние x достигает заданного множества J_{ij} в \mathcal{R}^n . Гибридная траектория (рис. 4.7) определяется так. Предположим, что начальное дискретное состояние гибридной системы есть m_0 и ему соответствует переходное множество $J_{m_0 m_1}$ или просто J_{01} . До тех пор, пока $x \notin J_{m_0 m_1}$, система функционирует с вектором $f(m_0, \cdot)$ в

соответствии с $\dot{x}(t) = f(m_0, x(t))$. Если, например, в момент времени t_1 , $x(t_1) \in J_{m_0 m_1}$ и дискретное состояние сменится на m_1 , то функционирование $\dot{x}(t) = f(m_1, x(t))$ продолжится с начальным условием $x(t_1) = x_1$. В момент времени t_1 также выдается выходной сигнал $O_1 = \omega(m_1)$.

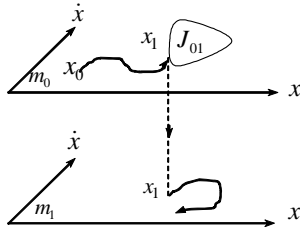


Рис. 4.7 — Функционирование гибридной системы

Очевидное достоинство H_W - модели — представление и манипулирование гибридным состоянием для имитации неоднородности в поведении системы с комбинированием плавных изменений значений x в соответствии с аналитическими знаниями (разд. 3.2), со скачками значений m в соответствии с теоретико-множественными представлениями знаний функцией d . Тем не менее, H_W -модель не учитывает

такие дискретные феномены, как автономные или управляемые скачки в непрерывном пространстве состояний.

Развитием базовых моделей в сторону многокомпонентности и связности следует считать модель [199], гибридную систему коммутируемых и координируемых компонент — гибридных автоматов [200], систему [201], использующую расширенную H_W - модель для математического моделирования управляющей сети, и гибридную систему — спецификацию систем для элементарных и связанных моделей [202].

Гибридную систему из координируемых компонентов предложили А. Puri и Р. Varaija [200].

Пусть \mathfrak{R}^+ — множество неотрицательных действительных чисел. Тогда дифференциальные уравнения можно представить как $\dot{x} = f(x)$, где $x \in \mathfrak{R}^n$ и $f: \mathfrak{R}^n \rightarrow \mathfrak{R}^n$. «Дифференциальное включение» есть $\dot{x} \in f(x)$, где $x \in \mathfrak{R}^n$, а f — многозначное отображение \mathfrak{R}^n в \mathfrak{R}^n , т.е. $f(x) \subset \mathfrak{R}^n$.

Гибридный автомат имеет конечное множество позиций управления. В каждой позиции непрерывное состояние изменяется в соответствии с некоторым дифференциальным включением. Ребра между позициями помечаются условиями необходимости. Из одной позиции управления в другие позиции может быть сделан скачок, когда условия необходимости будут выполнены.

Формально гибридный автомат есть:

$$H = (L, D, J, \mu, \Delta) , \quad (4.6)$$

где L — множество управляющих позиций; D — связывает дифференциальное включение $D(l)$ с позицией $l \in L$; $J \subset L \times \mathfrak{R}^n$ — множество начальных состояний; μ — связывает условие инвариантности $\mu(l) \subset \mathfrak{R}^n$ с позицией l ; $\Delta \subset L \times \Phi \times \Lambda \times L$ — множество ребер, причем $e = (l, \phi, \lambda, l') \in \Delta$ — ребро из позиции l в позицию l' , помеченное условием необходимости $\phi \in \Phi$ и отношением инициализации $\lambda \in \Lambda$.

Состояние гибридного автомата H есть $(l, x) \in L \times \mathfrak{R}^n$ и изменяется двумя различными путями. В первом — позиция l сохраняется постоянной и x преобразуется в соответствии с дифференциальным включением $D(l)$. Для $t = 0$ имеем $(l, x) \xrightarrow{0} (l, x)$. Для $t > 0$ имеем $(l, x_0) \xrightarrow{t} (l, x)$, где $x = \varphi(x_0, l)$ и $\varphi(x_0, [0, t]) \subset \mu(l)$, где $\varphi(x_0, l)$ — решение дифференциального уравнения $\dot{x} = D(l)(x)$.

Второй путь — выполнение дискретного преобразования из позиции l в позицию l' , т.е. $(l, x_0) \xrightarrow{(\phi, \lambda)} (l', x)$, где $x_0 \in \phi$, $x \in \lambda[x]$ и $(l, \phi, \lambda, l') \in \Delta$. Вводится также понятие множества достижимых состояний, элементы которого достижимы из начального состояния за некоторое количество последовательных моментов времени и дискретных переходов.

Гибридная система состоит из взаимосвязанных компонентов (4.6) с координацией между ними. Каждый компонент моделируется гибридным автоматом, а координация между компонентами определена как произведение гибридных автоматов. Пусть дано M компонентов. Определим алфавит или множество событий гибридной системы как $\Sigma = L_1 \times \dots \times L_M$, где L_i — множество позиций управления для компонента i . Непрерывное состояние гибридной системы есть,

$$\mathbf{x} = (x_1, \dots, x_M) \in \mathfrak{R}^n \text{ где } x_i \in \mathfrak{R}^{k_i} \text{ и } \sum_{i=1}^M k_i = n . \text{ Непрерывное состояние } i -$$

го компонента есть $x_i \in \mathfrak{R}^{k_i}$. Условия необходимости на ребрах — функции общего состояния как комбинации позиций управления и непрерывных состояний других компонентов. Координация компонентов $H_i, i = 1, \dots, M$ определяется как произведение:

$$H = H_1 \otimes \dots \otimes H_M. \quad (4.7)$$

Позиции управления в произведении гибридных автоматов (4.7) — декартово произведение позиций управления компонентов. В точке $l = (l_1, \dots, l_M)$ компонента x_i непрерывного состояния эволюционирует в соответствии с дифференциальным включением компонента i в позиции l_i . Переход может быть сделан из одной позиции в другие в пределах одного автомата или в другие компоненты.

В работах Г.С. Осипова 1998 — 1999 г.г. [203] рассматривается новый класс систем, выделившихся из гибридных систем — гибридные системы с дискретной частью, основанные на знаниях, например, на правилах или семантических сетях. К ним относятся системы управления сложным движением технических объектов, эко-биологические системы, где состояние описывается набором параметров, а некоторые из законов поведения носят характер эмпирических правил.

Динамику гибридной системы, основанной на знаниях, автор описывает следующим уравнением:

$$S(t+1) = \phi(\cup F_i(t) \cup \varphi(S(i), U(i), RD), RS), \quad (4.8)$$

где \cup — теоретико-множественное объединение берется по всем $j \leq t+1$ и всем $i \leq t$; $U(i)$ — некоторый факт или множество фактов, добавляемый в состояние $S(i)$, который называется управлением; RS и RD — компоненты базы правил, содержащие правила замыкания и перехода соответственно; ϕ и φ — процедуры управления выполнением правил из множеств RS и RD соответственно.

Вычисления по (4.8) организованы следующим образом. С каждым правилом из множества RD связано действие исполнительного органа во внешней среде или действия по изменению состояния DB_1, \dots, DB_n базы данных DB . Правила из RS не изменяют окружающую действительность, они изменяют наши знания о ней и могут быть заменены подходящей теорией изучаемой действительности.

Вычисления начинаются с построения описания состояния, которое по существу есть замыкание начального множества фактов. При этом выполняются процессы, для которых «готовы» фактические параметры и можно применить правила RS . Процесс применения правил RS прекращается при исчерпании множества применимых правил или стабилизации состояния DB . Таким образом, каждое состояние модели -

результат решения уравнения неподвижной точки $\phi(S) = S$. С таким «стабилизировавшимся» состоянием S работает процедура ϕ , применяющая правила RD к S_i . После перевода модели из состояния $S(t)$ в состояние $S(t+1)$ к последнему вновь применяется замыкание ϕ .

Рассмотренная модель обеспечивает естественную интеграцию различных парадигм моделирования. Она легко адаптируется к изменениям среды, имеет невысокую «чувствительность» к априорной полноте описания состояния. Модели присущи и не столь жесткие требования к формализму, что становится принципиальным при изучении поведения сложных открытых систем, описываемых как количественными, так и качественными параметрами. Состояния таких систем часто не имеют полного описания, а законы поведения могут иметь как замкнутый вид, так и задаваться множеством эмпирических или экспертных правил и связей.

4.4. Интегрированные экспертные системы

Интегрированная экспертная система — это программная система, в архитектуре которой наряду с компонентом «экспертная система», используемым для решения неформализованных задач, как правило, методологию простых продукционных экспертных систем, содержатся компоненты « N », расширяющие функциональные возможности экспертных систем [64].

Анализируя модели интеграции знаний, Г.В. Рыбина подчеркивает, что понятие «интегрированная интеллектуальная система» не синонимично понятию «гибридная интеллектуальная система» (см. разд. 4.2.3) и является более широким, чем понятие «гибридная интеллектуальная система». Интегрированная интеллектуальная система не обязательно гибридная и наоборот; интегрированная интеллектуальная система — всегда гибридная только в случае полной интеграции компонентов [64].

Ниже приводится классификация интегрированных экспертных систем Г.В. Рыбиной, которая различает: системы с поверхностной и с глубинной интеграцией компонентов (рис. 4.8). При поверхностной интеграции взаимодействие компонентов осуществляется на уровне передачи сообщений. В более сложных вариантах поверхностной интеграции один из программных компонентов в ходе работы обращается для уточнения данных или решения подзадач к другому компоненту. Здесь идет более тесное сопряжение ЭС и N , например, «программами-мостами».

Глубинная интеграция компонентов предполагает модификацию с передачей им функций, отсутствующих в классических типах этих компонентов.

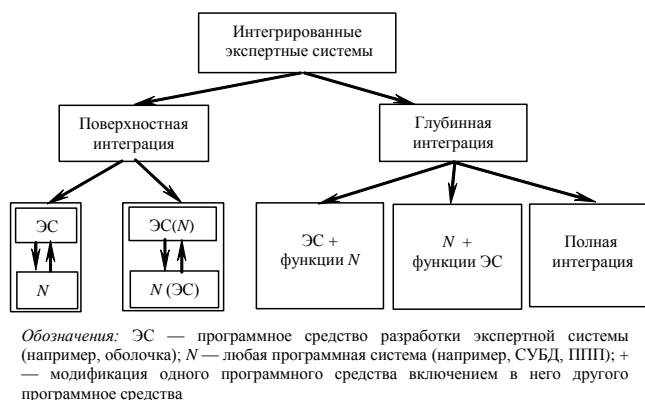


Рис. 4.8 — Классификация интегрированных экспертных систем
Г.В. Рыбиной

Полная интеграция — высший уровень интеграции, предполагающий соединение лучших качеств компонентов. Например, полная интеграция СУБД и ЭС состоит в выявлении плюсов и минусов в механизмах СУБД и ЭС, в определении новых моделей данных и знаний, унификации структур моделирования фактов и правил и однородной обработки данных и знаний.

В табл. 4.1 рассмотрено сочетание интеграции с необходимостью выполнения гибридизации методов, моделей и алгоритмов.

В случае сильного сцепления предъявляются более серьезные требования к системе и в оболочку включаются библиотека функций, работающих с СУБД, как сделано в KAPPA-PC. В оболочке GURU (ИНТЕР-ЭКСПЕРТ) интеграция обеспечивалась включением в посылки (левую часть) продукционного правила: рабочих переменных с одним значением; многозначных нечетких переменных; статистических переменных; ячеек электронных таблиц; полей реляционной базы данных; числовых функций; операторов отношений, булевых операторов, операторов над числами; операций над текстовыми строками; символов. В другом программном продукте G2 (Gensym Corp.) интеграцию обеспечивают специальные «мостики».

Интеграция экспертных и обучающих систем. Наибольшую популярность здесь получили обучающие экспертные системы, экспертно-обучающие системы и интеллектуальные обучающие системы. В них в

архитектуру системы дополнительно введены компоненты N , поддерживающие модель обучаемого, по которой может быть определена текущая подцель обучения и обеспечивающие наблюдение за обучаемым.

Таблица 4.1 — Интеграция и гибридизация

Способы интеграции	Необходимость гибридизации
Поверхностная интеграция компонентов	Отсутствует (используются автономные модели, методы и алгоритмы)
Глубинная интеграция компонентов (ЭС + функции N)	Слабая (используются в основном автономные модели и методы с частичной модификацией отдельных моделей, методов и алгоритмов)
Глубинная интеграция компонентов (N + функции ЭС)	Слабая (используются в основном автономные модели и методы с частичной модификацией отдельных моделей, методов и алгоритмов)
<p>Полная интеграция</p> <p>1) Селекция лучших особенностей, функций и механизмов ЭС и N;</p> <p>2) Определение новых моделей данных и знаний;</p> <p>3) Построение систем с новыми функциями и возможностями</p>	Сильная (разрабатывается унифицированная структура для моделирования фактов и правил; присутствует однородная обработка данных и знаний; все компоненты (модули) находятся в одной системе и т.д.)

Интеграция экспертных и гипертекстовых систем. Практическая направленность подобных систем: «интеллектуальные тексты», «эксперт-тексты», адаптивные модели пользователей, актуальна из-за распространения Интернет-технологий, электронных энциклопедий и т.д. При этом возможности экспертных систем добавляются к гипертексту, что позволяет имитировать рассуждения экспертов при поиске релевантных фрагментов текста, особенно в предметных областях регламентируемых нормативными документами и электронном бизнесе.

Интеграция экспертных систем и пакетов прикладных программ. Интеграция ЭС и ППП идет на уровне обмена сообщениями. Типичный пример — система ПРИИУС [204], объединяющая в своей архитектуре: ЭС, фактографическую информационно-поисковую систему, а также программы на языках Си, Паскаль, ФОРТРАН. В системе БОСЭС [205] для проектирования летательных аппаратов используются математические модели, идентификаторы которых разрешено применять в левых частях производственных правил.

Интеграция экспертных и имитационного моделирования систем (см. также разд. 4.6). Необходимость такой интеграции объясняется проблематикой представления и работы с категорией «время» в экспертных системах. Так, например, в системе СИГМА ТУ [206] разработанный экспертной системой сменно-суточный план морского порта

передавался на «воспроизведение» в автоматную модель, которая не только имитировала поведение ресурсов порта относительно времени, но и выдавала для анализа качества плана количественные показатели.

Интеграция экспертных и приобретения знаний систем. Это направление интеграции призвано уменьшить влияние одного из главных недостатков экспертных систем — получение профессиональных знаний для наполнения баз знаний. В этой связи в инженерии знаний была сформулирована проблема приобретения знаний (англ. knowledge acquisition), появилась методология KADS (англ. knowledge acquisition and documentation structuring) и инструментальные программные средства, ее поддерживающие, например, ROGET, MOLE, OPAL, KNACK, МЕДИКС, SIMER+MIR для работы с экспертами.

Следует отметить, что большинство коммерческих программных продуктов для разработки экспертных систем по-прежнему не содержат средств автоматизированного извлечения знаний из источников.

Интеграция экспертных систем с другими программными продуктами. В современных условиях интеграционные процессы наиболее заметны оживились в архитектурах систем, предназначенных для реинжиниринга бизнес-процессов предприятий. Здесь важна общность жизненных циклов разработки экспертных систем и объектно-ориентированного подхода к бизнес-процессам, а также сходство понятий «производственное правило» и «бизнес-правило». Перспективно совместное применение геоинформационных и экспертных систем.

4.5. Методы интеграции знаний в условиях расхождения мнений экспертов

Применение рассмотренных в разд. 4.1 — 4.4 моделей и методов интеграции знаний экспертов может проходить в различных условиях и ситуациях эксплуатации КСППР. Некоторые из таких обстоятельств осложняются для ЛПР еще и необходимостью учета в своих рассуждениях таких факторов, как: 1) независимости мнения экспертов; 2) конфронтации экспертов и широкий разброс их мнений; 3) мнение меньшинства. Ниже рассмотрены два метода интеграции поиском консенсуса (лат. consensus) в условиях расхождения мнений экспертов [22].

Модифицированный метод Дельфи. Метод Дельфи в современном понимании — процесс, в результате которого участники СППР (или независимые эксперты) приходят к консенсусу, не прибегая к дискуссии лицом к лицу. Метод Дельфи способствует выработке независимости мышления членов группы, препятствует конфронтации экспертов, лишая их возможности отстаивать свои идеи. Поиск решений проблемы данным методом позволяет учесть мнение меньшинства, а в отдельных

случаях оно может стать решающим. Существенное преимущество в том, что эксперты с широким разбросом мнений могут совместно получать информацию о позициях других экспертов, что способствует переосмыслению поставленной проблемы и достижению согласия по альтернативам решения проблемы.

Рассмотрим реализацию метода Дельфи.

1. Задача должна быть сформулирована, собраны и интерпретированы все данные. Альтернативы выявляются мозговым штурмом или самостоятельно экспертами.
2. ЛПР составляет общий список решений. Повторения считаются одним предложением. Нельзя указывать, что решение было высказано несколькими экспертами. Как только общий список будет составлен, ЛПР копирует его по числу экспертов и раздает им.
3. Каждый эксперт, получив список решений, оценивает важность варианта решения и записывает свои оценки в специальную таблицу.
4. После заполнения таблиц ЛПР обрабатывает мнения экспертов. Задача — выявить и обобщить наиболее часто повторяющиеся оценки по каждому варианту решения. Полученные от экспертов данные сводятся в общую таблицу.
5. Если мнения экспертов близки, то окончательный результат получен. При наличии существенных разногласий процесс продолжается. Для этого на очередном заседании СППР руководитель знакомит экспертов с ранее сформулированными ими решениями и мнениями о выставленных оценках. Если разброс мнений большой по всем вариантам предложенных решений, ЛПР готовит обобщенную информацию, давая мнения всех экспертов по каждому варианту решения. Если есть противоположные мнения, то мнение большинства передается на обсуждение меньшинству и наоборот.
6. ЛПР собирает оценки экспертов и обрабатывает их аналогично шагу 4, сравнивает новые результаты с имеющимися. Если большинство участников оценили различные варианты решения почти одинаково, нет крайних оценок, то обсуждение заканчивается, и выделяются приоритетные варианты для дальнейшего рассмотрения. Если остались полярные мнения, то повторяются шаги 5 и 6 для отдельного обсуждения мнения меньшинства. В случае отсутствия согласия экспертов по некоторым вариантам решений используется метод «Обмен мнениями».

Метод обмена мнениями. Данный метод применяется в работе СППР, когда мнения экспертов разделились, и каждая коалиция твердо настаивает на своей позиции. Обмен мнениями приводит к лучшему пониманию точки зрения другой стороны и исключает конфронтацию.

Этот метод состоит из пяти шагов.

1. Эксперты делятся на две подгруппы с одинаковыми точками зрения. Подгруппе предлагается подготовить доклад по своему варианту решения. Далее проводится презентация мнения подгруппы для противоположной стороны.
2. Подгруппы готовят доклад по мнению противоположной стороны как своего мнения. Представление «чужого» мнения должно быть точным и беспристрастным. Кроме того, эксперты, представляя другое мнение, должны найти свои новые, отличные от представленных ранее аргументов в пользу данного варианта решения.
3. Каждая подгруппа проводит свою презентацию, могут быть заданы уточняющие вопросы, но без оценок.
4. После презентаций каждая из подгрупп подвергает критике решение противоположной стороны. При этом излагаются свои чувства, взгляды и реакция на происходящее без осуждения намерения противоположной подгруппы. Все мнения фиксируются.
5. После анализа информация с главными критическими аргументами выводится на табло вместе с материалами обоих докладов и ЛПР проводит дискуссию для достижения полного согласия.

Успех применения метода обмена мнениями зависит от настроения экспертов и способности ЛПР создать конструктивную обстановку.

4.6. Методы гибридных интеллектуальных систем

На сегодняшний день в мировой практике известны пять классификаций ГиИС. Исследуя искусственные нейронные сети и экспертные системы, L. Medsker в 1990 — 1995 гг. предложил описывать уровень их интеграции, используя понятия «автономных» (англ. stand-alone), «трансформационных» (англ. transformational), «слабо - и сильносвязанных» (англ. loose and tight coupled) и «полностью интегрированных» (англ. full integrated) моделей [74, 207].

S. Goonatilake и S. Khebbal в 1992 — 1995 гг. [208] расширили интегрируемые автономные модели нечеткими системами, генетическими алгоритмами и предложили три класса гибридов: «функциональнозаменимых» (англ. function replacing), «внутрисвязанных» (англ. intercommunicating) и «полиморфических» (англ. polymorphic). Первый класс объединяет объекты, в которых важные и в то же время слабые функции одного из методов заменяются сильными функциями, чаще всего другого метода. Связанность авторы трактуют аналогично L. Medsker, а полиморфизм понимают как эмуляцию гибридом нескольких методов. При этом он способен, как и хамелеон, изменять форму функционирования. К этому классу относятся, например, нейро-нечеткие гибриды.

М. Hilario в 1995 г. определила еще одну классификацию нейро-символьной интеграции [73]. Она различает «унифицированные» (англ. unified) коннекционистские модели как универсальные схемы для символьных моделей, а также «трансляционные» и «функциональные» гибриды. Последний класс по существу совпадает с «функциональнозаменимыми» гибридами S. Goonatilake и S. Khebbal, однако, автором выделено четыре типовых класса архитектур такой замены: цепочные, подчиненные, метапроцессорные и сопроцессорные архитектуры, подробнее рассмотренные ниже.

В работе [76], в рамках нейро-символьной интеграции, гибриды обобщены в классы одно- и многомодульных архитектур. Первые — это попытка объединить унификацию и трансляцию в смысле М. Hilario с автономными и трансформационными моделями L. Medsker. Гомогенные многомодельные гибриды включают несколько копий одной и той же базовой структуры, например экспертной системы, используемой для обработки одних и тех же значений входов. Гетерогенные многомодельные гибриды имеют много комбинаций разнородных автономных моделей. Эти гибриды автор классифицирует так же, как и L. Medsker, по связанности. Нововведение R. Sun — «грануляция» (англ. granularity), разбивающая гибриды на два класса: «крупно» - и «мелкозернистых» объектов (англ. coarse- and fine-grained) и различающая тем самым интеграцию на макро- и микроуровнях соответственно.

Наконец, в работе [209] введена еще одна классификация ГиИС. Авторы различают «унифицированные», «трансформационные» и «модульные» гибриды, смещая тем самым акценты на модульную природу объектов, правда, только в рамках нейро-символьной интеграции. Первая группа, «унифицированные ГиИС», включает системы, выполняющие различные виды обработки, например символьную, эволюционную, вычисления по формулам и другие, которые реализованы нейросетевыми представлениями и нейровычислениями. Эту группу методов, выделенную и в классификации М. Hilario, поддерживают специалисты, обосновывающие когнитивные процессы, в том числе и высокоуровневые, только через коннекционистские структуры. Наиболее ярко это направление представляется нейронным Дарвинизмом [210]. В унифицированных системах преимущества комбинирования символьных структур с индуктивными возможностями ИНС раздроблены (распылены) по архитектуре гибридов, использующих специальные нейросетевые элементы для выполнения символьных функций. Такие системы применяются ограниченно из-за трудностей разработки и эксплуатации. Трансформационные гибриды авторы понимают аналогично L. Medsker. Модульные ГиИС — системы, модульные по своей природе. Например, они обычно включают несколько нейросетевых модулей и модулей,

218

ориентированных на правила. Эти модули могут иметь различные уровни интеграции путем связывания в пары.

Из краткого обзора классификаций видно, что все они, имея много общего, идентифицируют следующие пять стратегий разработки ГиИС [1] (рис. 4.9): автономные, трансформационные, слабосвязанные, сильносвязанные и полностью интегрированные модели.

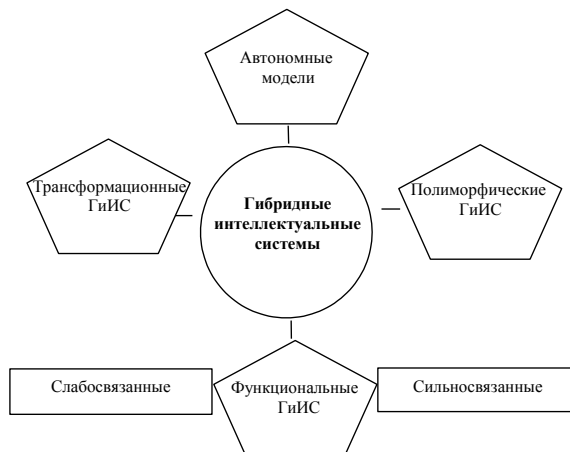


Рис. 4.9 — Классификация гибридных интеллектуальных систем

Автономные модели приложений ГиИС содержат независимые программные компоненты, реализующие обработку информации на моделях с использованием методов из ограниченного числа классов. Несмотря на очевидную вырожденность интеграции знаний в этом случае, разработка автономных моделей актуальна и может иметь несколько целей. Такие модели — способ сравнения возможностей решения задачи двумя или более различными методами. Новая автономная модель для решения решенной задачи верифицирует уже созданное приложение и приводит к адекватным моделям. Автономные модели могут использоваться для быстрого создания начального прототипа, после чего разрабатываются приложения, требующие большего времени.

Автономные модели имеют и существенный недостаток — никакая из них не может помочь другой в ситуации обновления информации — все должны модифицироваться одновременно.

Трансформационные ГиИС похожи на автономные, так как конечный результат разработки — независимая, не взаимодействующая с другими частями модель. Основное отличие состоит в том, что такая

модель начинает работать как система, использующая один автономный метод, а заканчивает как система, использующая уже другой метод.

Трансформационные модели дают несколько преимуществ: быстроту создания и меньшие затраты, поскольку эксплуатируется единая модель, а окончательный метод наилучшим образом адаптирует результаты к окружению. Есть и проблемы: автоматическое преобразование одной модели в другую; существенная модификация модели, сравнимая по объему с разработкой «заново».

Слабосвязанные ГиИС — это, по существу, первая реальная форма интеграции, когда приложение разбивается на отдельные элементы, связываемые через файлы данных. Классификация таких моделей рассмотрена ниже. Цепочные ГиИС используют как составные части два функционально завершенных элемента, один из которых — главный процессор, а другой — пре- или постпроцессор. В подчиненных ГиИС составные части — функционально завершенные элементы. Однако в этом случае один из них, подчиненный, включенный в другой, — главный решатель задачи. Метапроцессорные ГиИС используют как составные части один метапроцессор и несколько функциональных элементов. Сопроцессорные ГиИС при решении задачи применяют элементы, как равные партнеры. При этом каждый может передавать информацию каждому, взаимодействовать, обрабатывая подзадачи одной задачи.

Если сравнить рассмотренные слабосвязанные ГиИС с другими более интегрированными приложениями, то они проще для разработки и допускают применение коммерчески доступных программ, снижающих время программирования. Время на эксплуатацию сокращено из-за простоты интерфейсов файлов, однако увеличена цена коммуникации и ниже производительность ГиИС.

Сильносвязанные ГиИС имеют значительное перекрытие с классом слабосвязанных гибридов. Однако последние используют обмен информацией через резидентные структуры памяти, например DDE (англ. *date dynamic exchange*), в отличие от обмена через внешние файлы в слабосвязанных ГиИС. Это улучшает интерактивные возможности и дает более высокую производительность. Сильносвязанные модели могут функционировать в тех же формах, что и слабосвязанные, однако их пре-, пост- и сопроцессорные варианты по природе быстрее. Здесь используются следующие методы разработки: «доска объявлений», «конвейерная модель» [1] и «вложенные» системы.

Сильносвязанные ГиИС имеют низкие коммуникационные затраты и более высокую производительность по сравнению со слабосвязанными моделями. Тем не менее, эти ГиИС имеют и три принципиальных ограничения: 1) сложность разработки и поддержки возрастает как след-

ствие внешнего интерфейса данных; 2) сильная связанность страдает от излишнего накопления данных и 3) проверка адекватности затруднена.

Рассмотренные выше слабо- и сильносвязанные ГиИС в силу того, что их состав и структура во многом зависят от решаемой задачи, принято называть, следуя классификации М. Hilario, еще и функциональными ГиИС — предмет настоящей книги.

Полностью интегрированные ГиИС совместно используют общие структуры данных и представления знаний, а взаимосвязь между компонентами достигается посредством двойственной природы структур. Это бурно развивающийся в мировой практике класс гибридов, где можно выделить разработку концептуальных нейросетей, основанных на знаниях, коннекционистских экспертных систем, в которых элементы взаимодействуют быстро и просто, а общая информация для независимого решения задачи мгновенно доступна тому и другому компоненту. Еще один вариант полной интеграции — нечеткие нейросети — гибрид, по структуре похожий на нейросеть и реализующий одновременно нейро- и нечеткие вычисления [157, 159].

Преимущества полной интеграции — надежность, увеличение скорости обработки, адаптация, обобщение, снижение шума, аргументация и логическая дедукция, то, чего в сумме не найти ни в одном классе методов-родителей.

В мировой практике, особенно за рубежом, за последние десять лет накоплен значительный опыт гибридизации с целью разработки ГиИС с архитектурой из двух классов базисных методов (дигибридов [1]). В этой связи в [1] даны классификация и обзор архитектур дигибридов, а ниже приводится описание некоторых методов ГиИС, особенно популярных у разработчиков.

Архитектуры «экспертная система — нейросеть». Интеграция экспертных систем и нейросетей началась с 1990 года и идет различными путями. Рассмотрим наиболее известные из них: 1) интеграцию искусственных нейросетей и экспертных систем в ГиИС; 2) разработку коннекционистских экспертных сетей и унифицированных нейросетей; 3) повышение качества знаний за счет использования нейросетей; 4) интеграцию знаний о предметной области в нейросетях; 5) извлечение правил из обученных нейросетей для использования в экспертных системах; 6) конвертирование нейросетей в деревья решений или наоборот.

Первый подход — снижение слабостей друг друга в гибридных «экспертно-нейросетевых» системах. Одна из проблем экспертных систем — приобретение знаний, в то время как нейросети накапливают знания из обучающих примеров. Слабейшая сторона нейросетей — объяснительные способности, прекрасно реализованные в экспертных си-

стемах. В этом случае создается гибрид, решающий обе проблемы одновременно. Он должен обучаться на примерах и уметь давать объяснения. Разработать такой гибрид можно двумя методами. Один из них — это редуцировать задачу на части и каждую из подзадач решить экспертной системой или нейросетью. Это позволяет преодолеть часто встречающиеся на практике провалы (англ. gaps) в знаниях экспертов. Нейросеть, как родитель, может «взять на себя» решение подзадач, требующих быстрой реакции, в то время как экспертная система — поддерживать решения в статичной среде. Перспективно использование нейросетей как машины вывода в экспертных системах — ИНС может решать задачу, а ЭС только объяснять. Наконец, экспертные системы используются как препроцессор инициализации нейросети, выбора топологии, алгоритма обучения, начальных значений весов и смещений нейронов.

Второй подход — разработка коннекционистских экспертных систем, т.е. специальных программ, имитирующих обучение и вычисления в нейросети, в которых каждому нейрону сопоставлен символичный эквивалент — понятие естественного языка. Этот способ позволяет для специализированных нейросетей извлекать скрытые от разработчика знания. Пример такого гибрида — система MACIE (Matrix Controlled Inference Engine) [211]. Еще один пример — CONSYDERR (CONnectionist SYstem with Dual-representation for Evidential Robust Reasoning) [212] — унифицированная ГиИС с двухуровневой нейросетью. Система создана, чтобы преодолеть хрупкость экспертных систем. Хотя CONSYDERR полностью выполнена нейросетями, она способна рассуждать на символических, логико-лингвистических представлениях с помощью тщательно спроектированной иерархии уровней.

Третий подход к повышению качеств знаний за счет использования нейросетей основывается на идее совершенствования поверхностных знаний извлечением глубинных [98], применяемой в лингвистике. Язык профессиональной деятельности содержит помимо качественных единиц зашумленные численные данные, на которых можно обучить нейросеть. После обучения она способна контрастировать классы и кластеры, полезные для совершенствования знаний о проблемной среде.

Четвертый подход — интеграция знаний о предметной области в нейросетях, т.е. включение знаний о предметной области в нейровычисления с целью совершенствования последних. Полезность или бесполезность конкретного входа нейросети, число нейронов в слое неконтрастны без знаний о предметной области, которые можно получить из экспертной системы. Знаниями из ЭС также инициализируют веса, близкие к оптимальным.

Пятый подход — извлечение правил «условие-действие» из обученных нейросетей — одно из самых популярных направлений исследований. Прямой путь конвертирования нейросетевых знаний в символичные представления — это извлечение продукционных правил. Извлечение правил выполняется для различных нейросетей: многослойных персептронов [213], карт Кохонена [214], сетей с радиальными функциями [215] и рекуррентных сетей [216]. Выгоды от этого следующие: 1) обеспечение объяснительных способностей оценкой извлеченных правил для различных входных исходных конфигураций; 2) идентификация неполноты обучающих последовательностей и улучшение обобщающих возможностей добавлением новых классов или расширением существующих; 3) анализ предварительно неизвестных связей в данных, что имеет огромный потенциал для обнаружения-извлечения данных и индукции; 4) появляется возможность вставлять однажды извлеченные из сети правила для одной предметной области в другую нейросеть, уже для другой предметной области.

Типичный пример реализации пятого подхода — трансформационные ГиИС: ориентированные на знания концептуальные нейросети (англ. knowledge based conceptual neural network — KBCNN [217]), а также гибридные коннекционистские сети, основанные на знаниях (англ. hybrid knowledge-based connectionist network — НКВСН [218]), которые могут изменяться и обучаться знаниям трансляцией правил предметной области в нейросеть. В KBCNN правило может быть отображено в нейросеть и, наоборот, алгоритм обучения изменяется так, чтобы нейроархитектура модифицировалась адаптивно.

Еще один пример, — методология искусственных нейронных сетей, использующих знания (англ. knowledge based artificial neural network — KBANN [219]), состоит из двух этапов, поддерживаемых соответствующими алгоритмами. Первый вставляет приблизительно корректные, символичные правила в нейросеть. Второй алгоритм уточняет нейросеть обратным распространением. Сеть KBANN обобщает лучше, чем любая другая нейросеть и с меньшим значением ошибки.

Трансформационные экспертные нейросети для уточнения правил предложены в [220] для улучшения исходного множества правил, полученных от эксперта. По существу — эти системы предназначены для преодоления проблемы извлечения знаний. Эксперименты показали, что экспертные нейросети лучше обобщают, чем классические нейросети и требуют значительно меньше обучающих примеров.

Другая группа методов извлекает правила в процессе вычислений из обученной нейросети [219]. Третья группа методов — это создание процедур, встраиваемых в алгоритмы обучения нейросетей и работающих с ними параллельно. Извлечение в этом случае рассматривается как зада-

ча обучения. В качестве примера можно привести метод RULENEG [221] и Rule-Extraction-As-Learning [222].

Рассмотренные гибриды имеют и недостатки. Извлеченные правила могут и не давать четкой и ясной картины нейросетевого отображения входа в выход. Применение извлеченных правил может быть и менее точным, чем нейровычисления.

Наконец, последний, шестой подход основан на идее, что деревья решений более понятны для человека, чем нейросети. Время построения деревьев экспоненциально зависит от числа обучающих примеров, а нейровычисления уже фактически создали эти деревья и их необходимо только выделить и преобразовать в экспертную систему. Пример — система NNES (англ. a neural network explanation system for transforming trained neural networks into discussion trees [223]). На этом пути есть проблемы плохой адекватности (около 65 %) нейросетей и деревьев решений, а также их большие размеры для сложных задач.

Архитектуры «экспертная система — нечеткая система». Гибридикация здесь идет в следующих направлениях. Первое — встраивание элементов с нечеткой математикой в правые части правил экспертных систем (например, система EXSYS или ее вариант ReSolver фирмы MultiLogic [224]). В этом случае используется интерпретатор экспертной системы со всеми недостатками продукционных систем. Разработчик может расширить образцы данных до нечетких лингвистических переменных.

Второе направление — нечеткие экспертные системы (англ. fuzzy expert systems), например, оболочка FuzzyCLIPS [225] для создания экспертных систем, манипулирующих нечеткими фактами и правилами. FuzzyCLIPS обеспечивает точный, нечеткий и комбинированный вывод, разрешая перемешивание нечетких лингвистических переменных и качественных характеристик в левой и правой частях правил.

Перспективно также применение экспертных и нечетких систем как элементов функциональных ГиИС, используя их разную производительность. Первые значительно лучше, чем нечеткие системы, выглядят как управляющие компоненты в ГиИС, в то время как нечеткие системы лучше как управляемые модули. С другой стороны, нечеткая система привлекательнее для организации интерфейса с пользователем, привыкшим рассуждать приближенно. В гибридах экспертных и нечетких систем можно гибко манипулировать разными уровнями рассуждений — более абстрактным и отсюда более компактным в нечеткой системе и более точным и значительно более детальным в экспертной системе.

Архитектуры «нейросеть — нечеткая система». Разработка таких приложений началась с 1992 г. Нейро-нечеткие системы могут быть разделены на две группы: нейронные нечеткие системы вывода и нечет-

кие нейросети. Идея нейронных нечетких систем вывода состоит во внедрении таких нейронных парадигм, как обучение и параллелизм в нечетких системах. Нейронные нечеткие системы реализуют нечеткие вычисления. Архитектура таких систем параллельная и они используют те же обучающие алгоритмы, что и нейросети. Нечеткие вычисления осуществляются двумя путями: 1) использованием одной нейросети, реализующей всю нечеткую систему; 2) использованием для реализации каждого нечеткого правила отдельной нейросети, когда нечеткий вывод — результат нескольких ИНС [226].

В качестве примера рассмотрим нейронные нечеткие системы вывода, обучающиеся обратным распространением ошибки. В основе таких систем — представление вычислений в нечеткой системе многослойной сетью специального вида [157, 159]. Такой объект очень похож на нейросеть. Однако у нее две главных особенности. В отличие от «чистых» нейростетей, каждый из слоев с расположенными в них элементами, конфигурацией связей, а также все параметры и веса имеют физическую интерпретацию. Это чрезвычайно важное достоинство, когда знания не известны и могут быть с помощью наблюдателя из такой сети извлечены. Вторая особенность — такая структура обучается на примерах стандартным алгоритмом обратного распространения ошибки. Подобные архитектуры проектируются и используются с различными целями. Обучением сети «достраивают» функции принадлежности, а также формируют структуру нейро-нечеткой системы, что и специфицирует базу нечетких правил.

В [127] нейро-нечеткая система диагностирует заболевания мозга по энцефаллограммам. При этом сравнивались база знаний, извлеченная из эксперта, и база знаний, сформированная автоматически на обучающих данных энцефаллограмм. Результаты показали более высокое качество автоматически извлеченных правил, чем правил от эксперта-человека.

В последнее время интенсивно развивается класс нейро-нечетких систем, использующих самоорганизующиеся карты Кохонена [1, 227].

В нечетких нейросетях [228, 229] нечеткая математика внедрена в нейросети.

Архитектуры «генетический алгоритм — нечеткая система». Нечеткие и генетические системы хорошо работают в похожих средах с нелинейностями и жесткими требованиями к высокой производительности. Поэтому эти два альтернативных метода используются в ГиИС — как автономные, трансформационные модели и связанные модули. Использование функций принадлежности вносит гибкость и обобщения, упрощающие проектирование систем высокой сложности. Генетические алгоритмы облегчают оптимизацию функциональных характеристик нечетких систем. При разработке нечетких систем генетические алго-

ритмы используются для настройки значений принадлежности, упрощения разработки функций принадлежности и получения нечетких правил. Нечеткое управление может применяться в генетических алгоритмах как функция оценки. Менее исследовано создание систем с общими способностями обучения.

Общая архитектура интеграции генетических алгоритмов и нечетких систем показана на рис. 4.10. Например, индивидум в популяции может быть определен

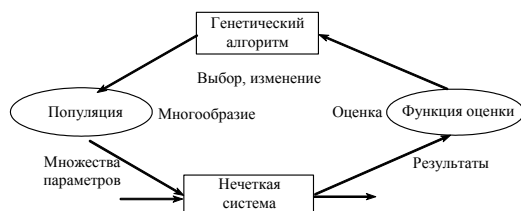


Рис. 4.10 — Архитектура генетической нечеткой системы

как конкатенация описаний всех функций принадлежности, примененных в нечеткой системе. Используя некоторое начальное множество значений параметров, нечет-

кая система дает на выходе значения-оценки приспособленности индивидуумов и формирует новую популяцию для генетического алгоритма. Этот цикл повторяется, пока не будут найдены лучшие множества значений параметров функций принадлежности.

Еще одна интересная архитектура интеграции — это включение в индивидум информации из левой и правой частей нечеткого правила. Иными словами, строится популяция индивидуумов, каждый из которых — закодированное нечеткое правило. Тогда генетический алгоритм может совершенствоваться в смысле функции оценки правила, используемые затем в нечеткой системе.

Архитектуры «генетический алгоритм — нейросеть». Заметное увеличение интереса к этому классу архитектур ГиИС началось с 1980 г., и большинство ученых использовало генетические алгоритмы для совершенствования технологии разработки и эксплуатации нейросетей. При этом прекрасные поисковые возможности генетических алгоритмов в больших пространствах применяются к подготовке данных для нейросетей и определению начальных значений параметров: числа нейронов в слое, весов и смещений. Совместное использование генетических алгоритмов и нейросетей в зарубежной литературе известно под названием COGANN (англ. Combination of Genetic Algorithms and Neural Networks). В [1] дана классификация и рассмотрено предназначение гибридов генетических алгоритмов и нейросетей.

Архитектуры «генетический алгоритм — экспертная система». Этот класс архитектур интеграции менее популярен у исследователей. Генетические методы применяются, чтобы найти или улучшить правила

для экспертных систем, а ЭС включаются в генетический цикл как эвристики, увеличивающие производительность генетических алгоритмов. Если в ГА аналитическая запись функции оценки затруднительна, то экспертная система используется как ее «аналог».

Архитектуры «экспертная система — имитационное статистическое моделирование». Этот класс архитектур развивается как «системы имитационного моделирования, основанные на знаниях». В традиционных исследованиях методом ИСМ построение модели, планирование экспериментов и анализ выходных данных выполняются самим разработчиком вручную, а качество результатов зависит от его опыта и интуиции. После завершения проекта искусство создателя утрачивается, так как извлечь знания из имитационной модели невозможно.

Известны следующие подходы к гибридизации. Прежде всего — это встраивание экспертных систем в моделирующие алгоритмы или наоборот. Так, например, ЭС могут понадобиться данные моделирования для того, чтобы предоставить их пользователю. Другой метод предполагает, что экспертная система дает ответы на вопросы о состояниях очереди, обслуживании и других текущих значениях параметров. Моделирующий алгоритм и экспертная система могут разрабатываться параллельно или взаимодействовать, в то время как пользователь имеет доступ к одной из них — управляющей. Вновь созданная экспертная система может быть оттестирована на имитационной модели объекта. Особенно это актуально для динамических экспертных систем. Или, наоборот, ЭС используется для начальной верификации моделирующего алгоритма и диагностики результатов имитации. Еще один подход предполагает, что экспертная система может помочь пользователю создавать моделирующий алгоритм и работать с законченной моделью, поясняя результаты моделирования в узкой предметной области. Обзор «экспертных систем имитационного моделирования» приведен в [230]. В [231] рассмотрена интегрированная среда моделирования IDS. Она структурирует информацию, комментирует результаты и содержит средства для визуального проектирования имитационных моделей.

Архитектуры «нейросеть — имитационное статистическое моделирование». Моделирующие алгоритмы и нейросети используются как партнеры при решении разных подзадач в функциональных ГиИС. Нейровычислениями решаются подзадачи, представленные черным ящиком, а моделирующим алгоритмом — задачи обслуживания. Возможно и последовательное использование моделирующих алгоритмов и нейросетей для пре- и постпроцессии. Например, моделирующие алгоритмы прекрасны для оценки качества поведения объекта управления при реализации плана субъекта управления. Однако в этом случае труд-

но реализовать систему предпочтений ЛПП. Это можно сделать нейросетью, обученной классификации решений.

Архитектуры «гибридные генетические алгоритмы». Основная концепция таких архитектур — применять генетические алгоритмы для оптимизации параметров эвристической стратегии решения задачи. Таким образом, возникает идея включить проблемно-специфические знания и эвристики, например, в генетические операторы. Такие алгоритмы называются «гибридными генетическими алгоритмами» (англ. hybrid genetic algorithms), пример архитектуры которых дан на рис. 4.11.

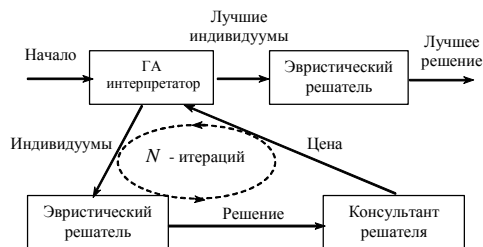


Рис. 4.11 — Архитектура гибридного генетического алгоритма

В эволюционный процесс вставлены два эвристических решателя, один из которых содержит знания, полезные для осуществления, например, селекции, и работает совместно с эвристическим консультантом, оценивающим решения-индивидуумы. Заметим, что в классическом ГА роль консультанта выполняла аналитически заданная функция оценки. Многократное повторение цикла эволюции дает некоторое множество индивидуумов, наилучший из которых выбирается еще одним эвристическим решателем.

Несколько иной подход предложен в архитектуре «гибридизированного генетического алгоритма» (англ. hybridizing genetic algorithm) [87]. В ней поиск оптимума — многошаговый процесс, и перед очередным шагом должно быть принято решение о том, с помощью какого метода продолжить решение задачи: одной из модификаций классического генетического алгоритма, случайным, пропорциональным поиском, наискорейшим спуском и др.

В [232] измерена эффективность шести различных генетических алгоритмов и четырех негенетических методов на сложной задаче планирования расписания. Сделан вывод о высокой эффективности гибридных распределенных генетических алгоритмов, которые позволили в 50 — 80 % случаев удовлетворить всем ограничениям.

Архитектуры «генетический алгоритм — имитационное статистическое моделирование», «нечеткая система — имитационное статистическое моделирование». Эти классы архитектур используются реже. Подробно они рассмотрены в [1].

4.7. Функциональные гибридные интеллектуальные системы

Главная отличительная особенность функциональных ГиИС [1] состоит в том, что их потенциал позволяет отобразить в представлениях данных и знаний, программных средствах их поддержки состав, структуру, свойства системности, неоднородности, динамичности и полиязыковый характер задачи-системы, что дает возможность моделировать в автоматизированном режиме решение сложных задач.

На рис. 4.12 сложная, неоднородная задача (объект-оригинал) и

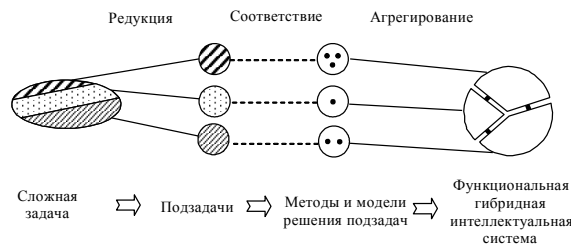


Рис. 4.12 — К понятию функциональной гибридной интеллектуальной системы

функциональная ГиИС (объект-результат) показаны вместе для построения методологии преобразования информации «оригинал → результат».

На рис. 4.12

показано, что сложная задача при разработке функциональных ГиИС рассматривается как неоднородная система, включающая несколько взаимосвязанных областей однородных параметров-переменных: детерминированных, стохастических, лингвистических, генетических, см. разд. 3.1, на которых одним или несколькими способами представлений: аналитическими, статистическими, экспертными, нечеткими, нейросетевыми, генетическими (см. разд. 3.2 — 3.4) заданы причинно-следственные связи в рассуждениях экспертов. В этих областях могут быть сформулированы одна или несколько подзадач. В СППР эти подзадачи решаются экспертами. Сложная задача в целом решается коллективными методами принятия решений, системой «ЛПР — эксперты». Особенности сложных задач рассмотрены в разд. 2.2.

Главное, что следует из этих особенностей, — это необходимость системного анализа сложной, неоднородной задачи, который должен обеспечить достижение в СППР определенного уровня понимания возникшей проблемы. Авторы настоящей работы убеждены, что без глубокого, системного анализа сложной задачи не только нельзя, а невозможно перейти к ее автоматизированному решению. Это следует из содержания понятия «задача-система», эмерджентности подобного рода субъективных сущностей. В этой связи природа (причина) возникнове-

ния интегративного свойства, обусловленная отношениями между подзадачами из разнородных областей, должна быть тщательно изучена.

Один из первых этапов углубленного системного анализа сложных практических задач в СППР — ее упрощение (редукция) по функциональному признаку. Это делается в СППР различными методами, которые будут рассмотрены в разд. 6.5. Цель редукции — выяснить, какие области однородных параметров существуют в задаче-системе, и сформулировать внутри этих областей другие более простые, одноаспектные задачи (однородные), которые уже нельзя соотнести с моделью «неоднородная задача» (5.10). Это особенно важно при разработке функциональных ГиИС потому, что, с нашей точки зрения, поручать экспертам решать неоднородные задачи — это ошибка, часто повторяемая как в СППР, так и КСППР. В СППР эта ошибка может быть обнаружена. В этом случае эксперт, скорее всего, объявит о том, что задача выходит за рамки его профессиональных знаний и опыта, и далее ошибка может быть исправлена за счет самоорганизации СППР, например, привлечение дополнительных экспертов. В КСППР подобная ситуация означает только одно, что редукция не завершена и должна быть продолжена. Редукция — это итерационный процесс отображения исходной задачи во множество языков профессиональной деятельности конкретной СППР с целью построения «проекции» сложной, неоднородной задачи на области однородных параметров. На рис. 4.12 исходная задача редуцирована на три подзадачи. Разными штриховками показано то, что задачи относятся к различным однородным областям переменных.

Следствие редукции — распределение подзадач между экспертами. Это распределение (субкоординация) может обдумываться и направляться ЛПР, т.е. быть централизованным, а может обдумываться и направляться экспертами, т.е. быть децентрализованным. В первом случае это выглядит императивно: «Вам (эксперту) поручается решение задачи», а во втором случае — декларативно: «Я (эксперт) могу высказать свое профессиональное мнение по такому-то аспекту решаемой задачи». Однако распределение может направляться не только профессиональной, предметной специализацией эксперта. Оно может учитывать и его инструментальную ориентацию. Например, аналитик — специалист, владеющий методами аналитических представлений, а статистик — специалист, владеющий методами статистических представлений.

Второй аспект редукции — построение декомпозиции, т.е. установление отношений на множестве подзадач. Это означает, что СППР внутри себя вырабатывает одну или несколько организационных форм взаимодействия членов коллектива: коалицию, сотрудничество и другие

(разд. 4.1). Это признак социальной самоорганизации СППР, гибридного интеллекта.

Поручение подзадачи эксперту означает многое. Прежде всего, от него требуются всесторонний анализ ситуации, сбор необходимых данных и собственно решение задачи. Однако к заседанию СППР обычно эксперты владеют необходимыми данными для выработки рекомендаций. Анализ должен помочь эксперту установить метод, приемлемый для рассуждений в осознанной ситуации решения задачи. Арсенал таких методов рассуждений был рассмотрен в разд. 3.

Иными словами, определяется соответствие на множествах подзадач, включенных в состав сложной задачи, и множестве методов рассуждений, которыми владеют приглашенные эксперты. Установление этого соответствия на рис. 4.12 обозначено пунктирными линиями, а черными точками внутри окружностей обозначена функциональная избыточность в ходе рассуждений экспертов. Она возникает, как следствие поручения нескольким экспертам дать консультации по решению одной и той же подзадачи или когда один эксперт, используя разные методы рассуждений, строит несколько моделей также для решения одной и той же подзадачи. Назовем это явление плюрализмом мнений экспертов в СППР. Такой плюралистический подход позволяет устраивать «спор» моделей для всестороннего рассмотрения сложных процессов и явлений и получения голографической модели [21] решения сложной задачи.

В итоге установления соответствия и моделирования рассуждений экспертов в СППР вырабатывается неоднородное (гетерогенное) модельное поле, релевантное сложности задачи. Однако простое суммирование результатов каждой линии рассуждений экспертов, результатов моделирования решения отдельных подзадач над гетерогенным модельным полем не даст ответа на стоящую перед СППР задачу. Поэтому гетерогенное модельное поле используется вместе с одной или несколькими декомпозициями сложной задачи, выработанными в ходе редукции. Рассмотрение их совместно — определяющее условие возникновения синергетических эффектов в СППР. Следствие — это преобразование декомпозиции исходной задачи заменой включенных в нее подзадач на модели их решения и замена отношений декомпозиции на отношения интеграции (см. разд. 6.2). В процессе замены могут использоваться разные «спорящие» модели. В итоге, комбинированием моделей из модельного поля синтезируется некоторая модель-система, функциональная ГиИС, релевантная декомпозиции сложной задачи, эксперименты с которой могут рассматриваться как выработка метода решения сложной задачи в СППР путем ее самоорганизации. Подобные действия не могут быть присущи экспертам, и они, скорее всего, выполняются ЛПР, игра-

ющим в данном случае роль интегратора. Параллельно с интеграцией мнений экспертов ЛПР координирует их линии рассуждений, согласовывая цели, ресурсы и время решения подзадач экспертами.

В рассматриваемом подходе ГиИС будет повторять (наследовать) функциональную структуру сложной задачи, чем и определяется специфика данного класса ГиИС.

Таким образом, в настоящей работе СППР рассматривается как инструмент выработки и применения метода решения сложной задачи. Этот метод существует только на время работы СППР и релевантен только конкретной ситуации решения задачи. В другой ситуации даже для решения похожей задачи потребуются повторная организация цикла «анализ сложной задачи → синтез функциональной ГиИС» и выработка нового метода.

Все вышесказанное сформулировано как результат наблюдений за работой СППР самого различного уровня от традиционного инструмента планирования «планерки» до «заседаний органов регионального и отраслевого управления», что позволяет высказать еще две гипотезы (гипотезы № 1, 2 см. в разд. 2.1).

Гипотеза № 3. Процесс разработки функциональных гибридных систем релевантен процессам решений сложных задач в СППР.

Гипотеза № 4. Машинные эксперименты с функциональными ГиИС релевантны процессам выработки методов решения сложных, неоднородных задач в СППР.

5. МОДЕЛИРОВАНИЕ ФУНКЦИОНАЛЬНЫХ ГИБРИДНЫХ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

5.1. Концептуальные модели — инструмент представления гибридных знаний

Взаимодействие субъектов в СППР в процессе решения сложных задач может осуществляться только с использованием механизма обеспечения такого взаимодействия — языка. Вместе с системами формирования и манипулирования образами, запахами и другими язык формирует изученный к настоящему времени в науке инструментарий повседневной и профессиональной деятельности человека. Будучи одновременно средством мысленного эксперимента и моделью внешнего мира, язык профессиональной деятельности — объект анализа и получения информации для гибридизации (см. разд. 4.1, 4.6) при разработке функциональных ГиИС. В этой связи излагаемый ниже подход к разработке ГиИС будем называть лингвистическим.

Практика работы с языками профессиональной деятельности в технических, социально-экономических и био-производственных системах [1], а также практика исследования мира моделирования показывает, что основные средства выражения (обозначения) знаний в них — это аналитические зависимости, эмпирические численные и лингвистические данные в виде таблиц, чертежей, рисунков, схем и текстов, содержащих многочисленные гиперссылки. Анализ языка профессиональной деятельности сталкивается со следующими проблемами: 1) неструктурированностью, т.е. отсутствием явного указания, к какой лексической группе относится то или иное слово, и пресуппозицией, т.е. тем, что смысл текста — не совокупность смыслов, образующих его предложение; 2) несистематизированностью знаний и запутанностью их связей в пределах одного-единственного или множества носителей (длинный контекст); 3) одновременным использованием нескольких выразительных средств разных уровней абстракции, которые друг без друга трудно понимаемы (неоднородность); 4) противоречивостью терминологии (если даже документ содержит определения); 5) многочисленными условными обозначениями, сокращениями, специальной, а не общепринятой лексикой, понимаемой только в узком кругу специалистов.

На языке профессиональной деятельности построен документооборот в системе управления, сформулированы знания, зафиксированные в учебниках, монографиях, методиках, стандартах, инструкциях, рекламных проспектах, каталогах услуг и т.п., что является одним из ценных вербализованных источников информации для гибридизации. Профессиональные языки управления и моделирования — это полязыки, сформировавшиеся в результате совместной деятельности различных профессиональных групп в СППР.

При таком взаимодействии два объекта: оригинал-задача на профессиональном полязыке СППР, а также прототипы-методы на ЯПД модельера и программа, записанная на языке машины Тьюринга, — формируют шкалу «оригинал-программа». На этой шкале могут быть расположены модели, занимающие и промежуточное положение. Один из вариантов такого размещения показан на рис. 5.1. Актуальность «промежуточных» моделей объясняется невозможностью в настоящее время преобразовать информацию, записанную на ЯПД, в информацию, «понятную» ЭВМ. Поэтому современная методология предусматривает последовательное, а зачастую итерационное моделирование на все более и более искусственных языках, свободных от недостатков языков профессиональной деятельности. Один из таких искусственных языков построения моделей, расположенных по шкале «оригинал-программа» ближе к левой границе, — язык концептуальных моделей.

Концептуальная модель (англ. *concept* — понятие, идея; общее представление; концепция) — модель предметной области, состоящая из перечня всех понятий, используемых для описания этой области, вместе со свойствами и характеристиками, классификацией этих понятий по типам, ситуациям, признакам в данной области и законов протекания процессов в ней [233, 234]. Такой смысл близок к термину «концепция» (от лат. *conceptio* — понимание, система), как определенный способ трактовки какого-либо предмета, процесса, явления.

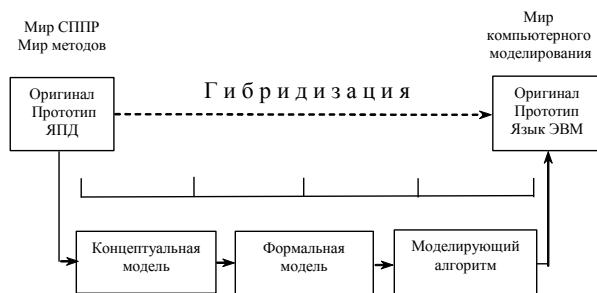


Рис. 5.1 — Место концептуальных моделей на шкале «оригинал-программа»

Таким образом, концептуальная модель носит двоякую сущность. С одной стороны, она ограничивает предметную область, как совокупность реальных или абстрактных объектов (сущностей), связей и отношений между объектами, а также процедур преобразования этих объектов при решении задач. С другой стороны, она предполагает внесение в гибридизацию субъективных представлений разработчика в виде его знаний и опыта — концепций. Это эвристики, специфицирующие понятия ресурса, свойства, действия, структуры, состояния, поведения объекта управления, цели, оценки, критерия, плана, задачи субъекта управления в мирах задач и их решения (рис. 7.3), а также метода, модели, алгоритма, программы в мире моделирования и раскрывающие содержание, логическую связанность отдельных видов деятельности при гибридизации. Отсутствие таких эвристик у разработчика делает гибридизацию нецеленаправленной, вносит в эту сложную деятельность стохастичность. Извлечение, представление, хранение, тиражирование и применение таких эвристик чрезвычайно важны при гибридизации.

Обычно подготовленный разработчик располагает богатым арсеналом подобных концептов и концепций, т.е. владеет языком концептуального моделирования и применяет его, чтобы построить и использовать при гибридизации упрощенное понимание, видение оригинала, прототипа и результата. Естественно-языковые концепции возбуждают и правополушарное мышление, хранящее концепции-образы. При этом возникает интересный эффект. Его можно представить как ситуацию, когда разработчик имеет набор очков (инструментов для коррекции зрения) и последовательно применяет их. Без очков он видит мир, пугающий его воображение многообразием предметов, явлений, процессов. Используя же очки, модельер может наблюдать миры управления и моделирования упрощенно, структурированно.

Такая «коррекция зрения» не возникает на пустом месте. Она — результат накопленных теоретических знаний и практического личностного опыта модельера. Следует также иметь в виду и другое важное обстоятельство. Выбрав и надев такие очки, разработчик так и «ходит» в них до получения результатов. И только анализ результатов может либо подтвердить правильность выбора концепции, либо заставить повторить концептуализацию заново.

Таким образом, выбор и применение концепций языка концептуального моделирования — ответственный этап организации взаимодействия миров управления и моделирования в ходе гибридизации. «Важно отметить, что переход непосредственно к формальным представлениям

в базе знаний без этапа концептуального описания в поле знаний приводит к многочисленным ошибкам, что замедляет процесс формирования базы знаний интеллектуальной системы» [235, с. 66].

Мир концептуальных моделей сегодня настолько разнообразен, что стал предметом специализированных международных конференций и объектом исследований многих ученых [236 — 238].

Представление информации для решения интеллектуальных проблем — важная и трудная задача. Ранние исследования в искусственном интеллекте использовали в качестве таких представлений семантические сети (англ. semantic networks), концептуальные зависимости (англ. conceptual dependencies), сценарии (англ. scripts) и фреймы (англ. frames). К современным представлениям следует отнести концептуальные графы (англ. conceptual graphs) для решения задач понимания естественного языка, категориальные архитектуры (англ. subsumption architectures) и др.

5.2. Концептуальный базис функциональных гибридных интеллектуальных систем

Развитие науки свидетельствует со всей очевидностью, что, скорее всего, не удастся построить единственную, законченную и стройную картину окружающего мира. Это происходит не только в силу субъективности представлений о структурированности «своего мира» у участников СППР, но и в силу субъективных представлений о структурированности «своего мира» у разработчика. Следуя Д.А. Поспелову [4], авторами предложен и развит класс ролевых концептуальных моделей, а, следуя А.И. Уемову [239, 240], триада «вещь — свойство — отношение» положена в основу структурирования языков профессиональной деятельности экспертов и ЛПР, языка профессиональной деятельности разработчика, а также неформальной аксиоматической теории концептуального моделирования при проектировании ГиИС для многомодельных, гибридных и гибридных адаптивных систем управления [1].

Концептуализация основывается на описании предметной области в виде неформальной (содержательной) аксиоматической теории. Приписывание значений первичным понятиям теории называется интерпретацией теории. Если эти значения удовлетворяют аксиомам теории, они называются ее моделью. Перенос теорем аксиоматической теории на интерпретируемую предметную позволяет развивать и обогащать конкретные теории. Аксиоматический метод предполагает такое построение научной теории, при котором в ее основу кладутся некоторые исходные суждения, аксиомы, из которых все другие утверждения этой науки (теоремы) должны выводиться чисто логическим путем, посред-

ством доказательства. Все понятия теории вводятся путем определений. Ограниченность аксиоматического подхода показана в 1931 г. Гёделем, доказавшем, что всякая достаточно богатая непротиворечивая, формальная система неполна. «Большинство аксиоматических теорий являются неформальными в том смысле, что они предполагают использование результатов общей теории множеств и следуют правилам вывода в логике. Совершенно формальные теории не признают никаких вне их полученных предпосылок: правило вывода должно быть сформулировано как теорема формальной теории прежде, чем им начнут пользоваться при доказательстве. Неформальная теория неформальна только по сравнению с формальной, в остальном неформальная теория вполне формализована [241]. При описании предметных областей вначале строятся неформальные (содержательные) аксиоматические теории. Для развитых неформальных аксиоматических теорий можно построить формальную аксиоматическую теорию [236].

Неформальная аксиоматическая теория Te включает в себя четыре списка [236]:

$$Te = \langle Te^{TH}, Te^{TO}, Te^A, Te^{BT} \rangle, \quad (5.1)$$

где Te^{TH} , Te^{TO} — неопределяемые и определяемые термины соответственно; Te^A — аксиомы; Te^{BT} — высказывания-теоремы, выводимые из Te^A по некоторым фиксированным логическим правилам.

Упорядоченная пара $\langle Te^{TH}, Te^A \rangle$ называется формулировкой теории. Вместе с T^{BT} эта пара определяет предметную область теории. В концептуальном моделировании понятийная структура предметной области описывается с помощью неформальных аксиоматических теорий, формулировки которых и называются концептуальными моделями.

Для построения теории (5.1) зададим, прежде всего Te^{TH} , Te^{TO} , т.е. ее категориальное ядро.

Описание категориального ядра теории. Определяя задачи общей теории систем как этапа развития системного подхода, В.Н. Сагатовский пишет: «Общая теория систем, на наш взгляд, должна строиться как теория среднего уровня, выступающая опосредствующим звеном между философией как общей стратегией деятельности и областями предметного знания. В ней еще предстоит синтезировать категориальные основы, разрабатываемые на философском уровне, с формальным аппаратом» [240, с. 23, 24]. Один из результатов в теории систем и системном анализе получен А.И. Уемовым [239], предложившим в качестве базиса системного подхода к анализу и синтезу сложных объектов

триаду категорий «вещь — свойство — отношение». В своих работах А.И. Уемов для раскрытия структуры окружающего нас мира главную роль отводит категориям вещи, свойства и отношения. Неправильное понимание этих категорий участниками СППР и разработчиками приводит не только к неправильному применению основного понятия системного подхода «система» и последующим ошибкам в ходе гибридизации, но и к развитию у разработчика «примитивного» мышления. Термин «примитивное мышление» рассмотрен в [1].

Понятие «структура» окружающего нас мира требует более тщательного философского, лингвистического и системотехнического осмысления. Для этого необходимо определить основной философский базис теории, методологии и технологии функциональных гибридных интеллектуальных систем.

Главная особенность триады «вещь — свойство — отношение» состоит в соотносительности, т.е. в том, что все эти категории определяются друг через друга, причем центральная, основная категория — категория вещи [239]. Из соотносительности следует, что в категориальном ядре не действует правило о запрещении круга в определениях и, что в неформальной аксиоматической теории (5.1) категории $Te^{TH} = Te^{TO} = \{\text{«вещь»}, \text{«свойство»}, \text{«отношение»}\}$.

Понимание вещи. Вещь, как часть материи, имеет границы. Особенно важно то, что изменение качественной границы в эволюционных, биологических, производственных и других процессах превращает одну вещь в другую, независимо от сохранения или изменения ее пространственных характеристик.

Понимая вещь качественно, будем предполагать следующее.

Определение 1. 1) вещь — это система качеств; 2) различные вещи — это различные системы качеств; 3) одна и та же вещь — это одна и та же система качеств; 4) все вещи существуют в пространстве и времени.

Человек образует в своем сознании единство признаков на основе объективно существующего в окружающем мире единства качеств. И только после этого он как-то называет это единство.

Результаты отражения материальных вещей представляют собой восприятия, представления и понятия, т.е. системы признаков — отображений объективно существующих качеств. Качественное толкование вещи полностью соответствует пониманию предмета в логике и лингвистике. Оно едино в онтологическом, логическом и грамматическом планах.

Понимание свойства. Анализ литературы, словарей показывает, что в определениях и толкованиях понятий качества и свойства много противоречий. Наличие этих противоречий негативно сказывается на рабо-

те экспертов и проектировщиков. Будем использовать в дальнейших рассуждениях следующие определения.

Качество — существенное свойство вещи. Качество — граница вещи. С исчезновением качества одна вещь превращается в другую.

Одно и то же качество может быть присуще разным вещам. Одна и та вещь может обладать множеством различных качеств, каждое из которых отделяет ее от других вещей. Качество одной вещи — свойство для другой.

Определение 2. Свойство — все то, что не является границами данной вещи. Это то, что, характеризуя вещи, не образует новых вещей.

Понимание отношения. Отношения более сложны и абстрактны, чем вещи и свойства. Последние можно воспринимать чувственно. Отношения же чувственно не воспринимаются.

Каждая вещь состоит из элементов — других вещей или свойства. Из одних и тех элементов могут быть составлены разные вещи. У разных вещей — разные способы их образования из этих элементов. Иными словами, различны взаимоотношения между элементами. Взаимодействие свойств дает систему свойств, т.е. вещь. Взаимодействие вещей также дает вещь. Следовательно, отношение можно определить через понятие вещи.

Определение 3. Отношение — то, что образует вещь из данных элементов (свойства или других вещей). Отношение — есть то, что, будучи установлено между вещами, образует новые вещи.

Взаимоотношения вещей, свойств и отношений. Между категориями вещи, свойства и отношения существует зависимость, выраженная «треугольником» А.И. Умова [239]. Все три категории — соотносительные и определяются друг через друга, причем центральная категория — вещь. Через нее непосредственно специфицируются свойства и отношения.

Отношение можно рассматривать как особый случай свойства. Рассматривая вещи, между которыми существует данное отношение, как систему, т.е. как одну вещь, можно сказать, что отношение соотносится с классом вещей. Таким образом, отношение полностью удовлетворяет определению свойства и может считаться частным случаем свойства. Что же отличает отношения от других свойств? То, что вещь, которую характеризует отношение, рассматривается не в своей целостности, а в своем многообразии, т.е. как совокупность других вещей — своих элементов. Факт возможности рассмотрения отношения, как частного случая свойства, находит отражение и в языке профессиональной деятельности. Свойства предметов обозначаются прилагательными. Однако отношения также могут быть выражены прилагательными, например, «равные вещи», «одновременные действия».

Свойство можно рассматривать как вещь. Свойство не есть что-то элементарное, не разложимое на части, и его можно рассматривать как систему других свойств. Например, в свойство «объем» элементами входят «высота» и «площадь». В свою очередь, «площадь» определяется через «длину» и «ширину». В языке профессиональной деятельности отражен подход к свойству как к вещи: когда называется то или иное свойство, то применяется слово, обозначающее вещь, т.е. существительное. Например, «производительность», «тип», «грузоподъемность», «мощность» и т.п.

В практике мышления человека свойство, взятое само по себе, рассматривается как вещь. Оно является особым, частным случаем вещи.

Отношение можно рассматривать как вещь. Подобно свойству отношение разложимо на совокупность отношений, т.е. оно есть система отношений. Например, отношение равенства — система из двух отношений «не меньше» и «не больше». Отношение «быть рядом» — система отношений «быть» и «рядом». Поскольку отношение можно рассматривать как свойство, то и система отношений — есть система свойств. Это значит, что отношение, как система других отношений, будет по определению частным случаем вещи и должно характеризоваться свойствами. Это действительно имеет место. Можно указать такие свойства отношения, как рефлексивность, симметричность, транзитивность, валентность и т.д. Можно дать также не формальные признаки отношения, а содержательные — пространственные, временные, причинно-следственные и другие отношения, что очень важно при вычленении в языке групп отношений, несущих определенную функциональную нагрузку.

Подобно свойствам, отношения в языке профессиональной деятельности также выражаются словами, обозначающими вещи, т.е. существительными. Например, «равенство», «сходство», «погрузка» и др.

Вещь можно рассматривать как свойство. Тот факт, что вещь может быть свойством другой вещи, часто проявляется в языке профессиональной деятельности. Рассмотрим два выражения «*судно типа буксир*» и «*буксир быть судном*». В первом выражении *судно* — вещь, а *буксир* — свойство. Во втором выражении уже буксир играет роль вещи, а судно — свойства. Кроме этого, вещи очень часто в языке обозначаются словами, выражающими свойства, т.е. прилагательными, например, *распределительный, гидравлический, красный* и т.д. Если слово *распределение* выражает свойство как вещь, то *распределительный* и другие одноклассные слова выражают вещь как свойство.

Свойство можно рассматривать как отношение. Одна и та же вещь, взятая в разных отношениях, обладает различными, иногда противоположными свойствами. Например, использование пестицидов — полез-

но, так как повышает урожайность, и вредно — так как ущербно для экологии и, в конечном итоге, — для человека.

Такая относительность свойств может быть понятна лишь в том случае, если рассматривать само свойство как отношение. Например, *низкие обороты* силовой установки судна, характеризуют не только сам двигатель, но и другие элементы движительной системы: вал, редуктор, винт, с которыми взаимодействует двигатель. Все эти элементы, благодаря данному свойству, образуют систему. Таким образом, свойство объединяет вещи в систему, т.е. удовлетворяет определению отношения. Чем же свойство отличается от других отношений? Отношение можно рассматривать, абстрагируясь от тех вещей, между которыми оно существует. Свойство же вещи — это отношение, существующее в той вещи, в которую включается данная вещь.

Вещь можно рассматривать как отношение. Каждая вещь организует системы из других вещей. Например, судно представляет собой сложное отношение: комбинацию внутренних отношений и отношений к другим вещам — океану, атмосфере, различного рода полям и т.д. Изменяя отношения вещи к другим вещам, в которых она существует, изменяют и саму вещь, превращая ее в другую. Меняя отношения, можно переходить от одних типов судов к другим, например, от траулеров к рефрижераторам.

Отличие вещей от других отношений заключается в том, что эти отношения, в отличие от отношений-свойств, сложны и охватывают бесчисленное множество других вещей.

Преобразование категориального ядра применительно к теории предметной области. Непосредственное применение рассмотренного выше категориального ядра S^{TH} к системному анализу предметной области затруднительно в силу нескольких причин. Во-первых, категория вещи — верхний предел абстракции, что вызывает трудности с ее интерпретацией участниками СППР и разработчиком. Во-вторых, имеются трудности с интерпретацией понятия «зависимость» между категориями ядра. Поскольку зависимость между ними — это отношение, то возникает неоднозначность в толковании отношений. Если использовать язык дискретной математики, то они обозначаются вершиной графа-треугольника и его ребрами одновременно. В-третьих, категория «отношение» носит общий характер и целесообразно выбрать категорию, которая, являясь отношением, была бы в то же время предметно-ориентированной.

Трудности преодолеем следующим образом. Из множества вещей будем рассматривать только вещи-ресурсы (в дальнейшем просто ресурсы), подчеркивая тем самым, что прагматичны не вещи сами по себе,

а только вещи, имеющиеся в распоряжении ЛПР для решения задач. Будем также считать, что природные явления также ресурс, вне зависимости от положительности или отрицательности эффекта их воздействия.

Из множества отношений выберем отношения-действия (далее — просто действия), подчеркивая тем самым интерес к отношениям на ресурсах, установление или разрыв которых может изменить состояние объекта управления, т.е. операциям и процессам, в частности технологическим и производственным.

Таким образом, возможны определения 4 и 5.

Определение 4. Ресурс — понятие, обозначающее вещь, имеющуюся у субъекта управления для решения задач.

Например, *главный двигатель, дизель-генератор, кран, бригада, судно, почва, семена* и др.

Определение 5. Действие — понятие, обозначающее отношения на ресурсах, как следствие деятельности, поступков и поведения.

Например, *перекачка топлива, выгрузка судна, боронование участка поля* и др.

После выполненных преобразований появляется возможность уточнить и понятие «зависимость» между категориями ресурса, свойства и действия, трактуя ее как отношение в широком смысле этого слова.

Выполненные преобразования изменяют вид треугольника А.И. Умова и изображены на рис. 5.2 в виде графа G .

Концептуальная модель на рис. 5.2 дает достаточно простое и прозрачное структурирование мира СППР, утверждая, что это есть мир ресурсов, действий их свойств и девяти классов отношений «ресурс-ресурс», «действие-действие», «свойство-свойство», «ресурс-свойство» и «свойство-ресурс», «ресурс-действие» и «действие-ресурс», а также «действие-свойство» и «свойство-действие», которые ниже обозначены как r , а их множества как R .

как r , а их множества как R .

Главная особенность этой модели — ее ролевой характер, который, по сути, и сформулирован при перечислении девяти классов отношений. Чтобы понять эту особенность, достаточно рассмотреть любой из девяти классов отношений. Например, что обозначает выражение «ресурс-ресурс»? Это неявно выраженное отношение с двумя ролями, причем обе роли играют ресурсы. В выражении «ресурс-действие» одну роль займет ресурс, а другую — действие. Ро-

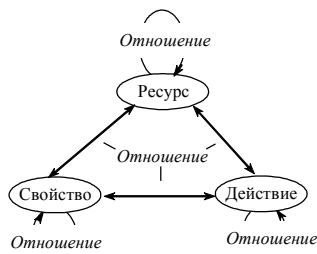


Рис. 5.2 — Категориальное ядро «ресурс – свойство – действие – отношение»

242

левой характер придает модели универсальность и делает ее независимой от специфики предметной области.

Введем множество $X^0 = \{^1X^0, ^2X^0, ^3X^0\}$, где $^1X^0, ^2X^0, ^3X^0$ — множества базисных понятий, обозначающих ресурсы, свойства и действия, соответственно, и построим на X^0 полный граф $G = (X^0, R^0)$. Тогда имеем классификацию базовых отношений $R^0 = \{^{11}R^0, ^{22}R^0, ^{33}R^0, ^{12}R^0, ^{21}R^0, ^{13}R^0, ^{31}R^0, ^{23}R^0, ^{32}R^0, ^{23}R^0\}$, где $^{11}R^0, ^{22}R^0, ^{33}R^0, ^{12}R^0, ^{21}R^0, ^{13}R^0, ^{31}R^0, ^{32}R^0, ^{23}R^0$ — множества отношений «ресурс-ресурс», «свойство-свойство», «действие-действие», «ресурс-свойство», «свойство-ресурс», «ресурс-действие», «действие-ресурс», «действие-свойство», «свойство-действие» соответственно.

Таким образом, после того как определен граф G , имеем $Te^{TH} = \{\langle \text{«вещь»}, \langle \text{«свойство»}, \langle \text{«отношение»}\rangle\}$, $Te^{TO} = \{\langle \text{«вещь»}, \langle \text{«свойство»}, \langle \text{«отношение»}, \langle \text{«ресурс»}, \langle \text{«действие»}\rangle\}$ и можно ввести аксиоматику теории (5.1).

Аксиоматика ролевых конструктов. Введем три аксиомы.

Аксиома 1 (аксиома существования одноролевого конструкта). Если ${}^\alpha X^0 \subseteq X^0$, а ${}^{\alpha\alpha}R^0 \subseteq R^0$, то существует одноролевой конструкт co^1 ,

которым назовем выражение $co^1 = \langle {}^\alpha X^0 \ {}^{\alpha\alpha}R^0 \ {}^\alpha X^0 \rangle$, где $\alpha = 1, 2, 3$.

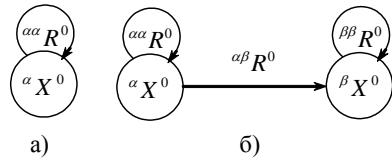


Рис. 5.3 — Пример одноролевого конструкта — а) и двухролевого — б)

Графическая интерпретация co^1 показана на рис. 5.3,а. Одноролевой конструкт имеет в качестве ролей одну и ту же категорию. Одноролевых конструктов по определению — три, столько, сколько петель на графе G . Они могут быть записаны или изображены, если переменная α будет принимать значения на множестве $\{1, 2, 3\}$. Одноролевой конструкт — это точечная диаграмма. Например, в языке профессиональной деятельности диспетчера морского порта можно выделить следующие одноролевые конструкты: *склад рядом причал*, *камера часть холодильника*, т.е. в которых присутствует только одна из базисных категорий.

Как видно из примеров, семантическая сила одноролевых конструктов невелика. Они выражают смысл на абстрактном уровне.

Аксиома 2 (аксиома существования двухролевого конструкта). Если $\alpha X^0, \beta X^0 \subseteq X^0$, а $\alpha\alpha R^0, \beta\beta R^0, \alpha\beta R^0 \subseteq R^0$, то существует двухролевой конструкт co^2 , которым назовем выражение $co^2 = \alpha X^0 \alpha\alpha R^0 \alpha X^0 \circ \beta X^0 \beta\beta R^0 \beta X^0 \circ \alpha X^0 \alpha\beta R^0 \beta X^0$, где $\alpha, \beta = 1, 2, 3$; $\alpha \neq \beta$; знак \circ обозначает конкатенацию.

Графическая интерпретация co^2 изображена на рис. 5.3,б. Двухролевой конструкт включает два одноролевых конструкта, связанных отношениями $\alpha\beta R^0$. Двухролевых конструктов по определению — шесть, по количеству ребер на графе G . Переход от одноролевой модели к двухролевой означает переход от точечной к линейной диаграмме.

Например, в языке профессиональной деятельности диспетчера морского порта можно выделить следующие двухролевые конструкты: *судно Агат стоит у причала № 16, ускоренная выгрузка используется на судне Топаз*, т.е. в которых есть только две из трех базисных категорий.

Применение двухролевого конструкта, несмотря на его кажущуюся теоретическую и практическую завершенность, в определенной степени тормозит развитие мира моделирования. Линейная диаграмма, какой бы длинной она не была, не обеспечивает целостную картину внешнего мира.

Аксиома 3 (аксиома существования трехролевого конструкта). Если $\alpha X^0, \beta X^0, \gamma X^0 \subseteq X^0$, а $\alpha\alpha R^0, \beta\beta R^0, \gamma\gamma R^0, \beta\alpha R^0, \gamma\beta R^0 \subseteq R^0$, то существует трехролевой конструкт co^3 , которым назовем выражение $co^3 = \alpha X^0 \alpha\alpha R^0 \alpha X^0 \circ \beta X^0 \beta\beta R^0 \beta X^0 \circ \gamma X^0 \gamma\gamma R^0 \gamma X^0 \circ \alpha X^0 \alpha\gamma R^0 \gamma X^0 \circ$

$\beta X^0 \beta\alpha R^0 \alpha X^0 \circ \gamma X^0 \gamma\beta R^0 \beta X^0$, где $\alpha, \beta, \gamma = 1, 2, 3$; $\alpha \neq \beta \neq \gamma$.

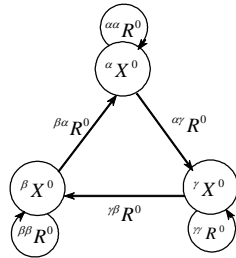


Рис. 5.4 — Пример трехролевого конструкта

Графическая интерпретация co^3 изображена на рис. 5.4. Трехролевых конструктов по определению — девять, т.е. по числу вариантов различной ориентации дуг на графе G . Переход от двухролевой модели к трехролевой означает переход от линейной разомкнутой диаграммы к фигурной (треугольной) замкнутой диаграмме, концептуально полной для триады «ресурс — свойство — действие»

картине мира. Например, в языке профессиональной деятельности диспетчера можно выделить следующий трехролевой конструкт: *судно стоит у причала №15 и на нем выполняется ускоренная выгрузка мо-*

роженой рыбы. В выражении одновременно присутствуют ресурсы, свойства, действия со связывающими их отношениями.

В итоге, имеем $Te^A = \{A1, A2, A3\}$ и три множества конструкторов $CO^1 = \{co_1^1, \dots, co_3^1\}$, $CO^2 = \{co_1^2, \dots, co_6^2\}$, $CO^3 = \{co_1^3, \dots, co_6^3\}$. На категориальном ядре, т.е. множестве X^0 , невозможно задать никаких других конструкторов.

Расширение категориального ядра. Для моделирования предметных знаний СППР об объекте управления категориальное ядро расширим ${}^{21}X^0 \subseteq {}^2X^0$ — физическими свойствами (параметрами), ${}^{22}X^0 \subseteq {}^2X^0$ — характеристическими свойствами (характеристиками), ${}^{23}X^0 \subseteq {}^2X^0$ — именными свойствами (именами), ${}^4X^0$ — единицами измерения (мерами), ${}^5X^0$ — значениями и ${}^6X^0$ — состояниями. Для моделирования предметных знаний о субъекте управления в X^0 включены ${}^7X^0$ — оценки и ${}^8X^0 = \{x_1^0$ — «задача», а о субъекте моделирования — ${}^9X^0 = \{x_1^0$ — «метод», x_2^0 — «модель», x_3^0 — «программа»}. Чтобы иметь возможность расширять категориальное ядро, введем и ${}^{10}X^0$ — экзотические понятия.

Определение 6. Параметр — свойство, имеющее меру, т.е. значение которого может быть измерено по некоторой шкале для данного свойства. Например, *температура* жидкости в трубопроводе, *частота* и *напряжение* электрического тока, *глубина* вспашки, *осадка* судна.

Определение 7. Характеристика — свойство, выражающее строение, устройство, особенности. Например, *знак автоматизации* в символе класса судна, *тип* насоса, *проект* судна, *тип* почвы.

Определение 8. Имя — свойство, идентифицирующее ресурсы, действия, задачи, методы и выделяющее их среди других с одинаковыми параметрами и характеристиками. Например, *серийный номер* двигателя, *номер датчика*, *бортовой номер* судна, *номер участка поля*, *наименование* сорта культуры.

Определение 9. Мера — понятие, обозначающее принятую единицу параметра свойства и используемую в качестве меры последнего. Например, *обороты/в минуту*, *вольт*, *тонна/час*, *метр* и др.

Определение 10. Значение — понятие или число, показывающее количество единиц меры. Например, длина причала *сто* метров, скорость потока жидкости в трубопроводе — *3 м/сек*, *высокая* влажность почвы.

Определение 11. Состояние — понятие, обозначающее внешнее проявление процессов, происходящих в ресурсе в некоторый момент времени. Например, *включенный прибор, вспаханное поле, занятый причал*.

Определение 12. Оценка — понятие, отображающее результат выполнения действий на оценочные шкалы. Установление меры близости текущего состояния ресурсов к желаемому. Например, *удачное включение прибора, низкие холостые обороты двигателя, хорошо вспаханное поле, частично занятый причал*.

Определение 13. «Задача» — понятие, обозначающее то, что требует размышления, и содержащее исходные данные, неизвестное и условия, конкретизирующие связи неизвестного с данными.

Определение 14. «Метод» — понятие, обозначающее совокупность того, что направлено на теоретическое или практическое освоение действительности и подчинено решению конкретной задачи.

Определение 15. «Модель» — понятие, обозначающее образ или прообраз оригинала, используемый при определенных условиях в качестве его «заместителя», «представителя».

Определение 16. Экзотическое понятие — понятие, не принадлежащее ни одному из нижеперечисленных классов ${}^1X^0, \dots, {}^9X^0, {}^{21}X^0, {}^{22}X^0, {}^{23}X^0$.

Рассмотренное расширение категориального ядра позволяет построить и другие конструкты. Их примеры изображены на рис. 5.5.

Правило склеивания ролевых конструктов. Чтобы перейти к построению множества высказываний-теорем Te^{BT} теории (5.1), сформулируем на X^0 правило склеивания следующим образом: 1) трехролевой конструкт $co^3 \in CO^3$ — есть схема $ch \in CH$ ролевых концептуальных моделей; 2) если $ch = A_1 \circ {}^\alpha X^0 \circ {}^{\alpha\alpha} R^0 \circ {}^\alpha X^0 \circ B_1$ — схема ролевых концептуальных моделей, где A_1, B_1 — любые фрагменты (в том числе и пустые) модели, не содержащие ${}^\alpha X^0 \circ {}^{\alpha\alpha} R^0 \circ {}^\alpha X^0$, исключая случай $A_1 = B_1 = \emptyset$ и $co = A_2 \circ {}^\alpha X^0 \circ {}^{\alpha\alpha} R^0 \circ {}^\alpha X^0 \circ B_2$ — двух или трехролевой конструкт, где A_2, B_2 — любые фрагменты (в том числе и пустые), не содержащие ${}^\alpha X^0 \circ {}^{\alpha\alpha} R^0 \circ {}^\alpha X^0$, исключая случай $A_2 = B_2 = \emptyset$, то $ch = A_1 \circ {}^\alpha X^0 \circ {}^{\alpha\alpha} R^0 \circ {}^\alpha X^0 \circ B_1 \circ A_2 \circ B_2$ — также схема ролевых концептуальных моделей (склеивание треугольников вершиной); 3) если $ch_1 = A_1 \circ {}^\alpha X^0 \circ {}^{\alpha\beta} R^0 \circ {}^\beta X^0 \circ B_1$, $ch_2 = A_2 \circ {}^\beta X^0 \circ {}^{\beta\alpha} R^0 \circ {}^\alpha X^0 \circ B_2$ — схемы ролевых концептуальных моделей, причем $A_1 = A_2$, $B_1 = B_2$ — непустые

фрагменты, то $ch_3 = A_1 \circ^{\alpha} X^0 \circ^{\beta} R^0 \circ^{\beta} X^0 \circ^{\beta} R^0 \circ^{\alpha} X^0 \circ B_1$ — также схема ролевых концептуальных моделей (добавление обратного отношения); 4) если $ch_1 = A_1 \circ^{\alpha} X^0 \circ^{\beta} R^0 \circ^{\beta} X^0 \circ B_1$, $ch_2 = A_2 \circ^{\alpha} X^0 \circ^{\beta} R^0 \circ^{\beta} X^0 \circ B_2$ — схемы ролевых концептуальных моделей, причем A_1, A_2, B_1, B_2 — непустые фрагменты, то $ch_3 = A_1 \circ A_2 \circ^{\alpha} X^0 \circ^{\beta} R^0 \circ^{\beta} X^0 \circ B_1 \circ B_2$ — также схема ролевых концептуальных моделей (склеивание треугольников стороной); 5) в схеме ролевых концептуальных моделей не может быть более одного не связанного одноролевого конструкта; 6) никаких других схем ролевых концептуальных моделей нет.

Таким образом, схемы ролевых концептуальных моделей $ch \in CH$ строятся из конструктов-аксиом, и есть высказывания-теоремы неформальной аксиоматической теории (5.1), полученные по правилу склеивания.

Применение аксиоматической теории ролевых концептуальных моделей в теории, методологии и технологии функциональных гибридных интеллектуальных систем. Чтобы преобразовать схему ролевой концептуальной модели в концептуальную ролевую модель, необходимо наполнить классы понятий и отношений из базового и расширенного ядра теории Te (5.1) лексемами и специфицировать элементы множеств X, R , использованных в схеме, т.е. выполнить интерпретацию теории Te (5.1).

Схемы ролевых концептуальных моделей — эвристики, которые могут использоваться в теории, методологии и технологии функциональных ГиИС с разными целями: 1) для построения в памяти компьютера семантической структуры текстов [98] на языках профессиональной деятельности экспертов, ЛПР и разработчиков или модели действительности [242] и автоматизированного построения концептуальных моделей, т.е. преобразования схемы ролевых концептуальных моделей в ролевые концептуальные модели; 2) для структурированного представления знаний об объектах-оригиналах, т.е. сложных, неоднородных задачах; 3) для структурированного представления знаний об объектах-прототипах: методах и моделях; 4) для структурированного представления знаний об объектах-результатах: элементах

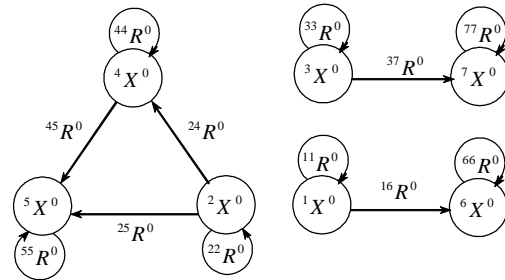


Рис. 5.5 — Примеры ролевых конструктов расширенного категориального ядра

ГиИС, а также ГиИС в целом; 5) для структурированного представления знаний о содержании и логической последовательности преобразований объектов-оригиналов и объектов-прототипов в объекты-результаты; 6) для описания системы операций, разработки информационной и инструментальной сред технологии функциональных ГиИС.

Ниже рассмотрены несколько примеров применения схем ролевых концептуальных моделей из рассмотренного выше перечня.

Схема ролевых концептуальных моделей для представления моделей действительности. «Семантическая структура текста образуется в сознании познающего субъекта с помощью знаний о языке, знаний о мире, а также общих знаний в той предметной области, которой посвящен текст. Тексты пишутся для посвященных» [98, с. 127]. В полной мере сказанное относится и к текстам на ЯПД — источнику информации об объектах-оригиналах и объектах-прототипах. Для извлечения такой информации, структурирования и использования при гибридизации применяются модели действительности.

Задача структурирования формулируется следующим образом: дан текст на языке профессиональной деятельности и необходимо классифицировать его лексемы по семантическим группам, например, соответствующим наименованиям множеств базисных понятий X^0 (определения 1 — 16), а также присвоить лексемам метки классов. Пусть имеем предложение: «судно Агат стоит у причала № 15». Тогда необходимо преобразовать его к виду: «судно (ресурс) Агат (имя) стоит у (отношение «ресурс-ресурс») причал (ресурс) № (имя) 15 (значение)», где в скобках указаны метки классов. Такое преобразование обеспечивает последующую интеллектуальную обработку: построение ролевых концептуальных моделей по схемам, наполнение словарей понятиями и отношениями, извлечение знаний, пополнение текста знаниями и др.

Проблему решают, распределяя функции анализа, принятия решения об отнесении лексемы в тот или иной класс между человеком и компьютером. Однако в любом случае [242] априори формулируется и вводится в компьютер модель действительности, которая может быть построена с использованием схем ролевых концептуальных моделей. На рис. 5.6 рассмотрена одна из возможных в теории (5.1) схем ролевых концептуальных моделей ch_1 (5.2).

Рассмотрим схему ch_1 подробнее. Она построена с использованием одно-, двух- и трехролевых конструктов на семи основных категориях: ресурса, свойства, действия, меры, значения, состояния и оценки. В схеме применены и три вспомогательных категории: параметра, характеристики, имени и ${}^8X^0$ — экзотических понятий. Последняя введена в схему, чтобы сделать модель полной и расширяемой методом логиче-

ского замыкания [23]. Кроме этого, схема выделяет и позволяет использовать при гибридизации до 24 классов отношений (не учтены отношения $^{88}R^0$, не играющие сколь-нибудь заметной роли до момента логического замыкания множества X^0).

$$\begin{aligned}
 ch_1 = & \ ^1X^0 \ ^{11}R^0 \ ^1X^0 \circ \ ^2X^0 \ ^{22}R^0 \ ^2X^0 \circ \ ^3X^0 \ ^{33}R^0 \ ^3X^0 \circ \ ^4X^0 \ ^{44}R^0 \ ^4X^0 \circ \\
 & \circ \ ^5X^0 \ ^{55}R^0 \ ^5X^0 \circ \ ^6X^0 \ ^{66}R^0 \ ^6X^0 \circ \ ^7X^0 \ ^{77}R^0 \ ^7X^0 \circ \ ^8X^0 \ ^{88}R^0 \ ^8X^0 \circ \\
 & \circ \ ^{21}X^0 \ ^{2121}R^0 \ ^{21}X^0 \circ \ ^{22}X^0 \ ^{2222}R^0 \ ^{22}X^0 \circ \ ^{23}X^0 \ ^{2323}R^0 \ ^{23}X^0 \circ \\
 & \circ \ ^{21}X^0 \ ^{212}R^0 \ ^2X^0 \circ \ ^{22}X^0 \ ^{222}R^0 \ ^2X^0 \circ \ ^{23}X^0 \ ^{232}R^0 \ ^2X^0 \circ \ ^1X^0 \ ^{12}R^0 \ ^2X^0 \circ \\
 & \circ \ ^2X^0 \ ^{21}R^0 \ ^1X^0 \circ \ ^1X^0 \ ^{13}R^0 \ ^3X^0 \circ \ ^3X^0 \ ^{31}R^0 \ ^1X^0 \circ \ ^2X^0 \ ^{23}R^0 \ ^3X^0 \circ \\
 & \circ \ ^3X^0 \ ^{32}R^0 \ ^1X^0 \circ \ ^1X^0 \ ^{16}R^0 \ ^6X^0 \circ \ ^6X^0 \ ^{61}R^0 \ ^1X^0 \circ \ ^3X^0 \ ^{37}R^0 \ ^7X^0 \circ \\
 & \circ \ ^7X^0 \ ^{73}R^0 \ ^3X^0 \circ \ ^2X^0 \ ^{24}R^0 \ ^4X^0 \circ \ ^4X^0 \ ^{45}R^0 \ ^5X^0 \circ \ ^2X^0 \ ^{25}R^0 \ ^5X^0 \circ \\
 & \circ \ ^5X^0 \ ^{52}R^0 \ ^2X^0 \circ.
 \end{aligned} \tag{5.2}$$

Этому же классу принадлежат отношения, определяющие одни свойства через другие (*длина, ширина и высота определяют объем*).

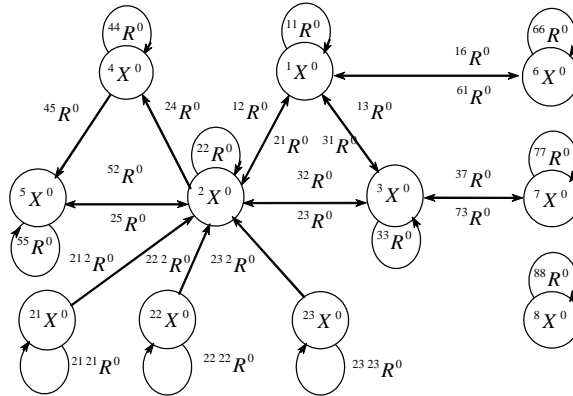


Рис. 5.6 — Схема ролевых концептуальных моделей действительности

Отношения $^{11}R^0$ применяются при гибридизации для моделирования классификации (например, *класс-подкласс*), включения (*целое-часть*)), принадлежности и пространственного положения (*рядом, на*) ресурсов. Отношения $^{33}R^0$ занимают центральное место при моделировании действий — процессов и операций, задавая их классификацию,

включение, подчиненность, временное положение (*раньше, одновременно*), причинно-следственные и каузальные связи. Отношения $^{22}R^0$ используются при гибридизации для сравнения свойств ресурсов и действий (*больше, равно, тождественно*).

Отношения $^{44}R^0$ играют важную роль для тождественных преобразований моделей, устанавливая соответствие шкал для единиц измерения (производительности *тонна/час* — $m^3/смена$, расстояния *метр* — *километр*). Отношения $^{55}R^0$ — задают порядок на значениях параметров и позволяют различать нечеткие значения свойств, выраженные качественно (*глубина* — *большая, средняя*). Отношения $^{66}R^0$ отображают в языке профессиональной деятельности смену состояний ресурсов, выражаются причинно-следственными отношениями для моделирования поведения ресурсов. Отношения $^{77}R^0$ — это предпочтения на оценочных шкалах, часто используемые в теории принятия решений, отношения сравнения оценок, преобразования одних оценок в другие, существенные для анализа результатов действий.

Отношения $^{12}R^0$ специфицируют ресурсы через свойства. Отношения $^{31}R^0$ определяют действия на ресурсах (*иметь объектом, субъектом, местом, средством* и др.). Несколько фундаментальных понятий искусственного интеллекта — «сценарий», «фрейм» могут быть определены на отношениях $^{31}R^0$. Отношения $^{32}R^0$ играют в языках профессиональной деятельности не менее заметное значение — они задают свойства действий, определяя их классификационные и видовые особенности, например, *скоростная погрузка, глубокая вспашка, дистанционное управление*. Отношения $^{24}R^0$, $^{45}R^0$, $^{25}R^0$ связывают свойства, меры и значения и хорошо изучены в теории измерений, теории множеств и теории нечетких множеств. Отношения $^{52}R^0$ были обнаружены в языке профессиональной деятельности конструктора систем судовой автоматики. Эти отношения присутствуют неявно, например, в выражениях *максимальное значение давления, предельное значение температуры* и других конструкциях вида «значение имеет свойство».

Отношения $^{21}R^0$, $^{22}R^0$, $^{23}R^0$ — это отношения *вид-род* или *под-класс-класс*. Отношения $^{16}R^0$ связывают состояние с его носителем — ресурсом, например, *аварийный двигатель, неисправный датчик* и другие, а отношения $^{36}R^0$ — это выполняемые субъектом действия по изменению состояния ресурсов.

В работах [4, 243] установлено соответствие данной оригинальной и традиционных классификаций.

Схема $ch_1 \in CH$, изображенная на рис. 5.6, специфицирует семантическую структуру текстов на языках профессиональной деятельности и используется при гибридизации для построения концептуальных моделей. Метод ее применения состоит в следующем. Компьютер выделяет из текста предложение за предложением и предъявляет по очереди слова из этого предложения, а инженер по знаниям, видя (помня) схему ролевых концептуальных моделей, например, изображенную на рис. 5.6, дает оценки этого слова по семимерной (${}^1X^0, \dots, {}^7X^0$) модели действительности. Если принятие решения затруднено, то лексема относится к экзотическим. После установления пороговых значений оценок автоматически наполняются словари. Класс ${}^8X^0$ используется для повторного анализа. Слова, попавшие в него, относятся в ${}^1X^0, \dots, {}^7X^0$, или множество X^0 должно быть расширено новой категорией.

Рассмотренный метод носит итерационный характер. Признак конвергенции — размер словаря экзотических понятий. Если он опустится ниже эмпирически установленного порогового значения (до 10 %), то первый этап формирования концептуальной модели завершен. Релевантность (5.2) подтверждена адаптацией определений 4 — 15 к трем источникам смысла слов русского языка — словарям В.И. Даля, С.И. Ожегова, Большой Советской Энциклопедии и экспериментально [1].

5.3. Моделирование состояний и ситуаций

Гибридизация с целью разработки функциональных ГиИС для решения сложных задач требует манипулирования информацией, существенное место в которой занимают представления ресурсов, действий, структур, ситуаций и состояний объекта управления (рис. 7.3).

Для современных систем управления характерна чрезвычайно высокая сложность ресурсов, действий, их структур, ситуаций и состояний. Так, например, на современном судне для решения задач проектирования автоматике можно выделить до шести уровней иерархии, причем на нижних находятся тысячи экземпляров ресурсов с десятками свойств, которые нужно учитывать при конструировании. Нечеткие формулировки, зашумленность данных в ЯПД, вместе со структурной сложностью, делают задачу конструирования знаков ресурсов, действий, структур, ситуаций и состояний как концептуальных моделей не тривиальной даже в условиях современных баз данных и баз знаний. Синтез знаков информационных языков описания в настоящее время — скорее, искусство, чем наука. Искусство состоит в выработке и применении при синтезе эвристических правил, отвечающих на такие вопросы, как

«Что нужно обозначать в знаковой системе?», «Каков должен быть синтаксис знаков?», «Как выразить семантику и прагматику знаков?» и др.

Поэтому актуальны эвристики, ограничивающие множество вариантов конструирования знаков и направляющие разработку баз данных и знаний из унифицированных строительных блоков. Один из возможных способов записи таких эвристик — схемы ролевых концептуальных моделей, и, что особенно важно, эти схемы должны основываться на априори созданной концептуальной модели действительности, например

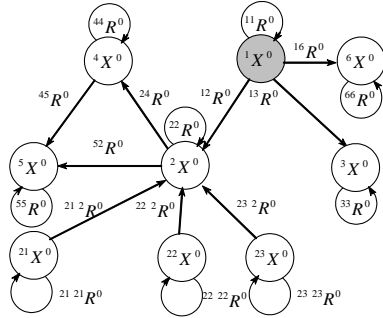


Рис. 5.7 — Схема ролевых концептуальных моделей для представления ресурсов

(5.2).

Анализ изображенной на рис. 5.6 модели показывает, что она дает также возможность формулировать эвристические записи производных знаков, синтаксис, семантика и прагматика которых не выражаются отдельным словом и требуют конструирования из составных частей. Чтобы сделать такие правила прозрачными, выделим из (5.2) фрагмент и изобразим

его на рис. 5.7. О чем «говорит» выделенный фрагмент? Очевидно, что следующее: «Чтобы построить знак-ресурс, необходимо показать его принадлежность к классу $1X^0$, указать его отношения с другими знаками в иерархии и пространстве (отношения включения и пространства из класса $11R^0$), задать отношения определения из $12R^0$ между вновь вводимым знаком и знаками свойств из $2X^0$. При этом достаточно использовать только параметры $21X^0$, характеристики $22X^0$ и имена $23X^0$. Задавая свойства, указать меру $4X^0$ и количество ее единиц $5X^0$ или, задав безразмерное значение $5X^0$, указать отношения $13R^0$ ресурса и выполняемых с его помощью или над ним действий $3X^0$. Задать состояние ресурса $6X^0$. Знак будет рассматриваться в контексте отношений $33R^0$, $44R^0$, $55R^0$, $66R^0$, $21R^0$, $22R^0$, $23R^0$ ». Ниже приведена схема ролевых концептуальных моделей ресурсов $ch_2 \in CH$:

$$\begin{aligned}
 ch_2 = & 1X^0 \ 11R^0 \ 1X^0 \ 2X^0 \ 22R^0 \ 2X^0 \ 3X^0 \ 33R^0 \ 3X^0 \ 4X^0 \ 44R^0 \ 4X^0 \circ \\
 & \circ 5X^0 \ 55R^0 \ 5X^0 \ 6X^0 \ 66R^0 \ 6X^0 \circ 21X^0 \ 21 \ 21R^0 \ 21X^0 \circ \\
 & \circ 22X^0 \ 22 \ 22R^0 \ 22X^0 \ 23X^0 \ 23 \ 23R^0 \ 23X^0 \circ 21X^0 \ 21 \ 2R^0 \ 2X^0 \circ \quad (5.3) \\
 & \circ 22X^0 \ 22 \ 2R^0 \ 2X^0 \ 23X^0 \ 232R^0 \ 2X^0 \circ 1X^0 \ 12R^0 \ 2X^0 \circ 1X^0 \ 13R^0 \ 3X^0 \circ \\
 & \circ 1X^0 \ 16R^0 \ 6X^0 \ 2X^0 \ 24R^0 \ 4X^0 \circ 4X^0 \ 45R^0 \ 5X^0 \circ 2X^0 \ 25R^0 \ 5X^0 \circ
 \end{aligned}$$

и пример концептуальной модели, построенной по схеме ch_2 для предметной области «проектирование систем судовой автоматики»:

двигатель = *быть* энергоустановкой
иметь параметр (оборотистость (*иметь единицу измерения* об/мин *иметь значение* 100) расход топлива (*иметь единицу измерения* литр/час *иметь значение* 7) стоимость (доллар США 1100000))
иметь характеристику (тип дизель марка RTA84T фирма-изготовитель SULZER)
иметь имя (серийный номер H3211)
быть частью (главная энергетическая установка)
иметь состав (картер, система смазки, система пуска воздухом, система питания)
структура (рядом сепаратор, быть на расстоянии 3 м от дизель-генератор, местоположение корма)
быть объектом (пуск, останов, холостой ход)
иметь состояние (пусковой режим).

Схема ролевых концептуальных моделей (рис. 5.7) и (5.3), при разработке ИЯ, может быть записана фреймом или сценарием и направлять конструирование знака. Поскольку даже примитивные шаблоны прагматичны для новичка, то сложные эвристики-схемы ролевых концептуальных моделей полезны и резко сокращают трудозатраты, возможные ошибки и итерации гибридизации. Конечно, по мере накопления опыта значимость эвристики снижается, однако возрастает необходимость ее модификации и тиражирования.

На рис. 5.8 изображен еще один фрагмент (5.2) для моделирования действий по аналогии с ресурсами, что сокращает разнообразие в обработке информации. Ниже эвристика моделирования действий записана схемой ролевых концептуальных моделей $ch_3 \in CH$:

$$\begin{aligned}
 ch_3 = & 1X^0 \ 11R^0 \ 1X^0 \ 2X^0 \ 22R^0 \ 2X^0 \ 3X^0 \ 33R^0 \ 3X^0 \ 4X^0 \ 44R^0 \ 4X^0 \ 5X^0 \ 55R^0 \ 5X^0 \ 7X^0 \ 77R^0 \ 7X^0 \ 21X^0 \ 21R^0 \ 2X^0 \ 22X^0 \ 22R^0 \ 2X^0 \ 23X^0 \ 23R^0 \ 2X^0 \ 3X^0 \ 31R^0 \ 1X^0 \ 3X^0 \ 32R^0 \ 1X^0 \ 3X^0 \ 37R^0 \ 7X^0 \ 2X^0 \ 24R^0 \ 4X^0 \ 4X^0 \ 45R^0 \ 5X^0 \ 2X^0 \ 25R^0 \ 5X^0, \\
 & \circ 5X^0 \ 55R^0 \ 5X^0 \ 7X^0 \ 77R^0 \ 7X^0 \ 21X^0 \ 21R^0 \ 2X^0 \ 22X^0 \ 22R^0 \ 2X^0 \ 23X^0 \ 23R^0 \ 2X^0 \ 3X^0 \ 31R^0 \ 1X^0 \ 3X^0 \ 32R^0 \ 1X^0 \ 3X^0 \ 37R^0 \ 7X^0 \circ. \quad (5.4)
 \end{aligned}$$

и дан пример концептуальной модели операции, построенной по (5.4):

пуск холодного двигателя = (проверка давления масла)
за которой следует (подогрев охлаждающей воды)
за которой следует (пуск воздуха для разгона)
проверка давления масла = *иметь объект* (масло для главного двигателя)
иметь субъект (датчик N)
иметь параметр (давление значение (P))
откуда (на выходе двигателя)
откуда (на входе двигателя)
куда (система мониторинга)
чем (кабель N)
время начала (время)
время конца (время) *оценка* (положительно).

Эвристики, аналогичные (5.3) и (5.4), могут быть сформулированы и для свойств, т.е. параметров, характеристик, имен, а также и для других категорий из X^0 .

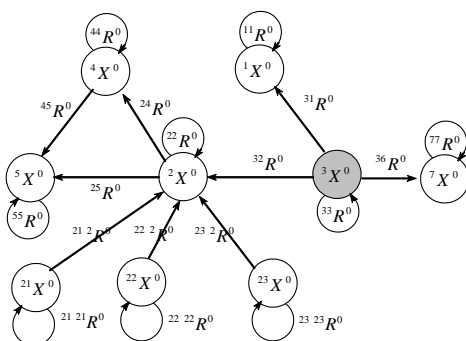


Рис. 5.8 — Схема ролевых концептуальных моделей для представления действий

В СППР помимо ресурсов, действий, свойств, их иерархий могут быть и другие реалии — структуры со связями «по горизонтали», более значимые, чем вертикальные иерархии подчиненности, принадлежности или вложенности. К таким структурам относятся пространственное расположение ресурсов, например акватория и территория в порту, план размещения полей в агро-

фирме, структурно - функциональные схемы судовой автоматики. Сюда же следует отнести и операциональные структуры на причинно-следственных и временных отношениях действий, например, технологические операции. Эти структуры, так же как и пространственные, относительно устойчивы во времени по сравнению с более динамичными структурами-ситуациями, которые формируются на состояниях, пространственных и временных отношениях ресурсов.

Эвристики их описания могут быть получены также в виде схем ролевых концептуальных моделей, построенных на отношениях, например, $^{11}R^0$ и $^{33}R^0$. Ниже приведены концептуальные модели знаков «причальная линия», «технологическая карта выгрузки» и «ситуация в индустриальной гавани» для морского порта:

Причальная линия = (причал № 14 **продолжение** причала № 13)
 (причал № 15 **продолжение** причала № 14)
 (причал № 16 **продолжение** причала № 15)
 (причал № 14 **непосредственно примыкает** к причалу № 13)
 (причал № 15 **непосредственно примыкает** к причалу № 14)
 (причал № 16 **непосредственно примыкает** к причалу № 15);

Технологическая карта выгрузки =
 (портальный кран **должен быть расположен рядом** с судном)
 (электропогрузчик **должен находиться в** трюме судна)
 (докеры **должны находиться на палубе в** трюме и **на причале**)
 (вагон **должен быть в** зоне действия крана)
 (электропогрузчик берет порцию груза **в трюме и перемещает ее в** зону действия крана) **раньше чем** (докер стропит порцию груза) **раньше чем**
 (докер на палубе дает сигнал крановщику о подъеме) **раньше чем**
 (кран перемещает груз **на причал на стол у входа в вагон**) **раньше чем**
 (докеры переносят груз по частям **в вагон**);

*Ситуация в индустриальной гавани = (судно Агат стоит у причала 13)
 (судно Топаз стоит у причала 14)(судно Изумруд стоит у причала 15)
 (электропогрузчик №126 в трюме судна Агат)
 (бригада Петрова на судне Агат)(Бригада Петрова на причале 13).*

В итоге последовательного построения и применения схем ролевых концептуальных моделей для анализа текстов языка профессиональной деятельности и выделения лексики, конструирования производных знаков ресурсов, действий, свойств их иерархий и структур может быть построена многоуровневая концептуальная модель информационного языка технологии функциональных ГиИС (7.2).

5.4. Модели задач и методов решения однородных задач

Изложим сказанное, исходя из модели задачи В.В. Нечаева (табл. 1.6):

$$\pi = \langle G, D, C \rangle. \quad (5.6)$$

Формула (5.6) отражает идеи индивидуального, несистемного, ассоциального интеллекта [13]. Гибридизация с целью разработки функциональных ГиИС требует ревизии и понятия задачи как объекта-оригинала. На рис. 5.9. приведена схема ролевых концептуальных моделей для определения понятия «однородная задача» — π^h (при необходимости ниже будем использовать обозначение ${}^8\pi^h$) через понятия «метода» — m , свойства — ${}^2X^0$ и состояния — ${}^6X^0$, а ниже дано ее описание:

$$\pi^h = {}^8x_1^0 \circ {}^{86}R^n \circ {}^6G^h \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2D^h \circ {}^8x_1^0 \circ {}^{89}R^n \circ {}^9C^h \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2K^h \circ {}^8x_1^0 \circ {}^{82}R^n \circ {}^2O^h \circ {}^8x_1^0 \circ {}^{88}R_1^n \circ {}^8x_1^0 \circ {}^8x_1^0 \circ {}^{88}R_2^n \circ {}^8x_1^0, \quad (5.7)$$

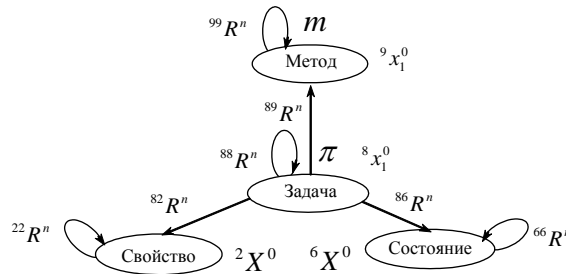


Рис. 5.9 — Схема ролевых концептуальных моделей для описания однородных задач

где ${}^6G^h, {}^2D^h, {}^9C^h$ — схемы ролевых концептуальных моделей цели, исходных данных и условий (m^h — метод, \dot{m}^h — модель, a^h — алгоритм, p^h — программа; при необходимости будем использовать обозначения ${}^9m^h, {}^9\dot{m}^h, {}^9a^h, {}^9p^h$) задачи π^h соответственно; ${}^2K^h$ — классификатор, т.е. схема ролевых концептуальных моделей, определяющая фазу (учет, контроль, анализ, нормирование, прогнозирование, планирование, регулирование, организация), класс переменных, которыми необходимо и достаточно манипулировать для решения π^h (детерминированные, стохастические, нечеткие или четкие лингвистические, генетические), класс задачи (размещения, расписания, маршрутизации, совершенствования и другие в соответствии с табл. 7.2 [1]) и множество классов переменных-отношений «сравнения», «пространства», «времени», «причинности» и другие, которыми необходимо и достаточно манипулировать для решения π^h ; ${}^2O^h$ — спецификатор, т.е. схема ролевых концептуальных моделей, определяющая идентификатор ${}^2O_1^h$, свойства среды разработки, проблемной среды, проекта, среды экспертов, измерений, информации для решения задачи; плана решения задачи [1]; ${}^{88}R_1^n$ — отношения с другими π^h из одной и той же π^u (5.10); ${}^{88}R_2^n$ — отношения включения π^h в π^u .

Такой подход к определению задачи вводит ее классификационную принадлежность фазам управления, учитывает модели личностей [244], свойства проекта, системы измерений, информации для решения задачи, а также обмен информацией между подзадачами в составе неоднородной задачи. Термин «план решения» трактуется аналогично [245].

Остановимся подробнее на определении идентификатора задачи, чему практически не уделяется внимание при гибридизации. Чтобы восполнить этот пробел, в работе [246] предложена следующая схема ролевых концептуальных моделей идентификатора:

$${}^2O_1^h = {}^8x_1^0 \circ {}^{82}R_1^n \circ {}^{22}X_1^0 \circ {}^8x_1^0 \circ {}^{82}R_2^n \circ {}^{22}X_2^0 \circ {}^8x_1^0 \circ {}^{83}R^n \circ {}^3X^0 \circ \circ {}^3X^0 \circ {}^{32}R^n \circ {}^{22}X^0 \circ {}^3X^0 \circ {}^{31}R_1^n \circ {}^1X^0 \circ {}^3X^0 \circ {}^{31}R_2^n \circ {}^1X^0, \quad (5.8)$$

где ${}^{22}X_1^0, {}^{22}X_2^0$ — фаза и характеристики задачи; ${}^3X^0$ — действие, на реализацию которого направлено решение задачи; ${}^1X^0$ — ресурсы для выполнения действия; ${}^{82}R_1^n, {}^{82}R_2^n$ — отношения определения «иметь фа-

зу», «иметь свойство»; ${}^8R^n$ — отношения «задача-действие»; ${}^{32}R^n$ — отношения «действие-свойства»; ${}^{31}R_1^n, {}^{31}R_2^n$ — отношения «иметь место» и «иметь средство» соответственно.

По (5.8) может быть получен, например, следующий идентификатор однородной задачи: *задача оперативного контроля погрузки-выгрузки судов в грузовом районе порта*.

В языке описания однородных задач L_6^h , как подмножестве языка концептуального моделирования, основы которого сформулированы в работе [247], по (5.7) может быть сконструирован знак ${}^8x^n|{}^h \in {}^8X^n|{}^h$, где ${}^8X^n|{}^h$ — множество знаков однородных задач:

$$L_6^h(L_1^S, \dots, L_5^S; {}^6X^0, {}^8X^0, {}^9X^0, R^n; \pi^h) = \{{}^8x^n|{}^h\}, \quad (5.9)$$

где L_1^S, \dots, L_5^S — языки описания производных отношений R^n , а также ресурсов, свойств, действий и их иерархий (${}^1X^n, {}^2X^n, {}^3X^n$), структур, ситуаций и состояний объекта управления (7.1).

Язык L_6^h — средство спецификации однородных задач в мире задач (рис. 7.3), на котором могут быть определены задачи-системы, т.е. неоднородные задачи π^u , специфицируемые в соответствии со следующей схемой ролевых концептуальных моделей (рис. 5.10):

$$\begin{aligned} \pi^u = & {}^8x_1^0 \circ {}^8R^n \circ {}^6G^u \circ {}^8x_1^0 \circ {}^8R^n \circ {}^2D^u \circ {}^8x_1^0 \circ {}^8R^n \circ {}^9C^u \circ {}^8x_1^0 \circ {}^8R^n \circ {}^2K^u \circ \\ & \circ {}^8x_1^0 \circ {}^8R^n \circ {}^2O^u \circ {}^8x_1^0 \circ {}^8R^n \circ {}^3X^n \circ {}^8x_1^0 \circ {}^8R^n \circ {}^7O^u \circ {}^8x_1^0 \circ {}^8R^n \circ \Pi^h \circ \\ & \circ \Pi^h \circ {}^8R^n \circ \Pi^h \circ {}^8x_1^0 \circ {}^8R^n \circ \Pi^h, \end{aligned} \quad (5.10)$$

где ${}^6G^u, {}^2D^u, {}^9C^u$ — схемы ролевых концептуальных моделей цели, исходных данных, условий (m^u, \dot{m}^u, a^u, p^u) соответственно; ${}^2K^u$ — классификатор; ${}^2O^u$ — спецификатор, включающий идентификатор π^u , схему ролевых концептуальных моделей ЛПР; схему ролевых концептуальных моделей операции ${}^3X^n$, исполняющей решение π^u ; ${}^7O^u$ — эвалюэйтор (от англ. evaluation), т.е. схема ролевых концептуальных моделей, определяющая оценки результатов ${}^3X^n$, оценки результатов решения однородных задач $\Pi^h = \{\pi_1^h, \dots, \pi_{N_h}^h\}$ из состава π^u , где N_h — здесь и ниже обозначает количество элементов множества;

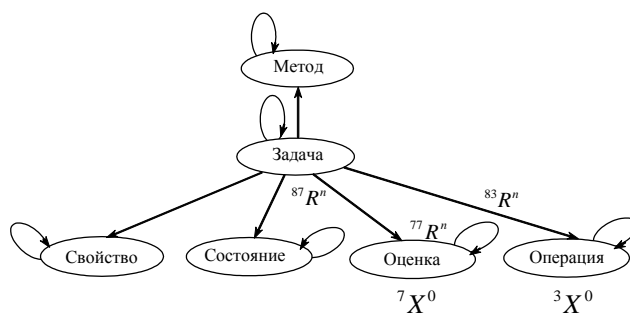


Рис. 5.10 — Схема концептуальных моделей для описания неоднородных задач

$\Pi^h \text{ } ^{88}R_3^n \Pi^h$ — обозначает $\hat{\Pi}^u = \{ \hat{\pi}_1^u, \dots, \hat{\pi}_{N_u}^u \}$ — множество декомпозиций задачи π^u , $\hat{\pi}_j^u = (\Pi^h, \text{ } ^{88}R_3^n) | j = 1, \dots, N_u$, $\text{ } ^{88}R_3^n$ — множество отношений декомпозиции π^u ; $\text{ } ^8x_1^0 \text{ } ^{88}R_2^n \Pi^h$ — обозначает состав π^u ; $\text{ } ^{88}R_4^n$ — отношения $\text{ } ^6G^u$ и $\text{ } ^6G^h$, $\text{ } ^2D^u$ и $\text{ } ^2D^h$, а также $\text{ } ^9C^u$ и $\text{ } ^9C^h$; серая заливка используется в описании операций ПС-технологии функциональных ГиИС (разд. 7.3).

В языке описания неоднородных задач L_7^u , как подмножестве информационного языка технологии функциональных ГиИС (разд. 7), может быть сконструирован знак $\text{ } ^8x^n |^u \in \text{ } ^8X^n |^u$, где $\text{ } ^8X^n |^u$ — множество знаков неоднородных задач:

$$L_7^u(L_6^h, \text{ } ^8X^0 |^{uh}, \text{ } ^8X^n |^\Pi; \pi^u) = \{ \text{ } ^8x^n |^u \}, \quad (5.11)$$

где $\text{ } ^8X^n |^{uh} \subseteq \text{ } ^8X^n |^h$ — множество знаков в L_6^h однородных задач из Π^h ; $\text{ } ^8X^n |^\Pi$ — множество знаков декомпозиций неоднородных задач в $L_7^\Pi(L_6^h, \text{ } ^{88}R_3^n, CH^\Pi) = \{ \text{ } ^8x^n |^\Pi \}$, где CH^Π — множество схем ролевых концептуальных моделей для формирования структур на Π^h .

Таким образом, в соответствии с (5.7), (5.9) — (5.11) следует различать схемы ролевых концептуальных моделей однородной π^h и неоднородной π^u , а также знаки однородной x^h и неоднородной x^u задач.

Исследуем отдельные элементы π^u (5.10). Цели ${}^6G^u$ и ${}^6G_1^h, \dots, {}^6G_{N_h}^h$, а также исходные данные ${}^2D^u$ и ${}^2D_1^h, \dots, {}^2D_{N_h}^h$, в общем случае, не совпадают. Например, ${}^2D_1^h, \dots, {}^2D_{N_h}^h$ могут быть детализированы, а ${}^2D^u$ более абстрактны. Решение однородной задачи экспертом не предполагает каких-либо действий в объекте управления и их оценку, в то время как по результатам решения неоднородной задачи такие действия выполняются, а их результаты оцениваются ЛПР. Иными словами, решение π^h носит консультативный, а π^u исполнительный характер. Такая классификация информации хорошо укладывается в традиционную модель управления и классификацию задач по фазам управления. Условия ${}^9C^u$ и ${}^9C_1^h, \dots, {}^9C_{N_h}^h$ отличаются в принципе, поскольку, если для ${}^9C_1^h, \dots, {}^9C_{N_h}^h$ должен быть известен $m_j^h | j = 1, \dots, N_h$, иначе субъект-эксперт, решающий π^h , некомпетентен, и могут быть известны a_j^h и p_j^h , то для ${}^9C^u$ метод m^u , а также a^u, p^u — искомые и конструируемые при гибридизации объекты.

Если $N_h = 1$, то имеем вырожденный случай существования (π^u ${}^{88}r^n \quad \pi^h \Big| {}^{88}r^n \in {}^{88}R_2^n$ — «целое-часть». При этом ${}^6G^u = {}^6G^h, {}^2D^u = {}^2D^h, {}^9C^u = {}^9C^h, {}^2K^u = {}^2K^h, {}^2O^u = {}^2O^h$ в мире задач (рис. 7.3), когда ЛПР и эксперт одно и то же лицо.

Таким образом, по отношению к миру методов моделирования (рис. 7.3) однородные и неоднородные задачи занимают соответственно крайнее левое и крайнее правое положения на шкале «известное—неизвестное». Это подчеркивает то, что если метод решения задачи неизвестен, то возможны два пути: 1) создавать такой метод в рамках одной из школ; 2) проверить задачу на неоднородность в соответствии с (5.10) и, если эти условия выполняются, перейти к конструированию метода решения.

Сравнительный анализ моделей (5.7) и (5.10) показывает также, что модель (5.10) — общий случай по отношению к (5.7). К (5.7) легко перейти, предположив в (5.10), что $\Pi^h = \emptyset$. Следствие этого предположения — невозможность построения декомпозиции $\hat{\Pi}^u$ и передача права ЛПР по организации действий в объекте управления и оценке этих действий эксперту.

Концептуальные модели методов решения однородных задач.

Метод m^h и построенные в соответствии с ним модели \dot{m}^h , алгоритмы a^h и программы p^h , входя в схему ролевых концептуальных моделей для спецификации однородных задач (5.7) — объекты-прототипы, подлежат анализу в ходе гибридизации. В этой связи исследуем отношения «метод-свойства» ($^{12}R^n$) и «метод-метод» ($^{11}R^n$) (рис. 5.11).

Многообразие в мире методов моделирования и, как следствие, многообразие моделей, наработанных в научных школах, следует отнести к одному из самых замечательных результатов XX столетия. Однако именно из-за многообразия эти достижения бывает подчас трудно использовать на практике, поскольку это требует от разработчика интеллектуальных информационных систем, прежде всего, широкого кругозора для сравнительного анализа и выбора метода в сочетании с глубокими знаниями конкретного метода, т.е. отношений «метод-свойства», а также тенденций в мире методов моделирования (рис. 7.3) в целом, т.е. отношений «метод-метод».

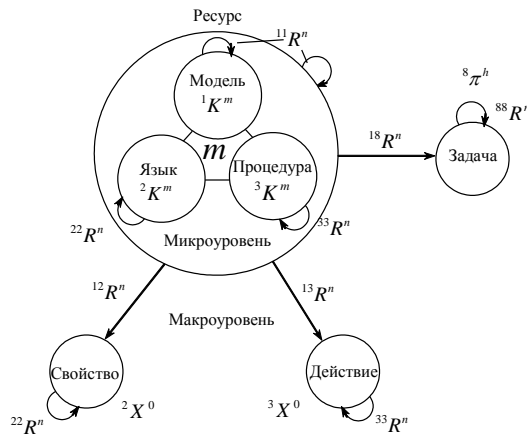


Рис. 5.11 — Представление метода как объекта-прототипа гибридизации

Рожденный в умах ученых и отображающий поэтому, безусловно, какие-то личностные, субъективные черты своего создателя, метод моделирования, как знание, будучи формализованным на некотором языке, будь то язык математики, моделирующих алгоритмов или другой, начинает свою «жизнь» в мире методов моделирования. В жизни,

применительно к миру задач, проявляются ограниченность, достоинства и недостатки метода, т.е. сильные и слабые стороны. Будучи изученным другими специалистами, имеющими свое субъективное видение внешнего мира, свои профессиональные знания, метод может быть ими модифицирован или видоизменен качественно настолько, что превратится уже в другой, новый метод. Такая деятельность в мире методов моделирования приводит к явлениям, когда метод-первооснова перестает су-

ществовать, в буквальном смысле впитывая в себя многочисленные «дополнения» и «изменения». Примеров таких явлений можно привести много. Типичный из них — экспертные системы, которые настолько «ушли» от своих теоретических первооснов — метода резолюции, продукционных систем, что не осталось даже чисто внешнего сходства.

Дополнения и изменения, как правило, заимствуются из других методов, в которых недостатки, характерные для метода-оригинала, отсутствуют. В итоге, в мире методов моделирования происходят изменения методов-сущностей, вызванные корректурой свойств методов. Это изменения в ходе эволюции метода, с присущими ей отбором методов-родителей для нового метода-потомка, оценка метода-потомка с точки зрения поставленных создателями целей и другими явлениями (см. разд. 4.1). Такой процесс можно рассматривать как эволюцию популяции методов в мире методов моделирования и распространить на него подходы, принятые в генетике. Для этого необходимо, прежде всего, представить методы-сущности на двух уровнях: в макро- и микромире.

Макроуровневое представление трактует метод как некоторый ресурс разработчика для выполнения действий при решении задачи и обладающий отличительными свойствами (рис. 5.11).

Для макроуровневого представления метода при гибридизации введем следующую схему ролевых концептуальных моделей:

$$\hat{m}^a = {}^1X^0 \text{ } ^{11}R^n \text{ } ^1X^0 \circ {}^1X^0 \text{ } ^{12}R^n \text{ } ^{22}X_1^0 \circ {}^1X^0 \text{ } ^{12}R^n \text{ } ^{22}X_2^0 \circ {}^1X^0 \text{ } ^{12}R^n \text{ } ^{22}X_3^0 \circ \quad (5.12)$$

$$\circ {}^1X^0 \text{ } ^{13}R^n \text{ } ^3X_1^0 \circ {}^1X^0 \text{ } ^{13}R^n \text{ } ^3X_2^0 \circ {}^1X^0 \text{ } ^{18}R^n \text{ } ^8\pi^h \circ {}^1X^0 \text{ } ^{12}R^n \text{ } ^2O^m,$$

где ${}^{22}X_1^0 \subseteq {}^2X^0$ — классификатор, т.е. одна из характеристик: ${}^{22}x_{11}^0$ — «аналитический», ${}^{22}x_{12}^0$ — «статистический», ${}^{22}x_{13}^0$ — «символьный», ${}^{22}x_{14}^0$ — «коннекционистский», ${}^{22}x_{15}^0$ — «эволюционный»; ${}^{22}X_2^0 \subseteq {}^2X^0$ — модель, т.е. одна из характеристик: ${}^{22}x_{21}^0$ — «черный ящик», ${}^{22}x_{22}^0$ — «система массового обслуживания», ${}^{22}x_{23}^0$ — «система автоматического управления», ${}^{22}x_{24}^0$ — «условие-действие», ${}^{22}x_{25}^0$ — «серый ящик», ${}^{22}x_{26}^0$ — «эволюция», ${}^{22}x_{27}^0$ — «ситуация-решение»; ${}^{22}X_3^0 \subseteq {}^2X^0$ — язык описания, т.е. одна или несколько характеристик: ${}^{22}x_{31}^0$ — «уравнения», ${}^{22}x_{32}^0$ — «алгоритм», ${}^{22}x_{33}^0$ — «продукции», ${}^{22}x_{34}^0$ — «матрицы» и др.; ${}^3X_1^0 \subseteq {}^3X^0$ — процедура получения решения,

т.е., например, действия: ${}^3x_{11}^0$ — «прямой» и ${}^3x_{12}^0$ — «обратный вывод» в ЭС, ${}^3x_{13}^0$ — «прямое распространение» в ИНС, ${}^3x_{14}^0$ — «нечеткий вывод», ${}^3x_{14}^0$ — «методы решения уравнений», ${}^3x_{15}^0$ — «машинные эксперименты с ГА или моделирующим алгоритмом», ${}^3x_{16}^0$ — методы установления соответствия на множествах прецедентов, поиска аналогов и сохранения новых единиц опыта в памяти и др.; ${}^3X_2^0 \subseteq {}^3X^0$ — процедура обучения, т.е., например, ${}^3x_{21}^0$ — «обратное распространение», ${}^3x_{22}^0$ — «алгоритм Кохонена», ${}^3x_{23}^0$ — «непараметрическое обучение» и др.; ${}^2O^m$ — спецификатор, т.е. схема ролевых концептуальных моделей, определяющая погрешность решения, гибридные возможности (количественные экспертные оценки потенциала комбинирования с другими методами, см. табл. 8.6 [1]), а также знания о преимуществах и недостатках (отображения на шкалы «хорошая — плохая» работа с шумом, «хорошая — плохая» адаптивность и др. [1]); ${}^{11}R^n, {}^{12}R^n, {}^{13}R^n$ — отношения определения; ${}^{18}R^n$ — отношения предназначения.

Микроуровневое представление (на рис. 5.11 внутри большой окружности) трактует метод как отношения его составных частей: модели, языка описания, процедуры, специфицируемые следующей схемой ролевых концептуальных моделей:

$$\begin{aligned} \tilde{m}^a = & {}^1K^m {}^{11}R_1^n {}^1K^m \circ {}^2K^m {}^{22}R^n {}^2K^m \circ {}^3K^m {}^{33}R^n {}^3K^m \circ \\ & \circ {}^1K^m {}^{12}R^n {}^2K^m \circ {}^1K^m {}^{13}R^n {}^3K^m \circ {}^2K^m {}^{23}R^n {}^3K^m, \end{aligned} \quad (5.13)$$

где ${}^1K^m, {}^2K^m, {}^3K^m$ — схемы ролевых концептуальных моделей модели, языка и процедуры соответственно (см. «гибридная модель интеграции знаний экспертов» в разд. 4.1); ${}^{11}R^n, {}^{22}R^n, {}^{33}R^n, {}^{12}R^n, {}^{13}R^n, {}^{23}R^n$ — отношения «модель-модель», «язык-язык», «процедура-процедура», «модель-язык», «модель-процедура», «язык-процедура», соответственно.

Обратим внимание на три обстоятельства. Первое состоит в том, что если рассматривать эволюцию в мире методов моделирования, то должно быть установлено отношение $\hat{m}^a \leftrightarrow \tilde{m}^a$ для выполнения преобразования «макроописание \leftrightarrow микроописание». Второе: (5.13) может быть включена в макроуровневую схему ролевых концептуальных моделей (5.12) как характеристика и отобразит факт, что, не выполнив каких-либо изменений в \tilde{m}^a , невозможно изменить метод m^a качественно. Из трех составных частей микроуровневого представления метода, что уже

отмечалось в разд. 4.1, наименее подвержены изменениям модель ${}^1K^m$ и язык ${}^2K^m$. Значительно чаще встречаются ситуации изменения процедуры «работы» с моделью. Именно разнообразие процедур и определяет разнообразие макроуровневых (внешних) свойств метода, не касаясь его сущности, во многом определяемой моделью и языком. Третье исходит из дуальности (5.13) и проясняет ответ на вопрос: «Что же все-таки предмет исследований междисциплинарных методов — «модели-системы» или «методы-системы»?». Метод и модель — два тесно взаимосвязанных друг с другом объекта. Будучи разработанной в соответствии с некоторым методом, модель, а далее — алгоритм и программа, всегда будут несвободны, связаны его свойствами, наследуя его плюсы и минусы. С другой стороны, будучи созданной, модель выступает в роли инструмента решения задач, и метод унаследует свойства модели. Более того, метод не может выйти за рамки ${}^1K^m$ (5.13) и наследует от нее структурирование внешнего мира, понятийный аппарат. Тем не менее, в дальнейшем, при необходимости для отличия «метода» от «модели», будем использовать обозначения m и \dot{m} соответственно.

По существу, (5.12) специфицирует метод с помощью ограничений, выделяя для него нишу в популяции методов. Эта ниша характеризуется, прежде всего, \tilde{m}^a (5.13). Если нужно выбрать метод для решения задачи, то выход за пределы ниши означает, что ограничения, введенные в схеме ролевых концептуальных моделей метода, не устраивают разработчика, и он вынужден искать инструмент за пределами ниши, т.е. в другой, третьей и т.д. нишах. В итоге компромисс «ограничения — адекватность» может быть найден, а метод идентифицирован.

Однако при решении сложных, неоднородных задач, т.е. задач-систем, значительно чаще подобного компромисса достичь не удастся. Поиск по нишам ничего не дает, и возникает необходимость выхода за рамки ограничений уже известных методов. Что такое явление может означать? То, что разработчик имеет дело с неоднородной по природе задачей и ему нужно отказаться от парадигмы формирования и поддержки устойчивой ниши в мире методов моделирования (рис. 7.3). Для неоднородной задачи, с изменчивостью ее состава и структуры, нет и не может быть релевантных ниш, а метод ее решения конструируется над некоторой совокупностью ниш, обеспечивающих решение только однородных задач из состава неоднородной проблемы.

В пределах одной ниши существуют модели, которые принято [74] называть автономными, чтобы подчеркнуть их несвязанность (автономность) с другими моделями решения задачи. Решение однородной задачи может моделироваться в конкретной нише, и поэтому методы, обра-

зующие эту нишу, называются автономными методами (m^a), применяемыми для разработки и экспериментов с автономными моделями \dot{m}^a , алгоритмами \dot{a}^a и программами \dot{p}^a , совокупность которых и образует мир автономных методов W^{m^a} (6.13).

В языке описания методов L_7^m , как подмножестве информационного языка (см. разд. 7), по схемам (5.12) и (5.13) могут быть сконструированы знаки ${}^9x^n|_m \in {}^9X^n|_m$, где ${}^9X^n|_m$ — множество знаков автономных методов:

$$L_7^m(L_1^S, \dots, L_5^S; L_6^h; \hat{m}^a, \tilde{m}^a) = \{{}^9x^n|_m\}. \quad (5.14)$$

5.5. Элементы функциональных гибридных интеллектуальных систем

Для моделирования функциональных ГиИС введены ограничения (табл. 5.1), суть которых состоит в выборе методов из базисных классов методов моделирования рассуждений экспертов (разд. 3). Там же показано распределение методов по трем стратам (разд. 6.3).

Из табл. 5.1 видно, что высшему уровню абстрагирования — параметрической страте S_3 системы S (2.1) релевантен наибольший плюрализм — три класса методов, остальным двум стратам — ситуационной S_1 и потоковой S_2 — по одному классу методов. Такая многоальтернативность на параметрической страте объясняется, прежде всего, необходимостью численного представления параметров задач, что свойственно аналитическим, нейросетевым и нечетким вычислениям.

Последние два столбца табл. 5.1 содержат ссылки на концептуальные модели, модели вычислений автономных методов из разд. 3.

Дальнейшие построения выполним в неформальной аксиоматической теории Te (5.1). Для этого детализируем верхний уровень в схеме концептуальной модели, изображенной на рис. 5.12, введя понятие «элемент». Под элементом (компонентой) будем понимать ресурс функциональной гибридной интеллектуальной системы для моделирования решения однородной задачи.

Элементы $\alpha^h|_j^\omega \in A^h|_j, j=1, \dots, 7; \quad \omega=1, \dots, N_h$, которые имитируют решение однородных задач π^h , будем называть функциональными. Кроме функциональных, в состав ГиИС могут включаться и технологи-

ческие компоненты $\alpha^{\tau|l}_j \in A^{\tau}|j=1,\dots,7; l=1,\dots,N_{\tau}$, организующие управление и (или) обмен информацией на множестве A^h и выполняющие вспомогательные действия.

Таблица 5.1 — Элементы функциональных ГиИС

Класс Обозначение Группа	Ограничения	Страта	Концептуальная модель	Модель вычислений
Аналитические вычисления $\alpha^h _1$ 1. Не составной I (3.3)	Алгебраические и дифференциальные уравнения	Параметри- ческая S_3	(3.1)	(3.3)
Нейровычисления $\alpha^h _2$ 1. Не составной I (3.26)	Сети прямого распро- странения сигнала Карты Кохонена		(3.24)	(3.25)
Нечеткие вычисления $\alpha^h _3$ 2. Составной I (3.24)	Mamdani Takagi-Sugeno		(3.22)	(3.23)
Рассуждения на основе опыта $\alpha^h _4$ 2. Составной I (3.34)	Рассуждения на осно- ве прецедентов	Ситуацион- ная S_1	(3.32)	(3.33)
Эволюционные вычисления $\alpha^h _5$ 3. Составной I (3.30)	Классические ГА с турнирной селекци- ей Парето ГА с нишами	Параметри- ческая S_3 Ситуацион- ная S_1	(3.28)	(3.29)
Статистические вычисления $\alpha^h _6$ 3. Составной, ите- рационный I (3.13)	Метод Монте-Карло	Потоковая S_2	(3.11)	(3.12)
Логические рассуждения $\alpha^h _7$ 3. Составной, ите- рационный I (3, 16, 3.17)	Продукционные экспертные системы	Ситуацион- ная S_1	(3.14)	(3.15)

Элемент, моделируя решение некоторой π^h или выполняя вспомогательные операции, построен в соответствии с некоторым автономным методом и имеет определенные свойства (рис. 5.12).

Отношения ${}^1X^0 \ {}^{13}R^n \ {}^3X^0$ раскрывают методологический аспект элемента: конструирование, тестирование, имитацию решения π^h и эксплуатацию. Этот аспект для элементов $\alpha^h|_1 - \alpha^h|_6$ из табл. 5.1 рассмотрен в [1].

Отношения ${}^1X^0 \ {}^{19}R^n \ {}^9x_1^0$, ${}^1X^0 \ {}^{19}R^n \ {}^9x_2^0$ — это отношения определения элементов-методов или элементов-моделей соответственно. Отношения ${}^1X^0 \ {}^{19}\hat{R}^n \ \hat{m}^a$, ${}^1X^0 \ {}^{19}\tilde{R}^n \ \tilde{m}^a$, где ${}^{19}\hat{R}^n, {}^{19}\tilde{R}^n \subseteq {}^{19}R^n$ детализируют элемент через схемы ролевых концептуальных моделей макро- (5.12) или микроуровневого (5.13) описания метода.

Отношения ${}^1X^0 \ {}^{12}R^n \ {}^2X^0$ специфицируют свойства элемента, из

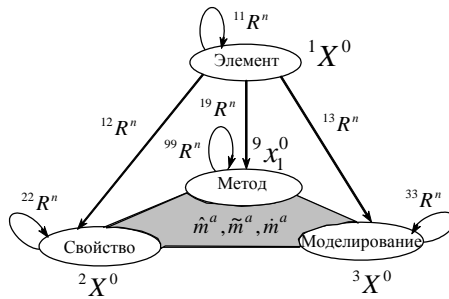


Рис. 5.12 — Схема ролевых концептуальных моделей элементов функциональных ГИИС (крупное зерно)

множеств которых наиболее существенны три: «вход» — ${}^2\tilde{x}_1^n$, «выход» — ${}^2\tilde{x}_2^n$ и «состояние» — ${}^2\tilde{x}_3^n$ относительно модельного времени $t \in T$.

Свойства «вход» и «выход» позволяют элементам воспринимать исходную информацию ${}^2D^h$ и выдавать результаты ${}^6G^h$ для решения однородных задач π^h . Свой-

ство «состояние» фиксирует совокупность параметров и характеристик процесса, определяемого соответствующей моделью вычислений из табл. 5.1.

Вышесказанное, по существу, определяет несколько из возможных языков конфигуратора для описания элементов функциональных ГИИС. В качестве наиболее абстрактного языка выберем представление элементов через их «вход», «состояние» и «выход». В качестве другого, детального представления можно использовать схемы ролевых концептуальных моделей и модели вычислений из табл. 5.1.

Представление элементов свойствами ${}^2X^0$ в неформальной аксиоматической теории Te (5.1) названо крупнозернистым описанием:

$$\alpha^h(t)|_j = {}^1X^0 \circ {}^{19}R^n \circ {}^9\hat{m}^a \circ {}^1X^0 \circ {}^{12}R^n \circ {}^2\tilde{x}_1^n \circ {}^1X^0 \circ {}^{12}R^n \circ {}^2\tilde{x}_2^n \circ \circ {}^2\tilde{x}_1^n(t) \circ {}^{22}R_1^n \circ {}^2\tilde{x}_2^n(t)|_{j=2,3,4} \quad (5.15)$$

или

$$\alpha^h(t)|_j = {}^1X^0 \circ {}^{19}R^n \circ {}^9\hat{m}^a \circ {}^1X^0 \circ {}^{12}R^n \circ {}^2\tilde{x}_1^n \circ {}^1X^0 \circ {}^{12}R^n \circ {}^2\tilde{x}_2^n \circ \circ {}^1X^0 \circ {}^{12}R^n \circ {}^2\tilde{x}_3^n \circ {}^2\tilde{x}_3^n(t) \circ {}^{22}R_2^n \circ {}^2\tilde{x}_3^n(t+1) \circ {}^2\tilde{x}_1^n(t) \circ {}^{22}R_3^n \circ {}^2\tilde{x}_3^n(t) \circ \circ {}^2\tilde{x}_3^n(t) \circ {}^{22}R_4^n \circ {}^2\tilde{x}_2^n(t)|_{j=1,5,6,7}, \quad (5.16)$$

где ${}^{22}R_1^n, \dots, {}^{22}R_4^n$ — отношения «вход-выход», «состояние-состояние», «вход-состояние» и «состояние-выход» соответственно.

Различия (5.15) и (5.16) объясняются семантикой векторов ${}^2\tilde{x}_1^n$, ${}^2\tilde{x}_2^n$ и ${}^2\tilde{x}_3^n$. Вычисления в $\alpha^h(t)|_2$, $\alpha^h(t)|_3$ и $\alpha^h(t)|_4$ (5.15) непрозрачны, промежуточные состояния имитационного процесса не могут интерпретироваться в терминах решаемой задачи, и наблюдение за поведением $\alpha^h(t)|_2$, $\alpha^h(t)|_3$ и $\alpha^h(t)|_4$ неинформативно.

В отличие от (5.15) компоненты $\alpha^h(t)|_1$, $\alpha^h(t)|_5$, $\alpha^h(t)|_6$, $\alpha^h(t)|_7$ имеют поведенческий характер (5.16), т.е. наблюдаемо и интерпретируемо их состояние и поведение, можно говорить о релевантности состояния элемента и состояния объекта управления, что дает экспертам информацию для решения задачи π^h .

Из сказанного следует, что крупнозернистое описание использует представление элемента и вычислений в терминах «вход — состояние — выход», определяется декомпозицией сложной задачи и прозрачно для установления соответствия между однородными задачами в составе неоднородной и элементами ГиИС, что очень важно в начале работ по гибридизации. При таком подходе «размер» зерна для гибридизации зависит только от π^h (5.7). Микроуровневое представление метода не затрагивается, и качественно новые свойства функциональной ГиИС получаются за счет установления отношений на макроуровне. Гибридизацию с использованием (5.15) и (5.16) будем называть крупнозернистой гибридизацией.

Мелкозернистое описание элементов ГиИС может быть проведено на различных уровнях детализации. Возможные варианты мелкозернистого описания элементов позволяет учесть группа оператора I в моделях вычислений (первый столбец табл. 5.1).

Классификацию этих формализмов выполним, используя выражение схемы ролевых концептуальных моделей метода на микроуровне (5.13). Систематизируем $\dot{m}_q^a|q = \{a, n, f, g, s, e, p\}$ по процедуре — ${}^3K^m$ и отношению «процедура-процедура» — ${}^{33}R^n$.

Тогда в первую группу попадут формализмы аналитических и нейровычислений, выражающих семантику метода не «составным» интерпретатором, т.е. не имеющим выделенные и обозначенные в I^a (3.3), I^{n2} (3.26), I^p (3.34) части-процессы интерпретатора. Вторая группа — это формализмы нечетких вычислений и рассуждений на основе опыта, использующие для выражения семантики автономного метода составной интерпретатор I^f (3.24) не циклического (итерационного) характера. Третья группа — это формализмы эволюционных, статистических вычислений и логических рассуждений, выражающих семантику автономного метода составными, итерационными интерпретаторами I^g (3.30), I^s (3.13) и I^e (3.17) соответственно.

Смысл разделения элементов на группы состоит в том, чтобы прояснить для разработчика, «что», дающее преимущества одному методу, может быть взято при гибридизации и на место «чего» в формализме другого элемента это «что» может быть вставлено. Для такой гибридизации необходимо располагать знаниями о сильных и слабых сторонах автономных методов. Такая работа проделана только на уровне шести классов методов [1].

Сформулируем формализмы трех групп. Для этого обратимся к рис. 5.13, построенному по аналогии с рис. 5.12, однако «метод» представлен детализировано, на микроуровне (5.13). Обозначения схем ролевых концептуальных моделей и отношений взяты из (5.13). Тогда для группы 1 имеем:

$$\begin{aligned} \tilde{\alpha}^h|_1 = & {}^1X^0 {}^{19}R^n {}^9\dot{m}_a^a|Te \circ {}^1X^0 {}^{11}R^n {}^1K^m \circ {}^1X^0 {}^{12}R^n {}^2K^m \circ \\ & \circ {}^1X^0 {}^{13}R^n {}^3K^m \circ {}^2K^m {}^{22}R_s^n {}^2K^m \circ {}^2K^m {}^{23}R^n {}^3K^m, \end{aligned} \quad (5.17)$$

$$\begin{aligned} \tilde{\alpha}^h|_2 = & {}^1X^0 {}^{19}R^n {}^9\dot{m}_n^a|Te \circ {}^1X^0 {}^{11}R^n {}^1K^m \circ {}^1X^0 {}^{12}R^n {}^2K^m \circ \\ & \circ {}^1X^0 {}^{13}R^n {}^3K_1^m \circ {}^1X^0 {}^{13}R^n {}^3K_2^m \circ {}^2K^m {}^{22}R_s^n {}^2K^m \circ \\ & \circ {}^2K^m {}^{23}R^n {}^3K^m \circ {}^3K_1^m {}^{33}R_1^n {}^3K_2^m, \end{aligned} \quad (5.18)$$

где ${}^9\dot{m}_a|Te, \dot{m}_n|Te$ — концептуальные модели, построенные по схеме \hat{m}^a (5.12); ${}^3K_1^m, {}^3K_2^m$ — схемы ролевых концептуальных моделей интерпретаторов I^{n1} и I^{n2} нейросети соответственно; ${}^2K^m$ — схема ролевых концептуальных моделей декларативных описаний, т.е. F (3.3) и Ar, X, Y, M^n, Ed (3.25) для вычислений по аналитическим зависимостям и нейровычислений соответственно; ${}^{11}R^n, {}^{12}R^n, {}^{13}R^n, {}^{19}R^n$ — отношения определения; ${}^{22}R^n$ — отношения включения одних декларативных описаний в другие ($\exists \tilde{\alpha}^h|_j \quad {}^{22}R_5^2 = \emptyset$); ${}^{23}R^n$ — отношения декларативных описаний и процедур, т.е. указывающие, в каких действиях интерпретатора используются те или иные описания из ${}^2K^m$; ${}^{33}R_1^n$ — отношения следования, определяющие порядок действий по обучению, тестированию и нейровычислениям.

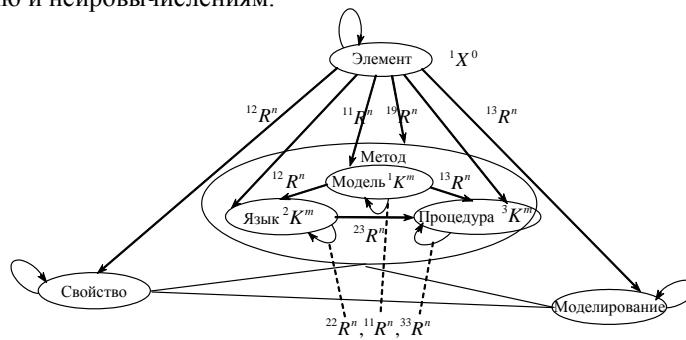


Рис. 5.13 — Схема ролевых концептуальных моделей элементов функциональных ГИС (мелкое зерно)

Для группы 2 имеем:

$$\begin{aligned} \tilde{\alpha}^h|_j &= {}^1X^0 \circ {}^{19}R^n \circ {}^9\dot{m}_l|Te \circ {}^1X^0 \circ {}^{11}R^n \circ {}^1K^m \circ \\ &\circ {}^1X^0 \circ {}^{12}R^n \circ {}^2K^m \circ {}^1X^0 \circ {}^{13}R^n \circ {}^3K^m \circ {}^2K^m \circ {}^{22}R_5^n \circ {}^2K^m \circ \\ &\circ {}^2K^m \circ {}^{23}R^n \circ {}^3K^m \circ {}^3K^m \circ {}^{33}R_1^n \circ {}^3K^m, \end{aligned} \quad (5.19)$$

где $j=3,4$; $l=\{f,p\}$; ${}^{33}R_1^n$ — отношение следования, упорядочивающее процессы I^{f1}, \dots, I^{f4} , если $l=f$, или I^{p1}, \dots, I^{p4} , если $l=p$.

Для группы 3 имеем:

$$\begin{aligned} \tilde{\alpha}^h|_j = & \mathop{\circ}\limits^1 X^0 \mathop{\circ}\limits^{19} R^n \mathop{\circ}\limits^9 m_l | Te \mathop{\circ}\limits^1 X^0 \mathop{\circ}\limits^{11} R^n \mathop{\circ}\limits^1 K^m \mathop{\circ}\limits^1 X^0 \mathop{\circ}\limits^{12} R^n \mathop{\circ}\limits^2 K^m \mathop{\circ}\limits^{\circ} \\ & \mathop{\circ}\limits^1 X^0 \mathop{\circ}\limits^{13} R^n \mathop{\circ}\limits^3 K^m \mathop{\circ}\limits^2 K^m \mathop{\circ}\limits^{22} R_5^n \mathop{\circ}\limits^2 K^m \mathop{\circ}\limits^2 K^m \mathop{\circ}\limits^{23} R^n \mathop{\circ}\limits^3 K^m \mathop{\circ}\limits^{\circ} \quad (5.20) \\ & \mathop{\circ}\limits^3 K^m \mathop{\circ}\limits^{33} R_1^n \mathop{\circ}\limits^3 K^m \mathop{\circ}\limits^3 K^m \mathop{\circ}\limits^{33} R_2^n \mathop{\circ}\limits^3 K^m, \end{aligned}$$

где $\mathop{\circ}\limits^{33} R_2^n$ — отношения, изменяющие порядок выполнения процессов интерпретатора, заданный $\mathop{\circ}\limits^{33} R_1^n$ на итерационный; $j=5,6,7$; $l=\{g,s,e\}$, причем, если $j=5$, то $l=c$, если $j=6$, то $l=s$, если $j=7$, то $l=e$.

Рассмотренные схемы ролевых концептуальных моделей (5.17) — (5.20) предполагают конструирование функциональных ГиИС, по архитектуре которых достаточно трудно судить об их сходстве с объектами-оригиналами, т.е. со сложной задачей. Вместе с тем открывается возможность разработки «хороших методов». Мелкое зерно имеет и свои недостатки. Трудно ответить на вопрос, какой размер модуля должен использоваться? От чего он зависит? Может быть легко потеряна смысловая нить, связывающая модуль и подзадачу.

Оба подхода: крупно- и мелкозернистое описание элементов приводят к различным логическим последовательностям и содержанию методологии и технологии гибридизации.

В языке описания элементов функциональных ГиИС L_8^{ah} , как подмножестве информационного языка, по схемам (5.15) — (5.20), могут быть сконструированы знаки $\mathop{\circ}\limits^1 x^n |^{ah}, \mathop{\circ}\limits^1 \tilde{x}^n |^{ah} \in \mathop{\circ}\limits^1 X^n |^{ah}$, где $\mathop{\circ}\limits^1 X^n |^{ah}$ — множество знаков элементов:

$$L_8^{ah} (L_1^S, \dots, L_5^S; L_6^m; \alpha^h(t)|_j, \tilde{\alpha}^h(t)|_j) = \{ \mathop{\circ}\limits^1 x^n |^{ah}, \mathop{\circ}\limits^1 \tilde{x}^n |^{ah} \}. \quad (5.21)$$

5.6. Крупнозернистые функциональные гибридные интеллектуальные системы

Синтез ГиИС (α^u) для решения неоднородной задачи π^u (5.10), имеющей вход, выход, состояние и состоящей из элементов $\alpha^h \in A^h$ и $\alpha^t \in A^t$, выполняется для модели внешнего мира $W^{\pi^h \pi^u m^a m^i}$ (6.19). Пусть имеем множество декомпозиций $\hat{\Pi}^u = \{ \hat{\pi}_1^u, \dots, \hat{\pi}_{N_u}^u \}$ задачи π^u ,

где $\hat{\pi}_j^u = (\Pi^h, {}^{88}r_{3j}^n) | j = 1, \dots, N_u$ — декомпозиция π^u по отношению ${}^{88}r_{3j}^n \in {}^{88}R_3^n$ — множество отношений декомпозиции π^u на Π^h , $\Pi^h = \{\pi_1^h, \dots, \pi_{N_h}^h\}$ — множество из N_h однородных задач в составе π^u . Для каждой однородной задачи π^h (5.7) должны быть заданы ${}^{88}R_1^n$ отношения с другими однородными задачами из одной и той же π^u . Заданы также ${}^{88}R_4^n$ отношения совпадения, уточнения, преобразования и другие соответствующих элементов π^u и π^h для каждой $\hat{\pi}_j^u$. Заданы также множества A^h и A^τ , функциональных и технологических элементов соответственно, причем $\exists \alpha^u$ ($A^\tau = \emptyset$). Кроме этого, установлено соответствие между элементами A^h и элементами Π^h .

Вход ${}^2\hat{\mathbf{x}}_1^n$ — это исходные данные ${}^2D^u$ задачи π^u , передаваемые на вход одной или нескольких $\alpha^h | j$ (или $\alpha^\tau | j$), в соответствии с $\hat{\pi}^u$. Состояние гибрида α^u в момент времени t — вектор: ${}^2\hat{\mathbf{x}}_3^n = ({}^2\hat{\mathbf{x}}_3^n | h, {}^2\hat{\mathbf{x}}_3^n | hp, {}^2\hat{\mathbf{x}}_3^n | \tau)$, где ${}^2\hat{\mathbf{x}}_3^n | h = (\{ {}^2\hat{\mathbf{x}}_3^n | h1 \}, \{ {}^2\hat{\mathbf{x}}_3^n | h3 \}, \{ {}^2\hat{\mathbf{x}}_3^n | h5 \}, \{ {}^2\hat{\mathbf{x}}_3^n | h6 \})$, ${}^2\hat{\mathbf{x}}_3^n | \beta j$, $j = 1, 4, 5, 6$; $\omega = 1, \dots, N_h$ — состояние «поведенческого» ω -го функционального элемента ($\beta = h$; $\omega = 1, \dots, N_h$) или технологического ($\beta = \tau$; $j = 1, \dots, N_\tau$) из j -го базового класса методов, ${}^2\hat{\mathbf{x}}_3^n | hp = (\{ {}^2\hat{\mathbf{x}}_3^n | hp2 \}, \{ {}^2\hat{\mathbf{x}}_3^n | hp3 \})$ псевдосостояние функциональных элементов с нейро- ($j=2$), нечеткими ($j=3$) вычислениями и ($j=4$) рассуждениями на основе опыта, ${}^2\hat{\mathbf{x}}_3^n | \tau = (\{ {}^2\hat{\mathbf{x}}_3^n | \tau1 \}, \{ {}^2\hat{\mathbf{x}}_3^n | \tau2 \}, \{ {}^2\hat{\mathbf{x}}_3^n | \tau3 \}, \{ {}^2\hat{\mathbf{x}}_3^n | \tau4 \}, \{ {}^2\hat{\mathbf{x}}_3^n | \tau5 \}, \{ {}^2\hat{\mathbf{x}}_3^n | \tau6 \})$ — состояние элементов из A^τ .

Выход ГИИС ${}^2\hat{\mathbf{x}}_2^n$ — это выход одного или нескольких элементов $\alpha^h | j$ в соответствии с $\hat{\pi}^u$, являющийся целью ${}^6G^u$ решения задачи π^u . Тогда схема ролевых концептуальных моделей для спецификации крупнозернистых функциональных ГИИС, функционирующих в неоднородном пространстве состояний, имеет вид:

$$\begin{aligned}
\alpha^n(t) = & {}^1X^{019}R^{n9}m^i \circ {}^1X^{012}R^{n2}\widehat{\mathbf{x}}_1^n \circ {}^1X^{012}R^{n2}\widehat{\mathbf{x}}_2^n \circ \\
& \circ {}^1X^{012}R^{n2}\widehat{\mathbf{x}}_3^n \circ {}^2\widehat{\mathbf{x}}_3^n(t) {}^{22}R_6^n \widehat{\mathbf{x}}_3^n(t+1) \circ \\
& \circ {}^2\widehat{\mathbf{x}}_1^n(t) {}^{22}R_7^n \widehat{\mathbf{x}}_3^n(t) \circ {}^2\widehat{\mathbf{x}}_3^n(t) {}^{22}R_8^n \widehat{\mathbf{x}}_2^n(t) \circ \\
& \circ {}^1X^{011}\ddot{R}^{n1}X^n \circ {}^2\widehat{\mathbf{x}}_1^n {}^{22}R_9^n X_1^n \circ {}^2X^n {}^{22}R_{10}^n \widehat{\mathbf{x}}_2^n,
\end{aligned} \tag{5.22}$$

где ${}^1X^n \subseteq X^h$ — множество знаков, построенных по (5.21) информационного языка *min* из двух элементов; ${}^2X_1^n, {}^2X_2^n$ — множества свойств «вход» и «выход» элементов из ${}^1X^n$ соответственно; ${}^{22}R_6^n, {}^{22}R_7^n, {}^{22}R_8^n$ — отношения функционирования ГиИС; ${}^{11}\ddot{R}^n$ — отношения интеграции элементов из (6.15); ${}^{22}R_9^n, {}^{22}R_{10}^n$ — отношения «входа» ГиИС и «входов» элементов, а также «выходов» элементов и «выхода» ГиИС соответственно. Поскольку ${}^2\widehat{\mathbf{x}}_3^n$ включает в качестве компонент разнородные переменные, назовем состояние функциональных ГиИС неоднородным.

На рис. 5.14 изображено пространство состояний ГиИС для трех элементов $\alpha^h|_1, \alpha^h|_5, \alpha^h|_6$ (по одному $\omega=1$ из трех классов, $j = \{1,5,6\}$) и двух однородных переменных: ${}^2\widehat{\mathbf{x}}_{31}^n, {}^2\widehat{\mathbf{x}}_{32}^n$ для каждого

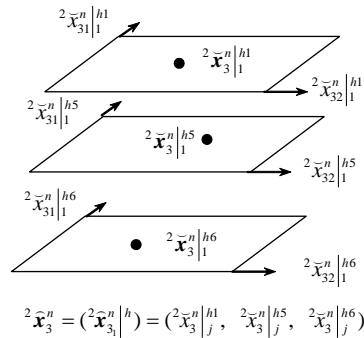


Рис. 5.14 — Пример графического представления состояния функциональных ГиИС

элемента.

Обозначим неоднородное пространство, в котором функционирует ГиИС как ${}^2\widehat{\mathbf{X}}_3^n = \{ {}^2\widehat{\mathbf{x}}_3|_1, \dots, {}^2\widehat{\mathbf{x}}_3|_{N_x} \}$.

Отношения ${}^{22}R_6^n, {}^{22}R_7^n, {}^{22}R_8^n$ не задаются априори, как это происходит с подобными отношениями ${}^{22}R_2^n, {}^{22}R_3^n, {}^{22}R_4^n$ при конструировании элементов, а фиксируются в ходе функционирования ГиИС и могут использоваться для объяснения решения неоднородной задачи.

Отношения ${}^{11}\ddot{R}^n$ — отношения интеграции (6.15), классификация которых рассмотрена в разд. 6.4. При разработке функциональных ГиИС они заменяют отношения декомпозиции ${}^{88}r_{3j}^n \in {}^{88}R_3^n$ и отношения ${}^{88}R_1^n$ между однородными задачами π^h , входящими в состав неоднородной задачи π^u .

Отношения соответствия ${}^{22}R_9, {}^{22}R_{10}$ устанавливаются из следующих соображений. Они задаются на множествах переменных ${}^2D^u, {}^6G^u$ и множествах переменных ${}^2D^h, {}^6G^h$ однородных задач, входящих в состав неоднородной задачи. При этом возможно несколько вариантов: 1) множество переменных для π^u может совпадать с соответствующим множеством для $\pi^h \in \Pi^h$, т.е. вся исходная информация по π^u передается для решения π^h и становится, в свою очередь, ее исходной информацией; 2) множество переменных для π^h — подмножество соответствующего множества для неоднородной задачи (т.е. только часть информации передается для решения π^h , и отношения в этом случае устанавливают взаимно-однозначное соответствие между переменными из соответствующих множеств для π^u и π^h); 3) множество для π^u — подмножество соответствующего множества π^h (это случай, когда множество исходных данных для π^h пополняется информацией из базы данных и знаний модели, соответствующей этой однородной задаче).

Модель (5.22) определяет, что функциональная ГиИС — это система, имеющая архитектуру обмена информацией, вход, выход и функционирующая в неоднородном пространстве состояний ${}^2\widehat{X}_3^n$. Назовем такое функционирование гибридным имитационным процессом. Если ГиИС решает задачу в СППР, то возможна аналогия, в соответствии с которой элементы имитируют решение однородных задач π^h экспертами, а α^u имитирует коллективное решение π^u под управлением ЛППР.

5.7. Мелкозернистые функциональные гибридные интеллектуальные системы

Для мелкозернистой гибридизации постановка задачи сохраняется аналогичной постановке задачи в предыдущем разделе. Отличия касаются спецификации множества методов, моделей, алгоритмов и программ решения однородных задач. Здесь предполагается, что практиче-

ски нет гетерогенного модельного поля (разд. 6.4), представляющего интерес для решаемой неоднородной задачи. Причиной такой ситуации может быть уникальный характер возникшей перед СППР проблемы в сочетании с усугубляющим трудность ситуации положением, когда известные методы не могут быть применены к решению однородных (или одной однородной задачи) из множества Π^h . Приведем пример такой ситуации. Пусть для решения некоторой однородной задачи, как знака в L_6^h (5.9), по всем свойствам π^h подходит экспертная система. Однако поскольку предположительно ее база знаний будет иметь большой размер, то известные алгоритмы ее интерпретации будут работать медленно. Такая ситуация не позволяет применить при гибридизации крупнозернистый элемент, построенный как экспертная система, и требует вмешательства в микроуровневое представление. Цель — заменить отдельные процедуры (I^{e1}, \dots, I^{e4}) интерпретатора I^e (3.17) — мелкие зерна на, чаще всего, полнофункциональный, т.е. крупнозернистый элемент, построенный по другому методу. Иногда возможна и замена «мелкое зерно — мелкое зерно». Это одно из объяснений. Другое объяснение следует из необходимости использовать не только классы трансформационных и функциональных ГиИС, как в случае крупнозернистой гибридизации, но и полиморфические ГиИС (разд. 4.5), которые сконструировать из крупного зерна невозможно. Наконец, в рамках мелкозернистых гибридов проще решать технологические задачи, сопутствующие решению $\pi^h \in \Pi^h$, и создавать элементы α^r . Типичный пример — задача извлечения знаний для экспертной системы из декларативной информации других элементов функциональных ГиИС.

Можно привести еще несколько подробных задач: корректировки знаний элементов ГиИС после того, как в базе знаний одного из элементов были внесены изменения; пополнения исходной информации, поступающей на вход элемента как за счет его собственной базы знаний и данных, так и баз данных и баз знаний других элементов. Список таких технологических, но, тем не менее, важных задач можно пополнить, и все они требуют разработки механизма вмешательства в декларативную и (или) процедурную составляющие элемента функциональных ГиИС априори, еще до синтеза ГиИС.

Для решения рассмотренных выше проблем можно перейти к гибридизации на мелкозернистом уровне. Следует иметь в виду, что для разработчика при этом теряется одно из преимуществ крупнозернистой гибридизации — использование широко распространенных коммерческих программных продуктов. Необходимы специальные, требующие

трудоемкого системного программирования программные средства-инструментарии модульного построения элементов.

К схеме ролевых концептуальных моделей мелкозернистых функциональных ГиИС (5.23) можно перейти от схемы (5.22), выполнив следующие преобразования: 1) заменить модели крупнозернистых элементов (5.15), (5.16) на модели мелкозернистых элементов (5.17) — (5.20); 2) ввести отношения гибридизации на (5.17) — (5.20).

Тогда введем следующую схему ролевых концептуальных моделей мелкозернистых функциональных ГиИС:

$$\tilde{\alpha}^u = \alpha^u \circ {}^1x^n \quad {}^{11}R^n \quad {}^1X^{n-1} \circ X^{n-1} \quad {}^{11}\ddot{R}^n \quad {}^1X^{n-1}, \quad (5.23)$$

где ${}^1X^{n-1}$ — множество элементов-зерен для конструирования $\alpha^h \in A^h$ и $\alpha^\tau \in A^\tau$; ${}^{11}R^n$ — отношения включения зерен из ${}^1X^{n-1}$ в знак ГиИС ${}^1x^n$; ${}^{11}\ddot{R}^n$ — отношения интеграции зерен из ${}^1X^{n-1}$.

Модель (5.23) — более общая, чем модель (5.22), к которой она сводится, если допустить, что ${}^{11}R_2^n$ на ${}^1X_2^n, {}^1X_2^{n-1}$ и ${}^{11}R_1^n$ на ${}^1X_1^n, {}^1X_1^{n-1}$ не установлены. Модель (5.23) предполагает двухуровневую гибридизацию. На верхнем выполняется функциональная гибридизация, т.е. «от неоднородной задачи», а на нижнем — гибридизация «от метода».

В языке описания функциональных ГиИС L_9^{au} , как подмножестве языка концептуального моделирования, по схемам (5.22), (5.23) могут быть сконструированы знаки ${}^1x^n(t) \Big|^{au}, {}^1\tilde{x}^n(t) \Big|^{au} \in {}^1X^n \Big|^{au}$, где ${}^1X^n \Big|^{au}$ — множество знаков функциональных ГиИС:

$$L_9^{au}(L_1^S, \dots, L_5^S; L_8^{ah}, {}^{11}\ddot{R}^n; \alpha^u(t), \tilde{\alpha}^u(t)) = \{{}^1x^n(t) \Big|^{au}, {}^1\tilde{x}^n(t) \Big|^{au}\}. \quad (5.24)$$

Модель (5.23) обобщает полностью интегрированные и полиморфические ГиИС (разд. 4.5).

5.8. Модели поведения функциональных гибридных интеллектуальных систем

Рассмотрим функционирование и организацию гибридного имитационного процесса в ГиИС, заданной (5.22) и (5.23).

Функционирует ГиИС в неоднородном пространстве состояний ${}^2\widehat{\mathbf{X}}_3^n$ (рис. 5.15). На этом рисунке использованы обозначения, аналогичные рис. 5.14.

Каждый элемент $\alpha^h|_j^\omega$ изменяет только свою компоненту вектора состояния ${}^2\bar{\mathbf{x}}_3^n|_{\omega}^{hj}(t)$. В некоторый момент t , определяемый $\alpha^\tau|_j^\omega$ (например, экспертной системой) или совпадающий с моментом времени завершения работы $\alpha^h|_j^\omega$, однородное функционирование текущей $\alpha^h|_j^\omega$ прерывается и α^u скачком переходит в состояние, определяемое уже другой $\alpha^h|_q^\omega, q = 1, \dots, N_h$.

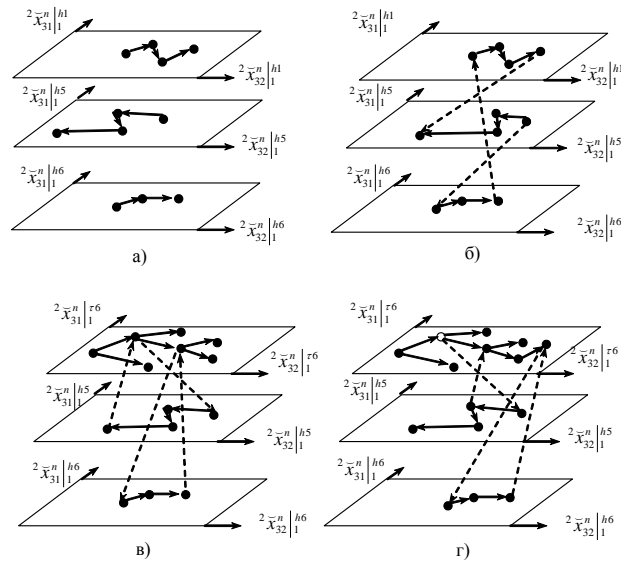


Рис. 5.15 — Графическое представление функционирования ГиИС

В момент t_G , также устанавливаемый $\alpha^\tau|_j^\omega$ или в соответствии с $\widehat{\pi}^u \in \widehat{\Pi}^u$, т.е. в момент времени, когда отработают все функциональные элементы ГиИС в порядке, определяемом декомпозицией неоднородной задачи, выдается результат ${}^6G^u$ решения задачи π^u . Из сказанного следует, что функционирование ГиИС определяется ее архитектурой, т.е.

отношениями ${}^{11}R^n$, задающими порядок обмена информацией между элементами. Кроме этого, должны быть определены отношения ${}^{22}R^n$ на свойствах ${}^2X^n$ (рис. 5.16), связывающие состояния из разных подпространств и задающие переход из одного однородного подпространства в другие во время функционирования ГиИС.

Тогда введем следующие схемы ролевых концептуальных моделей (рис. 5.16), специфицирующие поведение ФГиИС:

$$\alpha^{uu}(t) = \alpha^u \circ {}^2\bar{x}_3^n(t) {}^{22}R^n {}^2\bar{x}_3^n(t+1), \quad (5.25)$$

$$\tilde{\alpha}^{uu}(t) = \tilde{\alpha}^u \circ {}^2\bar{x}_3^n(t) {}^{22}R^n {}^2\bar{x}_3^n(t+1). \quad (5.26)$$

На рис. 5.15,а показано функционирование автономных моделей. В этом случае элементы работают независимо друг от друга, последовательно, т.е. сначала один элемент, затем другой, наконец — третий, решая одну и ту же или разные однородные задачи.

Функционирование ГиИС может быть организовано и по-другому, когда одна и та же однородная задача решается различными элементами одновременно, т.е. параллельно, а за изменением вектора состояния ${}^2\bar{x}_3^n|_h(t)$

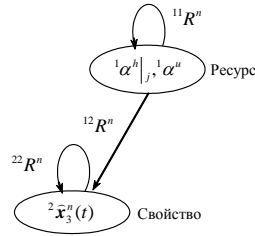


Рис. 5.16 — Схема ролевых концептуальных моделей функционирования ГиИС

ГиИС имеет возможность наблюдать пользователь. При этом возникает эффект, хорошо известный в электронике, когда одна и та же схема моделируется как аналоговое устройство, что дает возможность наблюдать тепловые, волновые эффекты и дискретно-логическое устройство, что позволяет исследовать диаграммы форм и амплитуд импуль-

сов. Отношения интеграции элементов здесь возникают как внутренние невербальные образы (разд. 3.5) в памяти пользователя, который, сравнивая динамику работы устройства с разных точек зрения, может «увидеть» то, чего не удастся обнаружить в рамках одномодельного моделирования.

Ниже приведена концептуальная модель функционирования ГиИС, построенная по схеме, изображенной на рис. 5.16:

$$\left\{ \begin{array}{l} {}^2\bar{x}_3^n|_1 h1(t) \rightarrow {}^2\bar{x}_3^n|_1 h1(t+1) \rightarrow \dots \rightarrow {}^2\bar{x}_3^n|_1 h1(t+n) \\ {}^2\bar{x}_3^n|_1 h5(t) \rightarrow {}^2\bar{x}_3^n|_1 h5(t+1) \rightarrow \dots \rightarrow {}^2\bar{x}_3^n|_1 h5(t+n) \\ {}^2\bar{x}_3^n|_1 h6(t) \rightarrow {}^2\bar{x}_3^n|_1 h6(t+1) \rightarrow \dots \rightarrow {}^2\bar{x}_3^n|_1 h6(t+n) \end{array} \right\}. \quad (5.27)$$

В (5.27) фигурные скобки обозначают параллельное функционирование и возможность наблюдения за динамикой изменения ${}^2\tilde{\mathbf{x}}_3^{|h}(t)$.

На рис. 5.15,б графически изображено функционирование наиболее распространенной последовательной (цепочной) архитектуры функциональных ГиИС. Начавшись в одном подпространстве (на рисунке они обозначены плоскостями), обработка информации последовательно проходит еще как минимум одно подпространство. Переход из одного подпространства в другое определяется завершением обработки информации каждым функциональным элементом ГиИС, которое априори и жестко установлено разработчиком. Только после того, как будет окончена работа элемента, сформируется информация для функционирования очередного элемента. При этом компоненты вектора состояния ${}^2\tilde{\mathbf{x}}_3^{|h}(t)$ меняются последовательно в соответствии с тем, в каком подпространстве идет обработка информации в данный момент модельного времени. После однократного прохождения всей цепочки элементов функционирование может быть продолжено в соответствии с итерационным циклом. Момент завершения итераций определяется пользователем или включением в архитектуру знаний в виде технологического элемента.

Ниже приведена концептуальная модель функционирования ГиИС с цепочной архитектурой:

$$\begin{aligned}
 & {}^2\tilde{\mathbf{x}}_3^{|h_1}(t) \rightarrow {}^2\tilde{\mathbf{x}}_3^{|h_1}(t+1) \rightarrow \dots \rightarrow {}^2\tilde{\mathbf{x}}_3^{|h_1}(t+n) \Rightarrow \\
 & \Rightarrow {}^2\tilde{\mathbf{x}}_3^{|h_3}(t+n+1) \rightarrow {}^2\tilde{\mathbf{x}}_3^{|h_3}(t+n+2) \rightarrow \dots \rightarrow \\
 & \rightarrow {}^2\tilde{\mathbf{x}}_3^{|h_3}(t+n+m) \Rightarrow {}^2\tilde{\mathbf{x}}_3^{|h_6}(t+n+m+1) \rightarrow \\
 & \rightarrow {}^2\tilde{\mathbf{x}}_3^{|h_6}(t+n+m+2) \rightarrow \dots \rightarrow {}^2\tilde{\mathbf{x}}_3^{|h_6}(t+n+m+k),
 \end{aligned} \tag{5.28}$$

где стрелки « \Rightarrow » обозначают отношения ${}^{22}R^n$.

Если модель, определяющую единственную итерацию, обозначить как ${}^{22}It$, то ниже приведена еще одна модель, но уже итерационного функционирования цепочного гибрида: ${}^{22}It(t) \mapsto {}^{22}It(t+1) \mapsto \dots \mapsto \mapsto {}^{22}It(t+f)$, где t — момент модельного времени окончания очередной итерации; символ « \mapsto » обозначает отношения ${}^{22}R^n$, связывающие одну итерацию с другой.

После завершения очередной итерации технологический элемент проверяет значение ${}^2\tilde{\mathbf{x}}_3^n|_1^h(t)$, а возможно, подсчитывает число итераций, или значение модельного времени, или ждет реакции пользователя для установления момента завершения моделирования.

На рис. 5.15,б показана более сложная организация функционирования ГиИС. Для этого использована архитектура ГиИС с одним технологическим элементом, функционирующим в подпространстве (${}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t)$). Пусть это будет производственная экспертная система, управляющая на мета-уровне двумя элементами с состояниями (${}^2\tilde{\mathbf{x}}_1^n|_1^{h5}(t)$) и (${}^2\tilde{\mathbf{x}}_1^n|_1^{h6}(t)$) и несущими функциональную нагрузку по решению однородных задач.

По достижении технологическим элементом некоторого состояния управление, безусловно, передается одному, а может быть, и нескольким параллельно работающим функциональным элементам. После завершения работы элемент возвращает управление, не изменяя при этом состояние $\alpha^{\tau 6}_1$. В рассмотренном варианте завершение работы ГиИС контролируется мета-элементом, который и отслеживает выполнение отношений интеграции на функциональных элементах.

Ниже приведена концептуальная модель функционирования ГиИС с мета-элементом:

$$\begin{aligned}
 & {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t) \rightarrow {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+1) \Rightarrow \\
 & {}^2\tilde{\mathbf{x}}_1^n|_1^{h5}(t+1) \rightarrow {}^2\tilde{\mathbf{x}}_1^n|_1^{h5}(t+2) \rightarrow \dots \rightarrow {}^2\tilde{\mathbf{x}}_1^n|_1^{h5}(t+n) \Rightarrow \\
 & {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+n) \Rightarrow \tag{5.29} \\
 & {}^2\tilde{\mathbf{x}}_1^n|_1^{h6}(t+n) \rightarrow {}^2\tilde{\mathbf{x}}_1^n|_1^{h6}(t+n+1) \rightarrow \dots \rightarrow {}^2\tilde{\mathbf{x}}_1^n|_1^{h6}(t+n+m) \Rightarrow \\
 & {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+n+m) \rightarrow {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+n+m+1).
 \end{aligned}$$

На рис. 5.15,г показано функционирование ГиИС, отличающееся от изображенного на рис. 5.15,б тем, что множество состояний мета-элемента разделено на два класса: обычные, они показаны черными точками, и особые, обозначенные окружностями. После того как технологический элемент достигнет особого состояния, он передает управление функциональному элементу, поведение которого может быть прервано в зависимости от выполнения некоторого условия. В простейшем случае задается число итераций. При этом технологический элемент не

прерывает свою работу и не ждет окончания работы функционального элемента, а переходит в следующее состояние.

Подобная схема функционирования предполагает параллельную обработку информации и возможность одновременного изменения компонент вектора состояния ${}^2\tilde{\mathbf{x}}_3^n|_1^h(t)$.

Ниже приведена концептуальная модель функционирования ГиИС с особыми состояниями:

$$\begin{aligned}
 & {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t) \rightarrow {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+1) \Rightarrow \\
 & {}^2\tilde{\mathbf{x}}_3^n|_1^{h5}(t+1) \rightarrow {}^2\tilde{\mathbf{x}}_3^n|_1^{h5}(t+2) \Downarrow \\
 & {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+2) \rightarrow {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+3) \rightarrow {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+4) \Rightarrow \\
 & {}^2\tilde{\mathbf{x}}_3^n|_1^{h6}(t+4) \rightarrow {}^2\tilde{\mathbf{x}}_3^n|_1^{h6}(t+5) \rightarrow \dots \rightarrow {}^2\tilde{\mathbf{x}}_3^n|_1^{h6}(t+n) \Rightarrow \\
 & {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+n) \rightarrow {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+n+1) \rightarrow \dots \rightarrow {}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}(t+n+m),
 \end{aligned} \tag{5.30}$$

где ${}^2\tilde{\mathbf{x}}_3^n|_1^{\tau 6}$ — особое состояние технологического элемента; символ « \Downarrow » обозначает установление отношений ${}^{22}R_3^n \subseteq {}^{22}R^n$ между технологическим и функциональными элементами, прерывающими работу первого.

Концептуальные модели функционирования ГиИС (5.27) — (5.30) будем называть гибридными стратегиями. Из сказанного выше следует, что гибридная стратегия определяется архитектурой ГиИС.

В языке описания поведения функциональных ГиИС L_{10}^{be} , как подмножестве информационного языка, по схемам (5.25), (5.26) могут быть сконструированы знаки ${}^6be^n \in {}^6Be^n$, где ${}^6Be^n$ — множество знаков поведения функциональных ГиИС:

$$L_{10}^{be}(L_1^S, \dots, L_5^S; L_9^{uu}; {}^{22}R^n; \alpha^{uu}(t), \tilde{\alpha}^{uu}(t)) = \{{}^6be^n\}. \tag{5.31}$$

6. МЕТОДОЛОГИЯ ФУНКЦИОНАЛЬНЫХ ГИБРИДНЫХ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

6.1. Понятие проблемно-структурной методологии функциональных гибридных интеллектуальных систем

В основе многих фундаментальных научных исследований лежит процесс образования и использования таких основных понятий, как «методология», «метод» и «теория». До полной ясности их смысла еще далеко. Есть мнение о тождественности теории, метода и методологии. Некоторые авторы видят в методологии не самостоятельную научную дисциплину, а часть философии. Отсутствие правильного понимания терминов «методология», «метод», «теория» вносит путаницу и затрудняет познание.

В работе [248] дан аналитический обзор содержания различных трактовок понятия «методология».

В соответствии с первой точкой зрения методология — это методики, совокупность методов, приемов, способов организации теоретической и практической деятельности.

Ранее такая трактовка была широко распространена в отечественной научной литературе. В работах современных авторов она встречается реже. Значительно чаще такое понимание методологии свойственно западным исследователям, что объясняется их ориентацией на решение в первую очередь не фундаментальных, а практических задач, вытекающих из потребностей развития кибернетики, теории систем, информатики, теории управления и т. д. Такой подход приводит к прагматизации исследований в сфере методологии.

Вторая точка зрения дополняет первую: ее сторонники исходят из того, что методология — это не только способы, процедуры и методы практической и теоретической деятельности, но и принципы, лежащие в ее основе.

Третья точка зрения объединяет две предыдущие и выводит на новый уровень толкования методологии. Под методологией понимается не только система принципов, способов и методов организации теоретической и практической деятельности, но и учение о такой системе. Своим появлением данная концепция во многом обязана Ф. Бэкону, Г. Гегелю, Р. Декарту, И. Канту, Д. Локку.

Четвертая точка зрения сложилась к концу 60 — 70-х годов XX века благодаря трудам Московского методологического кружка. Ее суть в том, что методология — не просто учение о средствах и методах мышления и деятельности, а совершенно особая форма организации всей мыследеятельности и жизнедеятельности людей. По объему своего со-

держания данная позиция более широкая, чем предыдущая, и поэтому включает в себя учение о принципах и методах (способах) деятельности. В соответствии с этой точкой зрения смысл методологической работы — обеспечить переход от мира знаний, в котором находятся образцы, правила, эталоны, теории и другие (синтагматика) к миру практической, научной и любой другой деятельности (парадигматике). Иными словами, исследователь, теоретик или практик, прежде чем приступить к конкретной деятельности, осознает себя в новом качестве (в качестве методолога) и обращается к синтагматике. Посредством существующих там знаний он формулирует или уточняет предмет, объекты и принципы своей деятельности, выбирает необходимые теоретические знания, соответствующие предмету подходы, методики и методы работы. При отсутствии последних методолог разрабатывает их и представляет свою будущую деятельность, проектирует ее и ожидаемые результаты.

Таким образом, значение методологической работы — максимально эффективно организовать последующую деятельность. Поэтому главная функция методологии — организационная. В связи с этим методология выступает не как теория деятельности, а как теория организации деятельности, в том числе и мыследеятельности. При этом методологическая работа превращается в методологию, т.е. теорию особого рода, только тогда, когда ученый переходит из позиции методолога в позицию теоретика, описывающего, систематизирующего и объясняющего собранный методологический материал. Именно в результате труда последнего и появляется методология.

Представляется, что четвертая точка зрения на настоящий момент достаточно полная, чтобы можно было сформулировать частное понятие «методология гибридных интеллектуальных систем».

Методология гибридных интеллектуальных систем — учение о структуре, логической организации, методах и средствах деятельности разработчика.

Методология в этом широком смысле образует необходимый компонент деятельности по разработке и использованию ГиИС, поскольку эта деятельность становится предметом осознания, обучения и рационализации другими учеными и специалистами. Методологическое знание выступает в форме как предписаний и норм, в которых фиксируются содержание и последовательность (организация) определенных видов деятельности, так и описаний фактически выполняемой деятельности. В обоих случаях основная функция этого знания — внутренняя организация и регулирование процесса познания и анализа сложных задач, синтеза и эксплуатации гибридов.

6.2. Жизненный цикл методологии

Разработку ГиИС принято называть гибридизацией [1, 74]. Это — трудоемкий процесс, требующий широкого спектра знаний о предметной области, задачах, методах их решения, длительной по времени, сложной обработки информации и экспериментов. Наш 20-летний опыт показывает, что на разработку «вручную» функциональной ГиИС, решающей сложную задачу, затрачивается 3 — 5 чел.-лет [249].

Выполняя гибридизацию, разработчик имеет дело с тремя основными субъективными сущностями: моделью сложной задачи (объектом-оригиналом), моделями нескольких методов решения задач (объектами-прототипами), моделью ГиИС (объектом-результатом). Первые два объекта — источники информации для гибридизации, а последний — основа направленного формообразования, изначально программирующая то, что должно появиться в конце, после трансформации информации.

Проблемно-структурная методология — учение о структуре, логической организации, принципах, методах и средствах деятельности разработчика, выполняющего гибридизацию, объект-оригинал которой — сложная задача, объекты-прототипы — методы из базисных классов, объект-результат — функциональные ГиИС.

Поскольку объект-оригинал — это сложная задача (англ. problem), а объект-результат — функциональные ГиИС, структура (англ. structure) которых зависит от состава и взаимосвязей подзадач из декомпозиции сложной, неоднородной задачи, методология и технология названы проблемно-структурными, сокращенно ПС (или PS).

Объект-оригинал в ПС — методологии представляется моделями — π^h «однородная задача» (5.7) и π^u — «неоднородная задача» (5.10), т.е. на микро - и макроуровне соответственно.

Объекты-прототипы имеют сильные и слабые стороны (табл. 1.12) наследуемые моделями решения задач-элементов (разд. 5.5). Методы (модели) должны быть так скомбинированы в объекте-результате (разд. 5.6 — 5.8), чтобы добиться резонансного, синергетического усиления возможностей элементов ГиИС по решению сложных, неоднородных задач. ПС-методология использует две модели объектов-прототипов: макроуровневую запись свойствами и микроуровневое представление метода как составного объекта, что позволяет гибко управлять гибридизацией по мере накопления разработчиком опыта решения однородных задач.

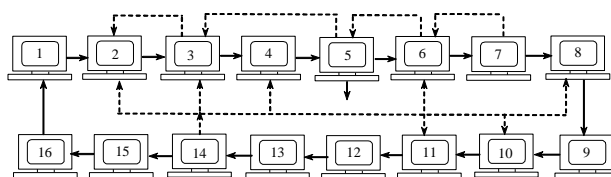
Объекты-результаты специфицированы в ПС-методологии через составные части-элементы, комбинируемые с помощью восьми классов

отношений интеграции (разд. 6.5), а также свойствами «вход», «выход», «состояние» и «гибридная стратегия».

В настоящее время на языке математики невозможно описать объекты ПС-методологии, а также преобразования объектов из одного представления в другое в ходе гибридизации. В этой связи в теории Te (5.1) построен информационный язык (разд. 7.2), на котором записаны эвристики-знания обо всех трех объектах и «жизненным циклом» представлена логическая последовательность (организация) действий разработчика (рис. 6.1).

Ниже поэтапно рассмотрено содержание деятельности по разработке функциональных ГиИС, представленной в их «жизненном цикле».

1-й этап «Идентификация сложной задачи». На этом этапе, относящемся к системному анализу задачи π^u , в соответствии со схемой ролевых концептуальных моделей (5.10), определяются цель, исходные данные для решения неоднородной задачи, ее идентификатор и спецификатор.



Обозначения: 1 — идентификация сложной задачи; 2 — редукция сложной задачи; 3 — спецификация подзадач; 4 — выбор методов; 5 — проверка неоднородности; 6 — разработка автономных моделей; 7 — спецификация областей релевантности моделей; 8 — спецификация связей сложных задач; 9 — разработка интерфейсов автономных моделей; 10 — разработка таблицы гибридных стратегий; 11 — выбор стратегии из таблицы гибридных стратегий; 12 — инициализация ГиИС по выбранной стратегии; 13 — машинные эксперименты и интерпретация результатов; 14 — формирование множества альтернатив; 15 — совершенствование множества альтернатив и анализ результатов; 16 — эксплуатация и модернизация

Рис. 6.1 — Проблемно-структурная методология решения сложных задач

2-й этап «Редукция сложной задачи и построение множества ее декомпозиций». На этом этапе продолжают работы по системному анализу неоднородной задачи. Однако акценты при этом смещаются на ее состав и структуру. Исходная задача-система редуцируется на множество подзадач (однородных задач) и строится множество возможных декомпозиций.

3-й этап «Спецификация подзадач». На этом этапе каждая однородная задача идентифицируется в соответствии со схемой ролевых концептуальных моделей (5.7). Определяются цель решения и исходные

данные, классификатор и частично спецификатор. Это позволяет построить знаки однородных задач в языке L_6^h (5.9) на рис. 7.5. Процедура спецификации — неформальная и выполняется разработчиком, который, прежде всего, должен повысить степень собственной уверенности в понимании задачи. Это можно сделать методами из разд. 6.5 и 6.6.

После того, как станет понятен смысл задачи и может быть сформулирована цель ${}^6G^h$, исходные данные ${}^2D^h$ и условия ${}^9C^h$, классификатор ${}^2K^h$ (5.7), необходимо задать идентификатор ${}^2O_1^h \in {}^2O^h$ (5.7), по которому ее можно будет отличить от других задач.

После формулирования целей, исходных данных, условий, классификаторов и идентификаторов задач необходимо определить остальные элементы спецификаторов ${}^2O^u$ и ${}^2O^h$. Рассмотрим эту процедуру на примере спецификаторов однородных задач ${}^2O^h$.

Спецификация выполняется экспертом в языке L_6^h (5.9) на следующих шести подмножествах характеристик: ${}^{22}X^0|DR$ — среды разработки; ${}^{22}X^0|PM$ — проблемной среды; ${}^{22}X^0|PR$ — проекта; ${}^{22}X^0|MI$ — измерений, действий и информации; ${}^{22}X^0|EX$ — среды экспертов и ЛПР и ${}^{22}X^0|PP$ — плана решения задачи. По сути, спецификация реализует отображение $\Psi^{\pi x}: {}^8X^n|h \rightarrow {}^{22}X^0$. Спецификация идет методом интервью. Полный список из около 100 специфицируемых характеристик и их возможных значений приведен в книге [1]. Спецификация заканчивается определением отношений ${}^{88}R_1^n$ и ${}^{88}R_2^n$ (5.7).

4-й этап «Выбор методов». Работы этого этапа выполняются для построенных на предыдущем этапе декомпозиций и специфицированных подзадач. Класс методов выбирает модельер в режиме консультаций с компьютером и с использованием системы с базой знаний. Метод установления соответствия объектов-оригиналов и объектов-прототипов рассмотрен в разд. 6.7.

5-й этап «Проверка неоднородности» выполняется, чтобы повысить у модельера степень уверенности в том, что сложная задача удовлетворяет модели «неоднородная задача» (5.10). Это оправдывает необходимость дальнейших работ по гибридизации и синтезу метода решения такой задачи. Заключение о неоднородности исходной задачи делает модельер на основании балльных оценок по выбору методов для каждой однородной задачи, входящей в декомпозицию исходной задачи. Если

для всех однородных задач оценки в пользу одного и того же класса базисных методов будут близки друг к другу, то можно с большой степенью уверенности предположить, что и сложная задача удовлетворяет модели «однородная задача». В этом случае работы следует продолжить в рамках одной из методологий автономного моделирования [1]. Если же в декомпозиции обнаружены области неоднородности, т.е. хотя бы для двух π^h будут выбраны различные классы методов, то это указывает на необходимость гибридизации (подробнее о проверке неоднородности см. разд. 6.6).

6-й этап «Разработка автономных моделей». На этом этапе в соответствии с методологиями элементов функциональных ГиИС [1] разрабатывается гетерогенное модельное поле \dot{M}^a (разд. 6.4), которое может быть и одноэлементным, для решения однородных задач из декомпозиции неоднородной задачи. Если такие модели уже имеются, то они также включаются в это множество. Работы, выполняемые на этом этапе, должны идти в строгом соответствии с методологиями автономных моделей и носят итерационный характер.

7-й этап «Спецификация областей релевантности моделей». Актуальность спецификации областей релевантности моделей объясняется альтернативностью (функциональной избыточностью) инструментариев решения одной и той же однородной задачи π^h . Кроме этого, на выбор альтернативы сильно влияют параметры внешней среды, например, для био-производственной системы [1] — это прогноз осадков и температуры, а также свойства ресурсов, например, плодородие почвы, учитываемые в описании состояния объекта управления. Такие параметры и характеристики названы существенными, а множества их значений — областями релевантности методов (моделей).

8-й этап «Спецификация связей сложных задач». На этом этапе заканчивается спецификация сложной задачи π^u с учетом обмена информацией при ее решении с другими сложными задачами. Для этого определяются отношения ${}^{88}R_4^n$ на множестве неоднородных задач Π^u (6.14).

9-й этап «Разработка интерфейсов автономных моделей». Интерфейсы играют значительную роль в гибридизации и, по существу, позволяют устанавливать отношения интеграции ${}^{\psi\phi}\ddot{R}^n$ (6.15) на гетерогенном модельном поле. Необходимость интерфейсов объясняется различиями языков описания ${}^2K^m$ в схемах ролевых концептуальных моделей базисных методов (табл. 5.1), с помощью которых построено гетерогенное

модельное поле. Для разработки интерфейсов используются адаптируемые программные утилиты, резко сокращающие трудозатраты на организацию межпрограммного обмена.

10-й этап «Разработка таблицы гибридных стратегий». Выполняется замена отношений декомпозиции ${}^{88}R_3^n$ (5.10) на отношения интеграции ${}^{\Psi\Phi}\ddot{R}^n$ (6.15) и устанавливаются отношения соответствия между исходными данными и целями неоднородной задачи, а также целями и исходными данными для однородных задач соответственно ${}^{88}R_4^n$ (5.10). Эти работы заканчиваются построением таблицы гибридных стратегий, отображающей перечисленные выше связи между однородными задачами π^h . Создание такой таблицы требует от разработчика априорных знаний о наилучших гибридных стратегиях. Такие знания, например, содержатся в табл. 8.6 [1] субъективной оценки полезности ди-гибридов [1]. В ПС-методологии эти знания учтены и доступны для разработчика.

11-й этап «Выбор стратегии из таблицы гибридных стратегий». На этом этапе экспертом выбирается стратегия из таблицы гибридных стратегий для синтеза функциональной ГиИС. Принятие решений на этом этапе отражает ситуацию, когда сложная, неоднородная задача имеет изменчивые состав и структуру и должна решаться всякий раз заново.

12-й этап «Инициализация ГиИС по выбранной стратегии (синтез метода решения сложной задачи)». На этом этапе, в режиме взаимодействия с пользователем, инициализируется функциональная ГиИС, интерпретация которой рассматривается как метод решения сложной задачи. Для синтеза архитектуры ГиИС разработаны эвристические алгоритмы (разд. 6.8). Их суть сводится к преобразованию гибридной стратегии, заданной матрицей «однородная задача — отношения интеграции — однородная задача», в список троек «модель — отношение интеграции — модель». Этот список, дополненный перечнем необходимых интерфейсных утилит, — исходная информация для инициализации архитектуры ГиИС.

13-й этап «Машинные эксперименты и интерпретация результатов». На этом этапе после инициализации ГиИС выполняются модельные эксперименты, в результате которых, в определенном декомпозицией π^u порядке, имитируется решение однородных задач и, в соответствии с отношениями ${}^{88}R_4^n$ для целей ${}^6G^u$ неоднородной задачи π^u и целей ${}^6G^h$ однородных задач $\pi^h \in \Pi^h$ (5.10), получается результирующая информация. Если результаты решения устраивают пользователя, то он

может и не выполнять последующие этапы. Однако предусмотрены и другие возможности.

14-й этап «Формирование множества альтернатив». Полученное на предыдущем этапе решение заносится в список альтернатив, после чего синтез повторяется с другими оценками пользователем альтернативных моделей из гетерогенного модельного поля. Продолжая такой итерационный процесс, можно сформировать множество альтернативных решений. Если оно содержит более 7—10, которые не могут быть обработаны пользователем вручную, то предусмотрен этап совершенствования.

15-й этап «Совершенствование множества альтернатив и анализ результатов». Здесь альтернативы рассматриваются как популяция индивидуумов, эволюция которой имитируется генетическим алгоритмом. Результаты анализируются пользователем и используются для принятия решений, спецификатор неоднородной задачи пополняется информацией об операции ${}^3X^n$ (5.10), реализующей принятую альтернативу. После реализации операции заполняется эвалюэтер — ${}^7O^n$ (5.10).

16-й этап «Эксплуатация и модернизация». Выполняется поддержка базы данных проекта в актуальном состоянии и баз знаний элементов функциональной ГиИС. В случае изменений в знаниях одного из элементов совершенствуется гетерогенное модельное поле.

Рассмотренную ПС-методологию отличает:

- 1) применение к практическим (неоднородным) задачам, которые подвергаются на первых этапах всестороннему и глубокому изучению с использованием методов системного анализа;
- 2) применение на всех этапах оригинального информационного языка функциональных ГиИС для представления объектов-оригиналов, объектов-прототипов и объектов-результатов гибридизации;
- 3) применение оригинального, основанного на знаниях метода подбора автономных методов для решения однородных задач π^h из декомпозиций неоднородной задачи π^n (разд. 6.7);
- 4) широкое применение схем ролевых концептуальных моделей, направляющих деятельность разработчика и позволяющих избежать в ходе проектирования функциональных ГиИС многочисленных ошибок;
- 5) разработка альтернативного гетерогенного модельного поля для π^n , в рамках которого строится метод решения сложной задачи; применение оригинального, эвристического алгоритма синтеза метода решения сложной задачи (разд. 6.8);
- 6) применение оригинального, эвристического алгоритма принятия решения о выборе моделей из гетерогенного модельного поля в соответствии с областями релевантности автономных методов (разд. 6.8).

6.3. Стратификация решения сложных задач в СППР

Решение сложных задач в системах поддержки принятия решений требуют многоуровневого, стратифицированного представления систем S (2.1). Выбор количества страт — одна из проблем стратификации. Если нижний предел очевиден — минимум две, то выбор верхнего — искусство разработчика. В работе [1] было обосновано, что верхний предел можно ограничить тремя стратами, если исходить из тесной связи страт и методов моделирования, классифицировав их (см. табл. 5.1) на аналитические, статистические и логико-лингвистические (символьные). Тогда формализованные и частично формализованные знания экспертов [1] могут быть упорядочены в эти классы.

Пусть есть система $S: {}^2X \rightarrow {}^2Y$, где ${}^2X = {}^2X_1 \times {}^2X_2 \times {}^2X_3$, ${}^2Y = {}^2Y_1 \times {}^2Y_2 \times {}^2Y_3$ — множества свойств «вход» и «выход» соответственно. Каждую пару $({}^2X_j, {}^2Y_j) | j=1,2,3$ припишем определенной страте $S_j | j=1,2,3$. Тогда можно ввести две модели и исследовать отношения «решение-решение» (рис. 7.3).

Модель \dot{m}_1^S . Страта $S_j | j=1,2,3$ представляется отображениями: $S_3: {}^2X_3 \times {}^3V_3 \rightarrow {}^2Y_3$, $S_2: {}^2X_2 \times {}^3B_2 \times {}^3V_2 \rightarrow {}^2Y_2$, $S_1: {}^2X_1 \times {}^3B_1 \rightarrow {}^2Y_1$, где ${}^3B_j, {}^3V_j$ — множества воздействий, исходящих от страт, примыкающих к S_j сверху и снизу соответственно.

Семейство $S_1 - S_2 - S_3$ назовем \dot{m}_1^S - стратификацией S , если существуют отображения $\Xi_j^1: {}^2Y_j \rightarrow {}^3V_{j+1} | j=1,2$ и $\Xi_j^2: {}^2Y_j \rightarrow {}^3B_{j-1} | j=2,3$, такие, что для каждого ${}^2x \in {}^2X$ и ${}^2y = S({}^2x)$ — ${}^2y_3 = S_3({}^2x_3, \Xi_2^1({}^2y_2))$, ${}^2y_2 = S_2({}^2x_2, \Xi_3^2({}^2y_3), \Xi_1^1({}^2y_1))$, ${}^2y_1 = S_1({}^2x_1, \Xi_2^2({}^2y_2))$. Будем называть семейство $S_1 - S_2 - S_3$ моделью заочных консультаций в СППР. Название модели объясняется тем, что эксперты находятся на своих рабочих местах, а не в зале совещаний. Задача π^u , которую должно решить ЛПР, возникшая на более высоком уровне абстрагирования, декомпозируется «вниз», а информация о решении экспертами подзадач передается «вверх». Такая модель присуща традиционным системам управления с преобладанием вертикальных связей.

Модель \dot{m}_2^S . Страта $S_j|j=1,\dots,4$ представляется отображениями: $\{S_4: {}^2X_4 \times {}^3V_j \rightarrow {}^2Y_4\}$, $\{S_j: {}^2X_j \times {}^3B_4 \rightarrow {}^2Y_j\}$, для $j=1,\dots,3$.

Семейство $S_1 - S_2 - S_3 - S_4$ назовем \dot{m}_2^S -стратификацией S , если существуют отображения $\Xi_j^1: {}^2Y_j \rightarrow {}^3V_4|j=1,2,3$ и $\Xi_4^2: {}^2Y_4 \rightarrow {}^3B_j|j=1,2,3$, такие, что для каждого ${}^2x \in {}^2X$ и ${}^2y = S({}^2x)$ — $y_4 = S_4({}^2x_4, \Xi_1^1(y_1), \Xi_2^1(y_2), \Xi_3^1(y_3))$, ${}^2y_j = S_j({}^2x_j, \Xi_4^2(y_4))|j=1,\dots,3$. Это модель очных консультаций в СППР, когда ЛПР «за круглым столом» формирует систему $S^u \subseteq S$ для решения задачи π^u , занимая в ней высший уровень S_4 относительно экспертных страт $S_j|j=1,\dots,3$, решающих однородные задачи Π^h .

Зададим множество языков профессиональной деятельности $L^p = \{L_1^p, \dots, L_{N_L}^p\}$ в S и соответствие $\Psi_1 \subseteq L^p \times \widehat{S} | \Psi_2 \neq \emptyset$, где $\widehat{S} = \{S_1, S_2, S_3\}$ для модели \dot{m}_1^S и $\widehat{S} = \{S_1, \dots, S_4\}$ для модели \dot{m}_2^S . Допускаем, что одной страте может соответствовать более одного $L_q^p \in L^p|q=1,\dots,N_L$.

Тогда неоднородной предметной областью назовем:

$$E^L = \langle \widehat{S}, L^p, \Psi_1 \rangle. \quad (6.1)$$

Пусть в E^L существуют неоднородные задачи $\Pi^u = \{\pi_1^u, \dots, \pi_{N_\Pi}^u\}$ и $\forall \pi_l^u \exists \Pi^h = \{\pi_1^h, \dots, \pi_{N_h}^h\}$, где $l=1,\dots,N_\Pi$, $\forall l (N_h = \text{var } y)$, $\pi_l^u \in \Pi^u$. Допустим, что π_l^u могут возникать только на $S_j|j=2,3$ и S_4 для \dot{m}_1^S - и \dot{m}_2^S -стратификаций соответственно. Зададим соответствия $\Psi_2 \subseteq \Pi^u \times \widehat{S}$ и $\Psi_3 \subseteq \widehat{\Pi}^h \times \widehat{S}$, где $\widehat{\Pi}^h = \bigcup_l^{N_\Pi} \Pi_l^h$, причем более чем одна π^h может принадлежать $S_j|j=1,\dots,3$ и для \dot{m}_2^S -стратификации $\forall \pi_4^u S_4 \in \Psi_2 \exists {}^1\widehat{\Pi}^h S_1, {}^2\widehat{\Pi}^h S_2, {}^3\widehat{\Pi}^h S_3 \in \Psi_3$, где ${}^1\widehat{\Pi}^h, \dots, {}^3\widehat{\Pi}^h \subseteq \widehat{\Pi}^h$, а для \dot{m}_1^S -стратификации — $\forall \pi_3^u S_3 \in \Psi_2 \exists {}^1\widehat{\Pi}^h S_1, {}^2\widehat{\Pi}^h S_2 \in \Psi_3$ и $\forall \pi_2^u S_2 \in \Psi_2 \exists {}^1\widehat{\Pi}^h S_1 \in \Psi_3$.

Тогда неоднородной проблемной средой назовем:

$$E^u = \langle E^L, \Pi^u, \hat{\Pi}^h, \Psi_2, \Psi_3 \rangle. \quad (6.2)$$

Сформулируем свойства неоднородной проблемной среды E^u : 1) в силу свойств E^L — это редуцированное, многоуровневое представление системы S , на каждой страте которого есть свои языки профессиональной деятельности и задачи; 2) задачи на страте S_4 , по определению, неоднородные (5.10), а на стратах $S_1 \div S_3$, по определению, — однородные (5.7); 3) в силу определения (5.7) методы решения π^h должны быть известны, что позволяет на каждой страте $S_1 \div S_3$ построить модели решения однородных задач; 4) в соответствии с гипотезой 4 (раздел 4.6) и в силу (5.10) методы решения π^u на страте S_4 неизвестны, что не позволяет на ней иметь априори заданные модели; 5) в силу свойств \hat{S} между стратами существуют двунаправленные потоки информации, координирующие решение однородных задач в составе неоднородной.

Пусть имеем неоднородную проблемную среду E^u (6.2), а также множество специалистов-управленцев из N экспертов в узких профессиональных областях и ЛПР $Iv = \{Iv_1, \dots, Iv_{N_{Iv}}, Iv_{N_{Iv}+1}^*\}$, где $Iv_1, \dots, Iv_{N_{Iv}}, Iv_{N_{Iv}+1}^*$ можно рассматривать, например, как модели личностей [244]. Существует соответствие $\Psi_4 \subseteq S \times Iv$, показывающее разную профессиональную специализацию участников, относящуюся к определенной страте. На S_4 находится элемент $Iv_{N_{Iv}+1}^*$, ответственный за принятие решений, а на $S_1 - S_3$ специалисты-эксперты $Iv_1, \dots, Iv_{N_{Iv}}$.

В ходе коллективной творческой деятельности, творческого процесса в системе поддержки принятия решений объект управления проходит состояния Z_q в результате выполнения над ними действий $\delta_q^* | q = 1, 2, \dots, N_\delta$, предпринимаемых ЛПР, решающим неоднородные задачи π_q^u , с учетом результатов $(\phi_q^j | j = 1, \dots, N_{Iv})$ решения (ϑ_q^j) задач $(\pi_q^h | j)$ экспертами. Таким образом, на нижнем уровне иерархии S^u вырабатываются множества результатов $\Phi_1 = \{\phi_1^1, \dots, \phi_1^{N_{Iv}}\}$,

$\Phi_2 = \{\varphi_2^1, \dots, \varphi_2^{N_h}\}, \dots, \Phi_{N_\delta} = \{\varphi_{N_\delta}^1, \dots, \varphi_{N_\delta}^{N_h}\}$, полученных в ходе множеств процессов решения $\Theta_1 = \{\vartheta_1^1, \dots, \vartheta_1^{N_h}\}, \Theta_2 = \{\vartheta_2^1, \dots, \vartheta_2^{N_h}\}, \dots, \Theta_{N_\delta} = \{\vartheta_{N_\delta}^1, \dots, \vartheta_{N_\delta}^{N_h}\}$, задач $\pi_q^h \in \Pi^h$. На верхнем уровне системы S^u , используя Φ_q , ЛПР вырабатывает результат φ_q^* , решения ϑ_q^* задачи π_q^u .

В соответствии с (6.2) между задачами π_q^u и π_q^h могут быть различные отношения. Например, редукции π_q^u на «профессиональные» образы $\pi_q^h|_1, \dots, \pi_q^h|_{N_h}$ (вариант 1) и более сложная редукция π_q^u в декомпозицию $\hat{\pi}_v^u = (\Pi^h, {}^{88}r^n)|_{{}^{88}r^n \in {}^{88}R_3^h}, \Pi^h = \{\pi_1^h, \dots, \pi_{N_\Pi}^h\}, v = 1, \dots, N_\delta$. Пусть это будет единственная декомпозиция по единственному отношению (вариант 2), а ЛПР знает способ установления взаимно однозначного соответствия $\Psi_5 : \Pi^h \times I_v^h | I_v^h \subseteq I_v, I_v^h = \{I_{v_1}, \dots, I_{v_{N_\Pi}}\}$, причем один эксперт решает одну подзадачу и $N_\Pi = N_{I_v}$. В результате формируется множество пар $\{\pi_q^h I_{v_j} | q = 1, \dots, N_\Pi, j = 1, \dots, N_{I_v}\}$, которые ниже будут записываться как $\pi_q^h|_j^v$, где v — номер декомпозиции. Во втором варианте деятельность ЛПР значительно усложняется, поскольку связана как с необходимостью редукции задачи (сохраняя в памяти или каким то другим образом декомпозицию), так и интеграции результатов — ответов экспертов.

Результаты $\{\varphi_q^*\}$ — план действий ЛПР. Тогда общая схема метода многоуровневого решения неоднородных задач выражается целенаправленной последовательностью шагов «решения Θ_q — решение ϑ_q^* — действие δ_q^* »:

$$(\Theta_1, \vartheta_1^*, \delta_1^*) \Rightarrow (\Theta_2, \vartheta_2^*, \delta_2^*) \Rightarrow \dots \Rightarrow (\Theta_{N_\delta}, \vartheta_{N_\delta}^*, \delta_{N_\delta}^*). \quad (6.3)$$

Очевидно, что последовательности шагов (6.3) могут соответствовать различные варианты решения задачи π_q^u , например, для варианта 1 имеем (символы « \Rightarrow » и « \rightarrow » обозначают переход от одного шага к другому и переход от решения одной задачи к другой соответственно;

открывающиеся фигурные скобки — редукция, закрывающиеся фигурные скобки — интеграция):

Для варианта 1 имеем:

$$\pi_1^u \left\{ \begin{array}{l} \pi_1^h | 1 \\ \pi_1^h | 2 \\ \vdots \\ \pi_1^h | N_{fj} \end{array} \right\} \varphi_1^*, \delta_1^* \Rightarrow \pi_2^u \left\{ \begin{array}{l} \pi_2^h | 1 \\ \pi_2^h | 2 \\ \vdots \\ \pi_2^h | N_{fj} \end{array} \right\} \varphi_2^*, \delta_2^* \Rightarrow \dots \Rightarrow \pi_{N_\delta}^u \left\{ \begin{array}{l} \pi_{N_\delta}^h | 1 \\ \pi_{N_\delta}^h | 2 \\ \vdots \\ \pi_{N_\delta}^h | N_{fj} \end{array} \right\} \varphi_{N_\delta}^*, \delta_{N_\delta}^*. \quad (6.4)$$

Для варианта 2 имеем:

$$\begin{array}{ccc} \tilde{\pi}_1^u & \tilde{\pi}_2^u & \tilde{\pi}_{N_\delta}^u \\ \left\{ \begin{array}{l} \pi_1^h | 1 \\ \pi_2^h | 2 \\ \vdots \\ \pi_{N_{\Pi}}^h | 1 \\ \pi_{N_{\Pi}}^h | N_{fj} \end{array} \right\} \varphi_1^*, \delta_1^* \Rightarrow \pi_2^u \left\{ \begin{array}{l} \pi_1^h | 2 \\ \pi_2^h | 2 \\ \vdots \\ \pi_{N_{\Pi}}^h | 2 \\ \pi_{N_{\Pi}}^h | N_{fj} \end{array} \right\} \varphi_2^*, \delta_2^* \Rightarrow \dots \Rightarrow \pi_{N_\delta}^u \left\{ \begin{array}{l} \pi_1^h | n \\ \pi_2^h | n \\ \vdots \\ \pi_{N_{\Pi}}^h | n \\ \pi_{N_{\Pi}}^h | N_{fj} \end{array} \right\} \varphi_{N_\delta}^*, \delta_{N_\delta}^*. & & (6.5) \end{array}$$

Если сравнивать варианты (6.4) и (6.5), то в (6.5), эксперты сами и естественным образом упрощают исходную неоднородную задачу, превращая ее в профессиональный эквивалент. Функции ЛПР при этом — интеграция и координация решений экспертов, включающая оценку, прогнозирование последствий, сравнение с собственным вариантом, выбор альтернативы и организация действий по ее выполнению. Это параллельная и быстродействующая схема. Она имеет место, когда ЛПР не могло заранее подготовиться к проведению мероприятия в силу чрезвычайности возникшей проблемы или занятости (режим чрезвычайных, непредвиденных, кризисных ситуаций). Вариант (6.4) характеризуется возможностью заочного участия экспертов, параллельностью решения подзадач и последовательным характером работы ЛПР. Процессы редукции и интеграции здесь значительно сложнее. Такая схема имеет место тогда, когда времени на выработку, оценку и выбор альтернатив достаточно. Такую схему можно использовать в режиме планового обсуждения проблемы (разд. 2.3).

Модели (6.4) и (6.5) позволяют сформулировать метод решения сложных задач в неоднородной проблемной среде (6.2). Его суть в том, что решение неоднородной задачи должно основываться на ее многостороннем анализе, относительно множеств узкопрофессиональных точек зрения. Это позволяет, используя редукцию, выявить в ней подобласти задач, которые могут быть поняты и решены с использованием зна-

ний специально подобранных экспертов, а также целенаправленно синтезировать результат решения проблемы в целом интеграцией профессиональных точек зрения. Иными словами, неоднородные задачи решаются применением гибридного интеллекта, для реализации которого требуется компьютерная технология гибридизации, поддерживающая анализ неоднородной задачи, моделирование деятельности экспертов и разработку функциональных ГИИС для имитации деятельности ЛПР.

Таким образом, если в качестве модели проблемной среды разработчик принимает (6.2), то это требует применения методов системного анализа неоднородных задач и синтеза методов, модели, алгоритмов и программ в условиях стохастичности системы S и внешней среды.

6.4. Классификация отношений интеграции знаний

В настоящем разделе модели миров задач (объектов-оригиналов) и моделирования (объектов-прототипов) рассматриваются во взаимодействии, что дает возможность определить отношения интеграции, построить их классификацию, специфицировать понятия интегрированного метода, модели и неоднородных рассуждений [1], используемые в проблемно-структурной технологии функциональных ГИИС.

Модель взаимодействия миров однородных задач W^{π^h} и автономных методов W^{m^a} изображена на рис. 6.2,а.

В верхней плоскости W^{π^h} окружностями обозначены однородные задачи (5.7) из декомпозиции Π^h (5.10) неоднородной задачи.

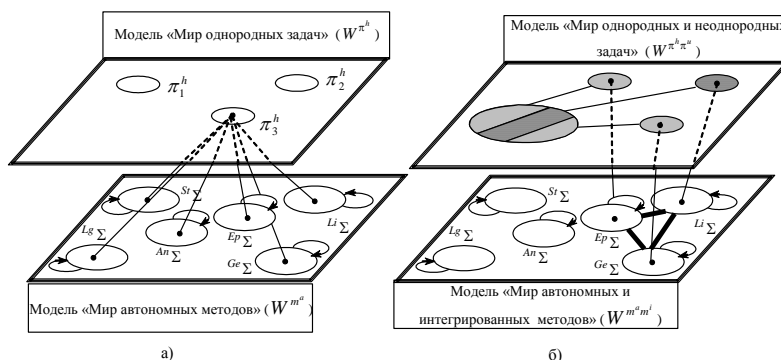


Рис. 6.2 — Взаимосвязи моделей миров задач и методов моделирования

Тогда модель мира однородных задач имеет вид:

$$W^{\pi^h} = \Pi^h, \quad (6.6)$$

где $\Pi^h = \{\pi_1^h, \dots, \pi_{N_h}^h\}$.

В нижней плоскости W^{m^a} также окружностями показаны шесть объектов $^{An}\Sigma, ^{St}\Sigma, ^{Lg}\Sigma, ^{Li}\Sigma, ^{Ge}\Sigma, ^{Ep}\Sigma$. Рассмотрим их подробнее, для чего введем шесть множеств: $M^a = \{^{An}M^a, ^{St}M^a, ^{Lg}M^a, ^{Li}M^a, ^{Ge}M^a, ^{Ep}M^a\}$, $^{An}M^a = \{m_1^a|^{An}, \dots, m_{N_{An}}^a|^{An}\}$, $^{St}M^a = \{m_1^a|^{St}, \dots, m_{N_{St}}^a|^{St}\}$, $^{Lg}M^a = \{m_1^a|^{Lg}, \dots, m_{N_{Lg}}^a|^{Lg}\}$, $^{Li}M^a = \{m_1^a|^{Li}, \dots, m_{N_{Li}}^a|^{Li}\}$, $^{Ge}M^a = \{m_1^a|^{Ge}, \dots, m_{N_{Ge}}^a|^{Ge}\}$, $^{Ep}M^a = \{m_1^a|^{Ep}, \dots, m_{N_{Ep}}^a|^{Ep}\}$ работы с An -, St -, Lg -, Li -, Ge - знаниями и Ep - опытом на $S_3, S_2, S_1[1]$, где $N_{An}, N_{St}, N_{Lg}, N_{Li}, N_{Ge}, N_{Ep}$ — число известных (разработанных) m^a -методов. Кроме этого, введем еще шесть множеств, элементы которых — автономные модели:

$$^{An}\dot{M}^a = \{\dot{m}_1^a|^{An}, \dots, \dot{m}_1^a|^{An}_{N_1}, \dots, \dot{m}_{N_{An}}^a|^{An}, \dots, \dot{m}_{N_{An}}^a|^{An}_{N_2}\}, \quad (6.7)$$

$$^{St}\dot{M}^a = \{\dot{m}_1^a|^{St}, \dots, \dot{m}_1^a|^{St}_{N_3}, \dots, \dot{m}_{N_{St}}^a|^{St}, \dots, \dot{m}_{N_{St}}^a|^{St}_{N_4}\}, \quad (6.8)$$

$$^{Lg}\dot{M}^a = \{\dot{m}_1^a|^{Lg}, \dots, \dot{m}_1^a|^{Lg}_{N_5}, \dots, \dot{m}_{N_{Lg}}^a|^{Lg}, \dots, \dot{m}_{N_{Lg}}^a|^{Lg}_{N_6}\}, \quad (6.9)$$

$$^{Li}\dot{M}^a = \{\dot{m}_1^a|^{Li}, \dots, \dot{m}_1^a|^{Li}_{N_7}, \dots, \dot{m}_{N_{Li}}^a|^{Li}, \dots, \dot{m}_{N_{Li}}^a|^{Li}_{N_8}\}, \quad (6.10)$$

$$^{Ge}\dot{M}^a = \{\dot{m}_1^a|^{Ge}, \dots, \dot{m}_1^a|^{Ge}_{N_9}, \dots, \dot{m}_{N_{Ge}}^a|^{Ge}, \dots, \dot{m}_{N_{Ge}}^a|^{Ge}_{N_{10}}\}, \quad (6.11)$$

$$^{Ep}\dot{M}^a = \{\dot{m}_1^a|^{Ep}, \dots, \dot{m}_1^a|^{Ep}_{N_{11}}, \dots, \dot{m}_{N_{Ep}}^a|^{Ep}, \dots, \dot{m}_{N_{Ep}}^a|^{Ep}_{N_{12}}\}, \quad (6.12)$$

где N_2, N_4, \dots, N_{12} — количества моделей, построенных для решения задач из Π^h ; например, $\dot{m}_j^a|_k^{An}$ читается так: «автономная (a) модель, полученная применением j -го метода работы с An -знаниями для решения k -й задачи из Π^h »; при этом будем полагать, что $\forall \pi_k^h | k=1, \dots, N_h, \exists m^a \in \dot{M}^a = \{^{An}\dot{M}^a, ^{St}\dot{M}^a, ^{Lg}\dot{M}^a, ^{Li}\dot{M}^a, ^{Ge}\dot{M}^a, ^{Ep}\dot{M}^a\}$,

причем $\dot{M}^a \neq \emptyset$ и чаще всего $\Psi \dot{M}^a \cap \Phi \dot{M}^{aa} = \emptyset$, где $\Psi, \Phi = \{An, St, Lg, Li, Ge, Ep\}$, $\Psi \neq \Phi$.

Будем называть $^{An}\dot{M}^a, ^{St}\dot{M}^a, ^{Lg}\dot{M}^a, ^{Li}\dot{M}^a, ^{Ge}\dot{M}^a, ^{Ep}\dot{M}^a$ гомогенными (однородными) модельными полями, а \dot{M}^a — гетерогенным (неоднородным) модельным полем.

Кроме введенных 12 множеств, существуют множества алгоритмов A и программ P , интерпретирующих модели \dot{m}^a на ЭВМ. Тогда $^{An}\Sigma = \langle ^{An}M^a, ^{An}\dot{M}^a, ^{An}A^a, ^{An}P^a \rangle$, $^{St}\Sigma = \langle ^{St}M^a, ^{St}\dot{M}^a, ^{St}A^a, ^{St}P^a \rangle$, $^{Lg}\Sigma = \langle ^{Lg}M^a, ^{Lg}\dot{M}^a, ^{Lg}A^a, ^{Lg}P^a \rangle$, $^{Li}\Sigma = \langle ^{Li}M^a, ^{Li}\dot{M}^a, ^{Li}A^a, ^{Li}P^a \rangle$ и $^{Ge}\Sigma = \langle ^{Ge}M^a, ^{Ge}\dot{M}^a, ^{Ge}A^a, ^{Ge}P^a \rangle$, $^{Ep}\Sigma = \langle ^{Ep}M^a, ^{Ep}\dot{M}^a, ^{Ep}A^a, ^{Ep}P^a \rangle$. Все эти объекты — носители знаний. В дальнейших рассуждениях ограничимся рассмотрением только методов и моделей.

Петлями на рис. 6.2,а обозначены отношения ${}^{\Psi\Phi}\dot{R}^n | \Psi, \Phi = \{An, St, Lg, Li, Ge\}, \Psi = \Phi$ получения новых знаний в автономных нишах, т.е. в рамках ограничений определенного вида знаний. Эти отношения — предмет исследования научных школ, исповедующих тот или иной вид знаний.

Тогда модель W^{m^a} имеет следующий вид:

$$W^{m^a} = \langle \widehat{\Sigma}, \widehat{S}, \Psi^{\Sigma S}, {}^{\Psi\Phi}\dot{R}^n \rangle, \quad (6.12)$$

где $\widehat{\Sigma} = \{^{An}\Sigma, ^{St}\Sigma, ^{Lg}\Sigma, ^{Li}\Sigma, ^{Ge}\Sigma, ^{Ep}\Sigma, \}$; $W^{\pi^h} = \Pi^h$, для модели \dot{m}_1^S (разд. 6.3) и $\widehat{S} = \{S_1, S_2, S_3, S_4\}$ для модели \dot{m}_2^S (разд. 6.3); $\Psi^{\Sigma S} \subseteq \widehat{\Sigma} \times \widehat{S}$ — всюду определенное, сюръективное соответствие (в случае \dot{m}_2^S -модели — взаимнооднозначное); $\dot{R}^n = \{^{AnAn}\dot{R}^n, ^{StSt}\dot{R}^n, ^{LgLg}\dot{R}^n, ^{LiLi}\dot{R}^n, ^{GeGe}\dot{R}^n, ^{EpEp}\dot{R}^n\}$, $^{AnAn}\dot{R}^n : ^{An}\Sigma \rightarrow ^{An}\Sigma$, $^{StSt}\dot{R}^n : ^{St}\Sigma \rightarrow ^{St}\Sigma$, $^{LgLg}\dot{R}^n : ^{Lg}\Sigma \rightarrow ^{Lg}\Sigma$, $^{LiLi}\dot{R}^n : ^{Li}\Sigma \rightarrow ^{Li}\Sigma$, $^{GeGe}\dot{R}^n : ^{Ge}\Sigma \rightarrow ^{Ge}\Sigma$, $^{EpEp}\dot{R}^n : ^{Ep}\Sigma \rightarrow ^{Ep}\Sigma$.

Тогда модель взаимодействия W^{m^a} и W^{π^h} есть:

$$W^{\pi^h m^a} = \langle W^{\pi^h}, W^{m^a}, \Psi^{\pi m} |_{ha} \rangle, \quad (6.13)$$

где $\Psi^{\pi m}|_{ha} = \{\Psi_1^{\pi m}|_{ha}, \dots, \Psi_6^{\pi m}|_{ha}\}$, $\Psi_1^{\pi m}|_{ha} \subseteq \Pi^h \times^{An} \dot{M}^a$, $\Psi_2^{\pi m}|_{ha} \subseteq \Pi^h \times^{St} \dot{M}^a$, $\Psi_3^{\pi m}|_{ha} \subseteq \Pi^h \times^{Lg} \dot{M}^a$, $\Psi_4^{\pi m}|_{ha} \subseteq \Pi^h \times^{Li} \dot{M}^a$, $\Psi_5^{\pi m}|_{ha} \subseteq \Pi^h \times^{Ge} \dot{M}^a$ и $\Psi_6^{\pi m}|_{ha} \subseteq \Pi^h \times^{Ep} \dot{M}^a$ — семейство всюду определенных, сюръективных соответствий.

Таким образом, два мира W^{π^h} (6.6) задач, решаемых в СППР экспертами, и W^{m^a} (6.13) методов имитации рассуждений экспертов связаны отношениями соответствия задачи π^h и метода ее решения m^a (может быть и более одного). В результате установления $\Psi^{\pi m}$, т.е. применения автономных методов к решению однородных задач в W^{m^a} , возникают эволюционные процессы и устанавливаются отношения ${}^{\Psi\Phi} \dot{R}^n | \Psi = \Phi$, т.е. отношения «автономный метод — автономный метод», и в результате деятельности модельера в W^{m^a} появляются новые объекты m^a (и далее \dot{m}^a , a^a и p^a). Очевидно, что модели $\{\dot{m}^a\}$, алгоритмы $\{a^a\}$ и программы $\{p^a\}$ также эволюционируют во времени.

Модель (6.13) дает упрощенное, несистемное представление СППР как мира экспертов на стратах \hat{S} и раскрывает применение автономных методов к неоднородной предметной области (6.1).

Релевантная сложности процессов в СППР картина дана на рис. 6.2,б. В верхней плоскости здесь изображена уже неоднородная проблемная среда (6.2) в виде модели мира однородных и неоднородных задач:

$$W^{\pi^h \pi^u} = \langle W^{\pi^h}, {}^s W^u \rangle. \quad (6.14)$$

где ${}^s W^u = \langle \Pi^u, {}^{88} R^n \rangle$ — модель структуры мира задач; $\Pi^u = \{\pi_1^u, \dots, \pi_{N_{\Pi}}^u\}$ — непустое множество π^u -задач; ${}^{88} R_n^n$ — множество отношений на Π^u , например, «часть-целое», «класс-подкласс», и др.

В ответ на редукцию сложной, неоднородной задачи π^u из $W^{\pi^h \pi^u}$ в модели $W^{m^a m^i}$ возникает соответствующий «системный отклик», образующий структуры знаний (методы, модели, алгоритмы и программы) за рамками автономного метода m^a . Интеграция в этом случае означает возникновение отношений (показаны на рис. 6.2 жирными линиями):

$$\Psi^{\Phi} \ddot{R}^n | \Psi, \Phi = \{An, St, Lg, Li, Ge, Ep\}, \Psi \neq \Phi \quad (6.15)$$

между разнородными знаниями и формирование m^i, \dot{m}^i, a^i, p^i , интерпретирующих такие отношения. Таким образом, в модели $W^{m^a m^i}$ нет и не может быть заранее известной структуры знаний, адекватной π^u . Такая структура является искомой и должна быть построена.

Сравнительный анализ рис. 6.2, а, б показывает, что вариант б) более общий и не отрицает важность отношений $\Psi^{\Phi} \dot{R}^n | \Psi = \Phi$, а дает возможность их улучшить, преодолеть минусы методов и моделей m^a, \dot{m}^a за счет отношений интеграции с другими методами и моделями, лишенных таких недостатков.

Выделение и классификация отношений $\Psi^{\Phi} \ddot{R}^n$ имеют большое значение в работе с интегрированными и распределенными знаниями, гибридным интеллектом, для создания интегрированных, междисциплинарных методов и функциональных ГиИС в проблемно-структурной технологии. Такая важность объясняется тем, что становится возможной, прежде всего, систематизация накопленного в этой области опыта за счет выделения и исследования типовых схем за пределами узких и «тесных» рамок автономных методов. Появляется возможность системных исследований свойств и возможностей таких схем, сравнительного анализа их достоинств и недостатков. Наконец, открывается путь для разработки технологий, создающих, управляющих и поддерживающих такие схемы в системах искусственного интеллекта.

На рис. 6.3 изображен один из возможных вариантов классификации отношений $\Psi^{\Phi} \ddot{R}^n$ [1].

Установление в $W^{m^a m^i}$ отношений извлечения $\Psi^{\Phi} \ddot{R}^n |^1$ («откуда извлекается», «что извлекается», «куда извлекается», «как извлекается», «чем извлекается» и др.) между m^a, \dot{m}^a , которые в разд. 5.5 названы элементами (компонентами), позволяет создавать интегрированные системы с целью выделения знаний из одного элемента для совершенствования или приобретения знаний другим элементом. Пример — извлечение неконтрастных знаний из искусственных нейронных сетей и использование знаний в ЭС и нечетких системах.

Установление отношений включения $\Psi^{\Phi} \ddot{R}^n |^2$ (целое-часть, иметь в составе, использовать, внести и др.) знаний одного элемента в знания

другого элемента дает возможность строить интегрированные системы с целью применения, чаще *An-*, *St-*, *Lg-* знаний, в *Li-*знаниях и алгоритмах манипулирования ими. Так, например, такие отношения актуальны в нечетких системах с машиной вывода Takagi-Sugeno [1], когда правая часть правил «условие-действие» — это полином первого порядка. Логические знания в виде правил вывода «модус-поненс» и «модус-толленс» встраиваются в управление выводом экспертных систем.

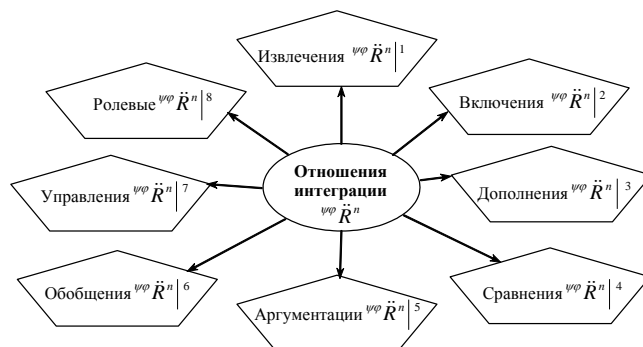


Рис. 6.3 — Классификация отношений интеграции знаний

Установление отношений дополнения $\psi\phi\ddot{R}^n|3$ (*последовательность, иерархия, сеть* и др.) одних знаний другими, когда на знаниях задан порядок, позволяет строить интегрированные системы, решающие сложные задачи, в которых знания одного вида помогают подготовить знания или данные для того, чтобы они могли использоваться другими элементами. Так, *Li* - знания из экспертных систем применяются для генерации множества альтернатив, оптимальные из которых отбираются с помощью *Ge* -знаний в генетических алгоритмах.

Установление отношений сравнения $\psi\phi\ddot{R}^n|4$ (*лучше, хуже, «полнота, абсурдность, противоречивость, похожесть, аналогичность, новизна, корректность, надежность* и др.) разнородных знаний позволяют строить интегрированные системы для анализа вводимой информации с целью модификации, перестройки имеющихся в интеллектуальных системах знаний. Отношения сравнения дают возможность заменять одни виды знаний другими в ходе рассуждений.

Установление отношений аргументации $\psi\phi\ddot{R}^n|5$ (*оправдание, объяснение, обоснование* и др.) результатов использования знаний одного

элемента с помощью знаний другого элемента позволяет строить интегрированные системы, повышающие у субъекта управления степень прозрачности рассуждений компьютера и, как следствие, увеличить доверие к его ответам. Отношения оправдания устанавливают соответствие или несоответствие решения элемента с одной ценностной структурой знаний решению другого элемента. Отношения объяснения позволяют, например, для непрозрачных нейросетевых вычислений дать объяснения с использованием лингвистических знаний в экспертных системах. По отношениям обоснования можно проверить, не противоречат ли решения одного элемента знаниям в других элементах. Отношения аргументации хорошо исследованы для лингвистических знаний, однако в интегрированных системах пока развиты недостаточно.

Установление отношений обобщения $\Psi^{\Phi} \ddot{R}^n$ ⁶ (*агрегирование, компиляция, трансформация* и др.) совокупности знаний одного элемента в знания другого элемента позволяет строить интегрированные системы с иерархией уровней абстрагирования, например, страт. В этом случае знания с меньшим уровнем абстракции отображаются, транслируются на другой уровень, где формируются знания менее детализированные, более абстрактные, чем имелись на ниже расположенном уровне. Известны примеры трансформации дифференциальных уравнений или детерминированных конечных автоматов в искусственные нейронные сети, а также трансформации нейросети в иерархии понятий.

Установление отношений управления $\Psi^{\Phi} \ddot{R}^n$ ⁷ (*«метапроцессия», «препроцессия», «постпроцессия», «сопроцессия»* и др.), когда знания одного элемента активируют знания других элементов, позволяет строить интегрированные системы для решения неоднородных задач, в которых декомпозиция на однородных задачах изменчива во времени, динамична. Например, *Li*-знания в продукционной экспертной системе могут выполнять функции метапроцессора, который вызывает в зависимости от ситуации в объекте управления порядке *An*-, *St*- и другие *Li*-знания для решения однородных задач.

Установление ролевых $\Psi^{\Phi} \ddot{R}^n$ ⁸ отношений между разными видами знаний позволяет строить интегрированные системы — хамелеоны, которые, имея структуру, присущую одному виду знаний, функционируют как системы, использующие другой вид знаний. Эти отношения похожи на отношения между артистом и героем, роль которого артист исполняет. Применяя эти отношения, невозможно структурно отделить один элемент от другого. Одна и та же структура используется для представ-

ления и манипулирования различными видами знаний. Характерный пример ролевых отношений — нейро-нечеткие системы. В них нечеткая система функционирует как искусственная нейросеть [1, 157, 159].

Рассмотренные классы отношений интеграции позволяют ввести модель мира автономных и интегрированных методов $W^{m^a m^i}$ на рис. 6.2,б, которая имеет следующий вид:

$$W^{m^a m^i} = \langle W^a, {}^{\psi\phi}\dot{R}^n, {}^{\psi\phi}\ddot{R}^n \rangle, \quad (6.16)$$

где ${}^{\psi\phi}\ddot{R}^n : {}^{\psi}\Sigma \rightarrow {}^{\phi}\Sigma, \psi \neq \phi$.

В модели $W^{m^a m^i}$ определим несколько понятий.

Под интегрированным методом будем понимать подмножество отношений ${}^{\psi\phi}\ddot{R}_\omega^n \subseteq {}^{\psi\phi}\ddot{R}^n$, заданное на множестве \tilde{M}^a микроуровневых концептуальных моделей методов, построенных по схемам \tilde{m}^a (5.13) (в настоящей работе не рассматривается), включающем *min* два элемента $\tilde{m}^a|_{\psi} \in {}^{\psi}\tilde{M}^a, \tilde{m}^a|_{\phi} \in {}^{\phi}\tilde{M}^a; {}^{\psi}\tilde{M}^a, {}^{\phi}\tilde{M}^a \subset \tilde{M}^a$, причем $\psi \neq \phi$, т.е.

$$m^i = \langle \tilde{M}^a, {}^{\psi\phi}\ddot{R}_\omega^n \rangle. \quad (6.17)$$

Под интегрированной моделью будем понимать некоторое подмножество отношений ${}^{\psi\phi}\ddot{R}_\varepsilon^n \subseteq {}^{\psi\phi}\ddot{R}^n$, заданное на множестве автономных

моделей $\bigcup_{q=\{An, St, Lg, Li, Ge, Ep\}} {}^q\dot{M}^a$, содержащем *min* две модели $\dot{m}^a|_{\psi} \in {}^{\psi}\dot{M}^a, \dot{m}^a|_{\phi} \in {}^{\phi}\dot{M}^a, q, \psi, \phi \in \{An, St, Lg, Li, Ge, Ep\}$ причем $\psi \neq \phi$, т.е.

$$\dot{m}^i = \langle \bigcup_{q=\{An, St, Lg, Li, Ge, Ep\}} {}^q\dot{M}^a, {}^{\psi\phi}\ddot{R}_\varepsilon^n \rangle. \quad (6.18)$$

Из (6.17) следует, что интегрированный метод — это метод-система, построенная с помощью отношений интеграции на множестве микроуровневых концептуальных моделей других методов, а из (6.18) то, что интегрированная модель — это модель-система, построенная с помощью отношений интеграции на множестве других моделей. Интерпретацию модели (6.18), разработанной в ответ на редукцию задачи π^u в мире $W^{m^a m^i}$, будем называть методом решения неоднородной задачи.

Неоднородными будем называть рассуждения с использованием интегрированных методов и (или) моделей с целью решения сложных, неоднородных задач.

Чтобы построить модель взаимодействия $W^{\pi^h \pi^u}$ (6.14) и $W^{m^a m^i}$ (6.16), необходимо, по аналогии с (1.8), установить соответствия $\Psi^{\pi^m} \Big|_{m^i}$, но уже в условиях плюрализма. Это выражается наличием в $W^{m^a m^i}$ гетерогенного модельного поля \dot{M}^a , т.е. когда: 1) однородная задача может быть решена несколькими методами и требуется подобрать инструментарий, релевантный спецификации (5.7); 2) для решения одной однородной задачи в \dot{M}^a может существовать более одной модели и при построении интегрированной модели (6.18) нужно делать выбор. Таким образом, актуальна разработка двух процедур.

Актуальность первой процедуры объясняется отсутствием в настоящее время аналитического решения задачи подбора метода, релевантного спецификации однородной задачи (5.7). В этой связи создан метод, основанный на знаниях, автоматизированного выбора классов базисных автономных методов для решения однородной задачи (разд. 6.7), разработана и протестирована система с базой знаний KB^m , на вход которой поступает спецификация однородной задачи, а на выходе выдаются балльные оценки отнесения задачи к классам базисных методов.

Актуальность второй процедуры объясняется альтернативностью инструментариев решения одной и той же однородной задачи π^h . Исследования био-производственной системы [1] подтвердили: 1) сам факт наличия альтернативности в мире моделирования; 2) сильное влияние на выбор альтернативы параметров внешней среды, например прогноза осадков и температуры, а также свойств ресурсов, например, плодородия почвы, учитываемых в описании состояния объекта управления. Такие параметры и характеристики названы существенными, а множества их значений — областями релевантности методов (моделей).

В этой связи и возникает необходимость принятия решений по выбору альтернативы при синтезе метода моделирования для неоднородной задачи. Пусть имеются релевантное однородной задаче π_j^h множество инструментариев $\{m_i^a\}$, информационный язык L^S (7.2) и группа экспертов. Требуется выделить на множестве значений ${}^4\bar{X}^0 \subseteq {}^4X^0$ свойств в ${}^2\bar{X}^0 \subseteq {}^2X^0$, включенных в описание ситуаций (состояний) объекта управления с учетом воздействий внешней среды,

${}^{44}x_v^j = [{}^4x_{v_{\min}}^0, {}^4x_{v_{\max}}^0] \in {}^4\bar{X}^0 \times {}^4\bar{X}^0 \mid {}^4x_{v_{\min}}^0, {}^4x_{v_{\max}}^0 \in {}^4\bar{X}^0, v=1, \dots, N_{\bar{X}},$ значения свойств ${}^2\bar{x}^0 \in {}^2\bar{X}^0$, релевантные ограничения инструментария \dot{m}_l^a .

Результаты исследований био-производственной системы [1] показали, что эксперты располагают знаниями, достаточными для решения задачи спецификации областей релевантности. Эти знания могут быть использованы в системе с базой знаний — $KB^{\dot{m}x}$ для принятия решений о выборе альтернативной модели.

Тогда может быть задана модель взаимодействия $W^{\pi^h \pi^u}$ и $W^{m^a m^i}$:

$$W^{\pi^h \pi^u m^a m^i} = \langle W^{\pi^h \pi^u}, W^{m^a m^i}, \Psi^{\pi m} \mid_{ui}, \Psi^{\dot{m}x}, KB^{\pi m}, KB^{\dot{m}x} \rangle, \quad (6.19)$$

где $\Psi^{\pi m} \mid_{ui}$ — соответствие, задающее множество пар «задача-методы» $\{\pi_1^h \{m_l^a\}_1^h, \dots, \pi_{N_h}^h \{m_l^a\}_{N_h}^h\}$; $\Psi^{\dot{m}x}$ — соответствие, задающее $\{\dot{m}_1^a \{{}^{44}x_1^1, \dots, {}^{44}x_{N_{\bar{X}}}^1\}, \dots, \dot{m}_{N_m}^a \{{}^{44}x_1^{N_m}, \dots, {}^{44}x_{N_{\bar{X}}}^{N_m}\}\}$ — множество пар «модель - интервалы» по каждой из N_m моделей.

Модель (6.19) устраняет терминологические противоречия, объясняет неудачные попытки моделирования сложных задач. В этих попытках «вырезаются» отдельные части из неоднородной задачи π^u , которые могут быть имитированы в рамках автономного метода. Такие упрощенные модели π^u , когда не учтена структура задачи (а точнее, не решена исходная π^u), после «успешных» лабораторных испытаний идут в практику КСППР без всяких перспектив быть использованными. СППР вынуждена по-прежнему решать общую (целую) задачу π^u и компьютерная технология «не вписывается» в этот творческий процесс. Наконец, (6.19) обладает гибкостью, открыта для модификации W^{π^h} (добавление и удаление π^h) и ${}^S W^u$ (добавление и удаление π^u). В нее включены знания «задача-методы» и «метод-интервалы», что делает (6.19) мощной эвристикой синтеза методов решения неоднородных задач.

6.5. Методы выяснения причин возникновения сложных проблем управления и обработки информации

Первый этап ПС-методологии — «Идентификация сложной задачи». При этом действия разработчика направляются схемой (5.7) в сочетании с методами, рассматриваемыми в данном разделе.

Метод «брита Оккама» [22]. Труды В. Оккама лежат в основе современных теорий управления. В. Оккам прославился своим афоризмом: «*Entia non multiplicanda praeter necessitatem*», что означает: «Сущности не следует умножать без необходимости». Это утверждение, получившее название «брита Оккама» (англ. *occam's razor*), подразумевает, что все явления, по возможности, должны быть представлены в простом и упорядоченном виде.

Метод состоит из трех шагов.

1. Если перед СППР стоит несколько задач, то выбираются задачи для работы экспертов и требуется ответить на следующие вопросы: а) Может ли проблема быть решенной в организации непосредственно? Отбрасываются проблемы, на которые невозможно повлиять; б) Возможно ли что-либо сделать по проблеме за приемлемый период времени? Рассматриваются проблемы, для решения которых потребуются недели или месяцы, но никак не годы; в) Можно ли собрать информацию по проблеме? Важно, чтобы решение основывалось на фактах, а не на мнениях; г) Действительно ли есть желание решить проблему? Учитывается желание каждого эксперта активно работать над решением проблемы.

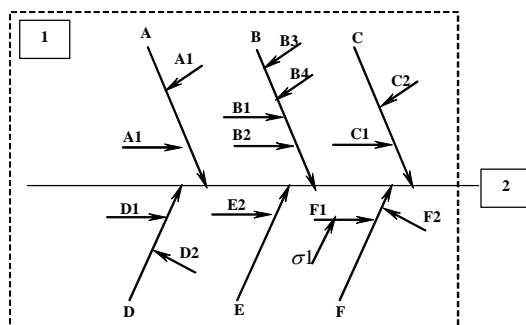
2. При идентификации сложной задачи уделяется внимание определению проблемы. Этот шаг включает в себя следующие действия: а) сформулированная проблема отображается так, чтобы ее видел каждый эксперт; б) эксперты тщательно обдумывают формулировку проблемы. Им следует объяснить, что часто правильная формулировка проблемы содержит и пути ее решения; г) эксперты передают свои решения ЛПР, и оно группирует ответы по темам, причем не исключая повторяющиеся результаты; д) эксперты обсуждают существо реальной проблемы, над которой предстоит работать в дальнейшем. Участники должны обсудить все возможности и прийти к консенсусу. Далее вырабатываются представления членов группы о том, что они хотят получить в результате решения проблемы. Процесс осуществляется аналогично п. 2.

3. Эксперты обсуждают вопрос: стоит ли тратить время на данную проблему? Если проблема заслуживает дальнейшего рассмотрения, то ее полная формулировка должна быть отображена на табло для напоминания экспертам о точной цели их работы.

Методы анализа задач. Ниже рассматриваются два наиболее распространенных метода анализа.

Диаграмма «рыбьи кости» или диаграмма К. Исикавы [22]. Диаграмма Исикавы в законченном виде напоминает скелет рыбы (рис. 6.4). Диаграмма «рыбьи кости» предназначается для отделения причин от следствий, помогает увидеть проблему целиком и удачно сочетает аналитическое и творческое мышления.

Как показано на рис. 6.4, следствие — «хребет» этого скелета и одновременно результат действия различных причин (факторов): *A, B, C* и т. д. На рисунке они обозначены стрелками — «большими костями». Эти причины, в свою очередь, — следствие других причин: *A1, A2,*



Обозначения: 1 — система причинных факторов; 2 — следствие (формулировка проблемы); *A, B, ...* — главные причины (или причины 1-го уровня); *A1, B1, ...* — причины 2-го уровня; *A2, B2, ...* — причины 3-го уровня и т. д.

Рис. 6.4 — Диаграмма К. Исикавы с разделением причин по уровням

...(для следствия *A*); *B1, B2, ...* (для следствия *B*), обозначенных стрелками («средние кости»). Вторичным причинам могут соответствовать третичные причины и т.д. Диаграмма строится за шесть шагов.

1. ЛПР перед началом работы убеждается в том, что все эксперты согласны с формулировкой проблемы. Затем в центре правой части табло в рамке записывается форму-

лировка проблемы и к ней проводится по центру экрана прямая линия (хребет рыбы).

2. Далее обозначаются главные уровни (ребра рыбы) — общие причины, влияющие на проблему. Если трудно определить главные ветви, то используются самые общие заголовки. Например, персонал: квалификация, опыт работы, обучение, психологический климат и т. д.

3. Подготовка и сессия мозгового штурма [22] для сортировки всех возможных и существующих причин проблемы по каждой категории главного уровня (ребра рыбы). Эксперты высказываются по всем заголовкам сразу. Допускается повторение одинаковых вариантов следующих уровней в нескольких местах (областях главных ребер), если эксперты предполагают прямую и многостороннюю связь.

4. На заседании СППР обсуждается, чем вызываются причины, записанные под заголовками главных уровней. Для этого эксперты отвечают на вопрос: «Что могло бы вызвать проблемы в данных областях?» Таким образом, анализ углубляется до максимально возможного уровня. Важно указать реальные связи между различными причинами. Если трудно построить правильную диаграмму сразу, она изучается, и причины группируются по-другому.

5. Обдумывание (сразу или через несколько дней) помогает обеспечить завершенность диаграммы. При этом диаграмма размещается на табло, где все участники смогут ее увидеть и вносить в нее добавления. С учетом всех мнений эксперты могут изменить первичную диаграмму.

6. Анализ диаграммы по принципу Парето [22]. Для этого поочередно рассматриваются причины под каждым из главных уровней для выявления выходящих за рамки обсуждаемой проблемы. Далее эксперты определяют причины, наиболее сильно влияющие на проблему. В данном случае эксперты еще не располагают полной информацией о проблеме, и они могут опираться на свой опыт и знания, чтобы определить области фокусирования своего внимания.

Диаграмма шести слов [22]. Изучить причины проблемы можно также анализом, заключающимся в постановке специальных вопросов. Данный метод иногда называется 5W1H по первым английским буквам слов, составляющих вопросы: Why? (Почему?); What? (Что?); When? (Когда?); Where? (Где?); Who? (Кто?) и How? (Как?).

Заседание СППР идет по следующему сценарию.

1. Экспертам следует убедиться в том, что проблема достаточно конкретно определена.

2. До обсуждения ЛППР необходимо четко сформулировать 12 вопросов на основе полученного определения проблемы. Слова в вопросах (почему, что, когда, где, кто, как) должны использоваться дважды: при наличии проблемы и при ее отсутствии. Вопросы могут быть сформулированы, например, следующим образом: когда проблема возникает и когда она не возникает?; как Вы определяете, что данная проблема существует, и как, что она не существует? и т.д. После этого следует записать все 12 вопросов в виде заголовков.

3. Подготовка и сессия мозгового штурма по определению ответов по каждому вопросу-заголовку. ЛППР задает тему обсуждения, устанавливая порядок рассмотрения сформулированных вопросов. Рекомендуется придерживаться следующего порядка: а) рассматриваются вопросы со словом «Когда?»; б) далее вопрос со словом «Как?»; в) «Где?»; г) «Что?»; д) «Кто?»; е) «Почему?».

4. Обдумывание ответов в течение заранее установленного времени. Затем информация размещается на табло.

5. Анализ диаграммы и выделение ответов для концентрации усилий по поиску решений и сбору данных.

6.6. Методы редукции сложности задач

Идея анализа и решения задач, сводимых к математическим задачам, делением целого на части, высказывалась еще Р. Декартом, рекомендо-

вавшим расчленив каждую изучаемую задачу на столько частей, на сколько можно и на сколько это потребуется, чтобы их было легко решить [250]. При этом очевидно, что великий мыслитель и математик оставлял все действия по расчленению за человеком.

Термин «расчленение», введенный еще Декартом, чтобы обозначить деятельность математика, снижающего сложность недоступной его пониманию задачи, с развитием методов оптимизации в кибернетике и системном анализе уступил место понятиям «редукции» и «декомпозиции». Чтобы придать дальнейшим рассуждениям однозначность, будем использовать термин «редукция» для обозначения действия, а термин «декомпозиция» для обозначения результата этого действия.

Проблематика анализа сложных, неоднородных задач двоякая. С одной стороны, по (5.10) они имеют множество декомпозиций $\hat{\Pi}^u$, структурирующих задачи-системы по множеству отношений ${}^{88}R_3^n$. Таким образом, многоаспектность анализа должна обеспечиваться не только по отношениям включения, формирующим древовидные структуры, но и по другим классам отношений, образующим уже сетевые структуры. С другой стороны, редукция во множество однородных задач (5.7) и построение декомпозиций $\hat{\pi}^u \in \hat{\Pi}^u$ — только один из этапов системного анализа неоднородных задач. За редукцией следуют спецификация однородных задач, анализ $\hat{\pi}^u \in \hat{\Pi}^u$ на неоднородность и выбор методов, релевантных свойствам задач $\pi^h \in \Pi^h$.

В настоящем разделе приведены методы редукции сложных задач, удовлетворяющих модели неоднородной задачи (5.10). Все рассмотренные здесь методы относятся к методам анализа, основанным на знаниях.

Метод редукции по фазам управления использует идею классификации задач по фазам управления (разд. 1.3). Метод исходит из предположения, что в сложной задаче можно выделить однородные подобласти (5.7), наследующие свойства фаз управления, т.е. подобласти однородных задач учета, анализа, прогнозирования и т.п.

Фазовая редукция заменяет анализ сложной задачи на одной фазе анализом множества альтернативных задач на других фазах управления, но обязательно с учетом их отношений. При этом между неоднородной задачей и задачами из альтернативного множества устанавливаются отношения включения, например, «целое-часть», как показано на рис. 6.5. Предполагается, что однородные задачи учета, контроля, прогнозирования и т.д., включенные в неоднородную задачу, не имеют в своем составе других задач.

Фазовая редукция позволяет автоматически установить горизонтальные отношения однородных задач по схеме ролевых концептуаль-

ных моделей взаимодействия фаз. Например, для задач планирования (рис. 6.5) может быть использована модель [1]. Тогда редукция сложной задачи планирования выполняется по следующей схеме:

$$\begin{aligned}
 & {}^8x_{nl}^u \xrightarrow{\pi_{nl}^u, \pi_{nl}^u |^{kr}} 1/{}^{88}r_2^n \{ {}^8X^n|_y, {}^8X^n|_k, {}^8X^n|_a, {}^8X^n|_h, {}^8X^n|_{np} \} \circ \\
 & \circ {}^8X^n|_y \quad {}^{88}r_1^n \quad {}^8X^n|_k \circ {}^8X^n|_y \quad {}^{88}r_1^n \quad {}^8X^n|_a \circ {}^8X^n|_a \quad {}^{88}r_1^n \quad {}^8X^n|_h \circ \\
 & \circ {}^8X^n|_a \quad {}^{88}r_1^n \quad {}^8X^n|_{np},
 \end{aligned} \tag{6.20}$$

где ${}^8x_{nl}^u \in {}^8X^n|_u$ — знак неоднородной задачи планирования в языке (7.3); π_{nl}^u — схема ролевых концептуальных моделей задачи планирования; $\pi_{nl}^u |^{kr}$ — схема ролевых концептуальных моделей процедуры координации решения однородных задач; ${}^8X^n|_y, {}^8X^n|_k, {}^8X^n|_a, {}^8X^n|_h, {}^8X^n|_{np}$ — множества знаков однородных задач учета, контроля, анализа, нормирования и прогнозирования соответственно; ${}^{88}r_1^n, 1/{}^{88}r_2^n$ — отношения «предшествовать» и «целое-часть» соответственно.

Схема (6.20) заменяет знак неоднородной задачи ${}^8x_{nl}^u$, стоящий

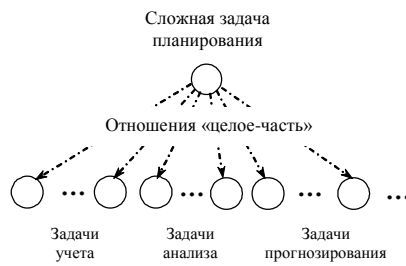


Рис. 6.5 — Фазовая редукция сложной задачи

слева от знака « \rightarrow », на указанную справа от него декомпозицию $\hat{\pi}^u$.

Метод редукции по информационно-управляющим отношениям предполагает, что сложная задача может быть разделена в результате анализа информационных отношений между ее подзадачами. Однородная задача в этом случае представляется моделью

типа «черный ящик», для которого определены вход и выход.

Полученная в результате такой редукции декомпозиция должна отображать обратные и координирующие связи источников и приемников информации, т.е. информационно-управляющие связи. Эти связи могут быть жесткими при вертикальном управлении и мягкими для горизонтального, интеллектуального и виртуального предприятий.

Существует ограниченное число таких элементарных структур, комбинируя которые, можно получать декомпозиции неоднородных задач. Наиболее известные структуры показаны на рис. 6.б.

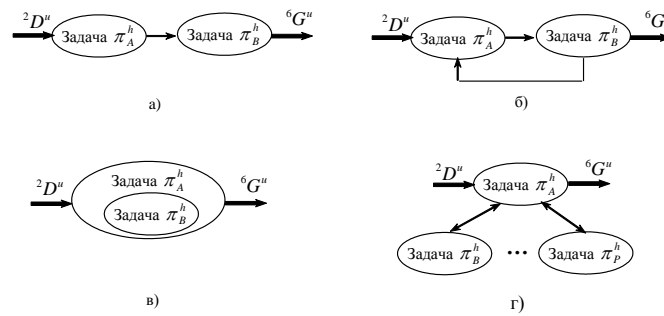


Рис. 6.б — Элементарные структуры в методе редукции по информационно-управляющим отношениям

Их смысл выражается такими словами, как «последовательность», «цепочка», «предобработка», «постобработка» для первых двух структур, «подчиненная обработка» для третьей структуры и «метаобработка» для четвертой структуры.

На рис. 6.б,*а* изображена цепочная обработка информации. Исходные данные ${}^2D^u$ неоднородной задачи π^u — одновременно вход задачи π_A^h , а цель ее решения передается на вход задачи π_B^h . Результаты решения π_B^h отождествляются с целью ${}^6G^u$ задачи π^u . На рис. 6.б,*б* показана цепочка с обратной связью между однородными задачами.

На рис. 6.б,*в* показано отношение включения одной однородной задачи в другую. На практике решатель задачи π_B^h может рассматриваться как консультант, помощник решателя задачи π_A^h , вход и выход которой совпадают с входом и выходом соответственно π^u .

На рис. 6.б,*г* показана элементарная структура для метаобработки информации. В этом варианте сложная, неоднородная задача π^u редуцирована во множество однородных задач, одна из которых (π_A^h) координирует (вызов, перераспределение ресурсов, ограничение деятельности, согласование, передача управляющей информации) решение задач π_B^h, \dots, π_p^h . Координация выполняется эвристическими знаниями, извлеченными из экспертов.

С учетом вышесказанного редукция сложной, неоднородной задачи по информационно-управляющим отношениям может быть выполнена по приведенной ниже схеме:

$${}^8X^n|u \xrightarrow{\pi^u|_1^2, \dots, \pi^u|_4^2; \pi^u|_{kr}} ({}^8X^n|h; {}^{88}R_3^n, {}^{88}R_4^n), \quad (6.21)$$

где ${}^8X^n|u \in {}^8X^n|u$ — знак неоднородной задачи в языке (7.3); $\pi^u|_1^2, \dots, \pi^u|_4^2$ — схемы ролевых концептуальных моделей элементарных информационно-управляющих структур (рис. 6.6); ${}^8X^n|h$ — множество знаков однородных задач; ${}^{88}R_3^n$ и ${}^{88}R_4^n$ — множества отношений декомпозиции (тонкие стрелки на рис. 6.6) и отношений элементов π^u и π^h (жирные стрелки на рис. 6.6) соответственно.

При использовании метода редукции по информационно-управляющим связям сложная задача ассоциируется в единстве с ее решателем, а декомпозиция представляет собой сеть решателей, связанных информационно-управляющими связями. Такой метод декомпозиции, как показывает практика, легко понимается модельерами и экспертами.

Метод редукции по стратам. Сложная задача заменяется множеством однородных задач, возникающих как альтернативные, упрощенные, профессиональные образы π_B^h, \dots, π_p^h одной и той же задачи (рис. 6.7).

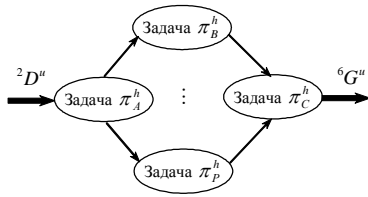


Рис. 6.7 — Редукция сложной практической задачи по стратам

Кроме этого, вводятся еще две задачи π_A^h и π_C^h , выполняющие пред- и постобработку информации соответственно. Предобработка может быть достаточно простой, когда каждая из задач π_B^h, \dots, π_p^h получает свою «порцию» информации, или более сложной. Постпроцессия может быть, например, упорядочением, ранжированием

${}^6G_B^h, \dots, {}^6G_p^h$ в ${}^6G^u$ или выполняться по более сложной схеме, определяемой ${}^{88}R_4^n$. Обратим внимание на то, что на множестве

$\{\pi_B^h, \dots, \pi_P^h\}$ информационно-управляющие или другие отношения не определены.

Редукция по стратам зависит от профессиональных точек зрения на одну и ту же проблему. В этом случае однородность в пределах страты проявляется в однородности языка профессиональной деятельности, однородности фундаментальных, полностью формализованных знаний и, в определенной степени, в частично формализованных знаниях, поскольку даже для одной и той же профессии можно получать различные образы одной и той же неоднородной задачи.

С учетом вышесказанного, редукция неоднородной задачи по стратам может быть выполнена по приведенной ниже схеме:

$${}^8x^n|_u \xrightarrow{\pi^u|_S \dots \pi^u|_{kr}} ({}^8x^n|_A, {}^8x^n|_C, {}^{88}R_4^n) ({}^8X^h, {}^{88}R_3^n), \quad (6.22)$$

где $\pi^u|_S$ — схема ролевых концептуальных моделей для стратифицированного описания системы S (2.1).

Проблемы данного метода редукции — выбор профессиональных точек зрения, т.е. экспертов, и смещение акцентов на согласование результатов решения однородных задач. Метод избыточен, поскольку заменяет моделирование одной задачи моделированием простых задач — профессиональных образов.

Метод смешанной редукции. В смешанной редукции методы ранжированы. Фазовая редукция имеет первый приоритет, что позволяет не только отделить задачи пассивных фаз (учет, контроль), которые могут быть решены в рамках традиционных технологий информатики — баз данных, электронных таблиц, тестовых и графических редакторов, от задач активных фаз. Второй приоритет отдается стратификации или информационно-управляющей редукции, которые отражают информационный и системный подходы, и, наконец, третий приоритет — методам редукции формализованных или трудноформализуемых задач [1]. Такая приоритетная эвристика позволяет гибко сочетать достоинства различных методов редукции — неформальное, резкое сокращение размерности области допустимых решений на первых, менее формализованных этапах анализа сложных задач, с последующими формально-логическими приемами анализа, где накоплены фундаментальные поисковые знания в символьных системах и аналитических моделях.

Пример подобной сети приведен на рис. 6.8. На нем сложная, неоднородная задача π^u первоначально была редуцирована по методу фаз с использованием отношения «целое-часть». В результате получены

$\pi_2^h, \pi_3^h, \pi_4^h$, которые затем по методу информационно-управляющей редукции связаны координирующей задачей π_1^h в структуру «метаобработка».

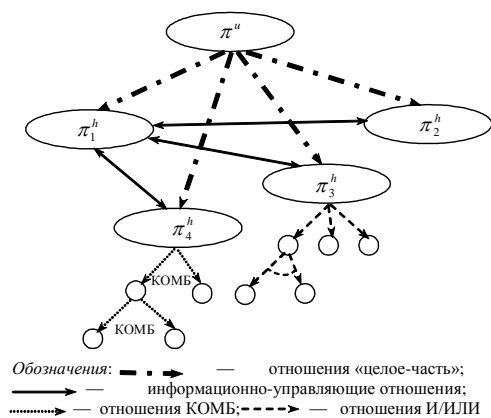


Рис. 6.8 — Смешанная редукция сложной практической задачи

Две однородные задачи π_3^h и π_4^h , оказалось, можно редуцировать еще и в граф И/ИЛИ и КОМБ [1] соответственно.

6.7. Выбор методов решения однородных задач

Один из этапов «жизненного цикла» функциональных ГиИС (рис. 6.1) — системный анализ областей неоднородности в сложной задаче с целью подбора инструментариев, релевантных спецификациям подзадач, входящих в эту область.

Эта проблема находится в ряду известных проблем информатики — выбора языка программирования или моделирования описания алгоритма, выбора инструментальной среды для разработки приложения, выбора СУБД для манипулирования данными и др. Однако если перечисленные проблемы могут решаться по принципу «работай на том, что знаешь и на чем умеешь», то с задачей так поступить не удастся. Во-первых, метод решения задачи должен быть принят значительно раньше, чем наступает время выбирать СУБД и писать программу. Выбор метода — это уже серьезный «намеки» на алгоритм решения. Во-вторых, метод специфицируется на ранних этапах, в условиях, когда задача до конца не понята и риск неправильного выбора слишком велик.

Проблема выбора, при условии, что она сформулирована на языке математики, хорошо исследована в теории принятия решений. Для плохо структурированных задач выбора строятся и применяются таблицы «задача-метод» [251], носящие синтаксический, поверхностный характер, в то время как знания и опыт автора этой таблицы не отражены в ней и остались со своим носителем — человеком. В других работах, посвященных гибридным генетическим алгоритмам, проблема выбора метода для продолжения поиска [87] решается автоматически по результатам анализа последствий принятого решения на последующих шагах работы алгоритма. В [252] предложено эвристическое решение проблемы выбора между численными методами для решения математических задач с помощью искусственной нейронной сети.

В настоящем разделе рассматривается подход к решению задачи выбора метода, основанный на использовании базы знаний.

Первая отличительная особенность предлагаемого метода состоит в том, что задача относится не к конкретному инструментарию, а к их классу, что в значительной мере снижает сложность проблемы. С другой стороны, такое решение упрощает последующий выбор конкретного метода внутри уже специфицированного класса. Вторая особенность — ориентация не на проблематичное в настоящее время получение аналитических зависимостей для записи соответствия «задача-методы», а на использование знаний, накопленных в мировой практике. Это знания об особенностях ситуаций применения того или иного метода, знания о выборе коммерческого программного продукта, реализующего тот или иной автономный метод в зависимости от спецификации задачи, знания авторов метода о спектре задач, на которые он ориентирован, и, наконец, знания и опыт специалистов, эксплуатирующих автоматизированные системы обработки и управления, отдельные задачи в которых решены с использованием того или иного метода моделирования.

Наконец, третья особенность — гибридизация здесь рассматривается как процесс сбора, системного анализа и преобразования информации об объектах-оригиналах (сложных задачах), объектах-прототипах (методах моделирования) в объекты-результаты — функциональные ГиИС.

Пусть по результатам первых трех этапов проблемно-структурной методологии функциональных ГиИС (см. рис. 6.1) построена декомпозиция Π^h сложной задачи и в ее составе имеется N_h подзадач, удовлетворяющих модели «однородная задача» π^h . Пусть имеется также N_m инструментальных средств, и каждая задача $\pi_j^h | j = 1, \dots, N_h$ по результатам спецификации представлена в языке описания однородных задач

L_6^h (5.9) знаком ${}^8X^n|_j^h \in {}^8X^n|_h$. Каждому знаку ${}^8X^n|_j^h$ может соответствовать более чем один метод $m_l^a \in M^a | l=1, \dots, N_m$, $M^a = \bigcup_{q=\{An, St, Lg, Li, Ge, Ep\}} {}^qM^a$, где An, St, Lg, Li, Ge, Ep — аналитические, статистические, логические, лингвистические, генетические знания и опыт соответственно, и, наоборот, методу m_l^a может соответствовать множество знаков в ${}^8X^n|_h$.

Пусть методы из M^a разбиты на шесть классов базовых автономных методов ${}^{AN}M^a, {}^{ИСМ}M^a, {}^{ЭС}M^a, {}^{НС}M^a, {}^{ИНС}M^a, {}^{ГА}M^a$, причем ${}^{AN}M^a \subseteq {}^{An}M^a$; ${}^{ИСМ}M^a \subseteq {}^{St}M^a$; ${}^{ЭС}M^a, {}^{НС}M^a \subseteq {}^{Lg}M^a$; ${}^{ЭС}M^a, {}^{НС}M^a \subseteq {}^{Li}M^a$; ${}^{ИНС}M^a \subseteq {}^{Lg}M^a$; ${}^{ИНС}M^a \subseteq {}^{An}M^a$; ${}^{ГА}M^a \subseteq {}^{Ge}M^a$ (обозначения $AN, ИСМ, ЭС, НС, ИНС, ГА$ даны в табл. 6.1; класс автономных методов рассуждений на основе опыта не рассматривался). Каждому из шести классов взаимно однозначно сопоставим подмножество знаков ${}^8X^n|_{ha} \subseteq {}^8X^n|_h$.

Таблица 6.1 — База знаний «задача-методы»

Классы базисных методов	Свойства подзадачи ${}^2O_1^a$					
	Свойства среды разработки ${}^{22}X^0 _{DV}$	Свойства проекта ${}^{22}X^0 _{PR}$	Свойства среды экспертов ${}^{22}X^0 _{EX}$	Свойства плана решения ${}^{22}X^0 _{PP}$	Свойства измерений, действий и информации ${}^{22}X^0 _{MI}$	Свойства проблемной среды ${}^{22}X^0 _{PM}$
	(14)	(9)	(12)	(16)	(15)	(35)
Аналитические методы (АН)	Значения свойств ${}^5X^0 \in \{0,1,2,\dots,40\}$					
Имитационное статистическое моделирование (ИСМ)						
Искусственные нейронные сети (ИНС)						
Экспертные системы (ЭС)						
Нечеткие системы (НС)						
Генетические алгоритмы (ГА)						

Тогда установление соответствия «задачи-методы» сводится к задаче отнесения знака ${}^8X^n|_j^h$ в класс задач ${}^8X^n|_{ha}$, решаемых подмножеством инструментариев ${}^qM^a | q = \{AN, ИСМ, ЭС, НС, ИНС, ГА\}$.

Таким образом, в рассматриваемой задаче классификации выбор метода $m_i^a \in M^a$ для решения однородной задачи ${}^8X^n|_j^h \in {}^8X^n|_h$ сведен к выбору подмножества методов ${}^qM^a \subseteq M^a$, что сделать значительно проще.

Суть предлагаемого метода состоит, прежде всего, в отказе от аналитического представления соответствия $\Psi^{\pi m}|_{ui} = \{\pi_1^h\{m_i^a\}_1^h, \dots, \pi_{N_h}^h\{m_i^a\}_{N_h}^h | \{m_i^a\}_1^h$ — множество релевантных спецификации однородной задачи π_1^h методов (6.19), на множествах объектов-оригиналов ${}^8X^n|_{ha}$ и объектов-прототипов M^a , т.е. $\Psi^{\pi m}: {}^8X^n|_{ha} \rightarrow M^a$ и переходе к эвристическому

представлению соответствия, т.е. $KB^{\pi m}$ (6.19): ${}^8X^n|_{ha} \xrightarrow{KB^{\pi m}} M^a$, где

$KB^{\pi m}$ — база знаний, фиксирующая накопленные к настоящему времени в мировой практике эмпирические знания по применению методов и рассмотренных выше базисных классов к решению задач обработки информации и управления. Таким образом, в предлагаемом методе множество $\Psi^{\pi m}|_{ui}$ задается порождающей процедурой, представленной базой знаний $KB^{\pi m}$, а понятие адекватности объектов-оригиналов и объектов-прототипов заменяется понятием их релевантности. Окончательное решение о выборе класса методов и конкретного метода внутри класса принимает разработчик функциональной ГиИС.

Представление знаний из $KB^{\pi m}$ дано в виде табл. 6.1 (полностью $KB^{\pi m}$ рассмотрена в [1]). Строки этой таблицы обозначены шестью классами базисных методов. Столбцы обозначены 101 свойством из спецификатора однородной задачи ${}^2O^h$ (5.7). Эти свойства сгруппированы в шесть множеств. В скобках указано количество свойств, включенных в соответствующее множество. С полным перечнем свойств можно познакомиться в [1]. Элементы табл. 6.1 — это значения свойств, задаваемые на интервале от 1 до 40, причем для спецификации большей части свойств используются двоичные переменные.

Таким образом, табл. 6.1 задает знания «спецификаторы однородных задач (${}^2O^h$) — классификаторы автономных методов (${}^{22}X_1^0$)», далее для краткости знания «задачи-методы», о соответствии на множестве из шести базовых классов методов и множестве из 101 свойства спецификаторов однородных задач.

При разработке базы знаний KB^{mn} учтен опыт более 130 экспертов, в том числе 73 зарубежных, опубликованный ими в различных литературных источниках, характеризующий достигнутое на сегодняшний день качество принятия решения о выборе метода для моделирования решения задачи π_j^h .

Знания из KB^{mn} могут быть представлены продукционными правилами «условие — действие», где условие рассматривается как некоторая ситуация, описанная в L_6^h (5.9), а действие — отнесение ситуации к тому или иному классу ${}^qM^a \subseteq M^a$.

Тогда построим экспертную систему, решающую сформулированную задачу классификации. На вход системы будут поступать знаки ${}^8x^n|_j^h$ информационного языка, а ее выход — балльные оценки релевантности классов базисных методов спецификации «однородная задача» (5.7). В результате работы системы может быть построено множество из N_h пар «задача-методы» $\pi_j^h\{m_l^a\}_j^h$, каждая из которых идентифицирует один или несколько ${}^qM^a$, релевантных свойствам однородной задачи из декомпозиции Π^h (5.10).

Ниже рассмотрен эвристический алгоритм 6.1 функционирования системы с базой знаний.

Дано: множества ${}^{AH}M^a, {}^{ИСМ}M^a, {}^ЭС M^a, {}^{НС}M^a, {}^{ИНС}M^a, {}^{ГА}M^a$ классов базовых методов, знак ${}^8x^n|_j^h \in {}^8X^n|_j^h$ однородной задачи $\pi^h \in \Pi^h$, в языке L_6^h , и база знаний KB^{mn} , записанная 220 продукционными правилами «условие-действие».

Найти: балльные оценки ${}^7x_{mn}^n \in {}^7X^n$, где ${}^7X^n$ — множество оценок в категориальном ядре теории Te (5.1), характеризующие применимость каждого из шести классов базовых методов ${}^{AH}M^a, {}^{ИСМ}M^a, {}^ЭС M^a, {}^{НС}M^a, {}^{ИНС}M^a, {}^{ГА}M^a$ для решения задачи ${}^8x^n|_j^h$, как по всем ее свойствам ${}^2O^h$ (обобщенный уровень, обобщенные оценки), так и по группам свойств из ${}^2O^h$ (детализированный уровень, детализированные оценки).

Обозначения: введем множество счетчиков $Cc = \{Cc_{jq}\}$, где $j=0, \dots, 6$ — индекс класса методов, включая $j=0$ — индекс счетчика

обобщенной оценки; $q = 1, \dots, 6$ — индекс счетчика для группы свойств ${}^8x^n|_h$ ($q = 1$ для ${}^{22}X^0|^{DR}$ — среды разработки; $q = 2$ для ${}^{22}X^0|^{PM}$ — проблемной среды; $q = 3$ для ${}^{22}X^0|^{PR}$ — проекта; $q = 4$ для ${}^{22}X^0|^{MI}$ — измерений, действий и информации; $q = 5$ для ${}^{22}X^0|^{EX}$ — среды экспертов и $q = 6$ для ${}^{22}X^0|^{PP}$ — плана решения задачи).

Таким образом, имеем по семь счетчиков на каждый из шести классов методов, в том числе по одному счетчику обобщенной оценки $C_{ch_{on}}$. Для проверки значения свойства знака ${}^8x^n|_j \in {}^8X^n|_h$ информационного языка, а общее количество таких свойств на сегодняшний день 101, разработаны группы правил «условие-действие». Количество правил в группе зависит от числа альтернативных значений того или иного свойства в KB^{tm} . В левой части каждого правила значения свойств связаны конъюнкцией, а его правая часть — действие, увеличивающее на единицу значений тех из счетчиков $C_{ch_{1q}}, \dots, C_{ch_{6q}}$, сопоставленных классам методов, к которым может быть отнесен знак ${}^8x^n|_j$.

Алгоритм б.1:

- 1) обнулить счетчики; $k = 1$;
- 2) увеличить на единицу содержимое счетчиков тех групп свойств, значения которых в ${}^8x^n|_h \in {}^8X^n|_h$ и в левой части правила совпадают.

Это реализуется с использованием базы знаний KB^{tm} следующим образом. Прежде всего, от ${}^8x^n|_h$ отделяется в порядке слева направо k -е свойство вместе со своим значением. По ним распознается группа правил из базы знаний, применяемая для решения задачи классификации. Затем используется стандартный интерпретатор оболочки экспертных систем КАРРА-РС для сопоставления значения k -го свойства с левыми частями правил (может быть и одного) идентифицированной группы;

- 3) вычислить балльные оценки применимости каждого из шести классов методов к ${}^8x^n|_h$ на обобщенном и детализированном уровнях сле-

дующим образом: ${}^7x_{nm}^n|_{jq} = C_{ch_{jq}} / C_{ch_{jq}}^{\max}$, ${}^7x_{nm}^n|_{on} = \sum_{j=1}^6 C_{ch_{jq}}$, где

${}^7x_{nm}^n|_{jq}$ — балльная оценка применимости j -го класса методов по q -й

группе свойств; C_{jq}^{\max} — max число свойств, отличающих j -й класс методов в q -й группе свойств;

4) $k = k + 1$;

5) если $k \leq 101$, перейти к п. 2, иначе «конец».

Программная реализация алгоритма 6.1 рассмотрена в разд. 7.5.

Реализация метода вероятностной нейронной сетью. Применение вероятностных нейронных сетей (разд. 3.3.4) для решения поставленной выше задачи классификации интересно из-за возможности получения на выходе такой сети вероятностных оценок отнесения задачи к тому или иному классу базисных методов. Для реализации была разработана программа, отличная от реализации PNN в пакете MATLAB (разд. 7.5).

Обучающие примеры были сформированы в соответствии с базой знаний KB^m (табл. 6.1), а для экспериментов использовались три группы свойств: среды разработки, проекта и среды экспертов. Для этих групп количество обучающих векторов оказалось равным 19, что и определило количество нейронов в слое образцов. В суммирующем слое — шесть нейронов по числу классов базисных методов $АНM^a, ИСМM^a, ЭСM^a, ИСM^a, ИНСM^a, ГАМ^a$.

Результаты экспериментов с PNN-сетью, решающей задачу классификации, приведены в разд. 7.5.

6.8. Метод проверки неоднородности сложных задач

Пятый этап жизненного цикла ПС-методологии (рис. 6.1) — это проверка неоднородности сложной задачи. От ее результатов зависит объем последующих действий разработчика: 1) либо он продолжит следовать методологии гибридизации, если сложная задача обладает свойством неоднородности; 2) либо он перейдет к одной из методологий автономного моделирования, если сложной задаче не присуща неоднородность.

В настоящем разделе рассматривается метод системного анализа областей с однородными параметрами в задаче-системе.

Пусть построена декомпозиция Π^h сложной задачи и в ее составе имеется N_h подзадач, удовлетворяющих модели «однородная задача» π^h (5.7). Каждая задача $\pi_j^h | j = 1, \dots, N_h$ по результатам спецификации представлена в языке описания однородных задач L_3^h (5.9) знаком ${}^8x^n |_{j \in {}^8X} |^h$.

Системный анализ областей однородных задач сложной задачи выполняется в несколько этапов. Сначала все множество ее подзадач разбивается на кластеры. Для этого формируется множество векторов параметров подзадач, затем с помощью алгоритма ISODATA это множество разбивается на кластеры, и для каждого кластера формируется представитель — вектор параметров.

При выборе алгоритма для кластеризации подзадач были учтены следующие особенности: количество параметров достаточно велико, около 100 (разд. 6.7) и некоторые из них могут быть описаны только категориальными переменными; число кластеров заранее неизвестно, т.е. необходимо применение так называемых алгоритмов обучения без учителя или неконтролируемых алгоритмов обучения; желательна гибкость в варьировании кластеризации, поскольку структура ожидаемых кластеров заранее неизвестна.

Всем этим требованиям удовлетворяет алгоритм ISODATA (название происходит от англ. ISODATA, что означает «Итеративная самоорганизующаяся система анализа данных»). В укрупненном виде этапы алгоритма следующие:

Алгоритм 6.2:

- 1) исходное положение центров выбирается произвольно;
- 2) определяются области, в которые входят объекты, близкие к исходным центрам;
- 3) делят на 2 каждую группу, внутри которой среднее расстояние между точками превышает некоторый порог;
- 4) определяются новые центры областей с учетом вновь появившихся областей;
- 5) вычисляются расстояния между каждой парой центров областей;
- 6) объединяются области, расстояние между которыми меньше некоторого порога;
- 7) выполняются процедуры заново, начиная с п. 2, до тех пор, пока количество кластеров не станет постоянным.

После завершения работы алгоритма кластеризации для всех подзадач π^h из декомпозиции Π^h получается результирующая информация, которая может быть использована для продолжения системного анализа сложной задачи с двумя целями.

Цель первая — применение информации для установления соответствия «кластер подзадач — класс автономных методов» (разд. 6.7) с помощью PNN-сети. После окончания процесса классификации разработчику будет предоставлена информация о количестве кластеров, о

подзадачах, вошедших в кластер, и оценки релевантности параметров кластеров классам методов обработки информации.

Вторая цель — применение информации для проверки сложной задачи на неоднородность.

Для этого определяются «плохо» определенные кластеры (подзадачи, для которых разработчик на большинство вопросов не дал никакого ответа). Для этих подзадач релевантные оценки для всех классов методов составляют не более 5%. Затем, не учитывая «плохо» определенные подзадачи, выявляется возможность использования каждого класса методов при решении всех подзадач. Если все подзадачи невозможно решить в рамках одного класса методов, то разработчику выдается сообщение о неоднородности сложной задачи. Иначе сообщается о возможности решения задачи в рамках одного класса методов с учетом доопределения «плохо» определенных подзадач.

6.9. Метод и алгоритмы комбинирования знаний экспертов в функциональных гибридных интеллектуальных системах

Центральная процедура на этапе синтеза функциональных ГиИС — конструирование метода решения сложной, неоднородной задачи. Эта процедура воплощает гипотезу № 4 (разд. 4.7) о том, что не существует одного-единственного, заведомо известного или выбираемого из множества существующих автономных методов решения неоднородной задачи. Такой метод должен быть сконструирован СППР в процессе решения задачи и существует только на текущий момент решения задачи. В следующий момент времени метод, сконструированный ранее, уже может быть нерелевантен состоянию объекта управления в условиях изменяющихся воздействий внешней среды.

Такой метод получается в результате интерпретации ГиИС, конструируемой связыванием отношениями интеграции ${}^{\psi\phi}\ddot{R}^n$ (6.15) элементов — автономных моделей из гетерогенного модельного поля (разд. 6.4), сопоставленного сложной задаче на этапе 6 жизненного цикла на рис. 6.1. Для этого разработчик выполняет редукцию и строит множество декомпозиций $\hat{\Pi}^u$ задачи π^u . Затем разрабатывает и (или) использует однородные модели $\{m^a\}$ и выполняет другие этапы жизненного цикла. Пусть таких функциональных моделей-компонент будет N_m . Таким образом, задаче π^u соответствует гетерогенное поле \dot{M}^a моделей размерности N_m , в котором выделены непустые, может быть, и одноэлементные, подмножества, сопоставленные подзадачам $\pi^h \in \Pi^h$. Далее

строится таблица гибридных стратегий T^u . При этом $\pi^h \in \Pi^h$ связываются с помощью ${}^{11}\ddot{R}^n$ (см. разд. 6.4), заменяющих отношения декомпозиции ${}^{88}R_3^n$ (5.10). Ограничимся случаем, когда между двумя $\pi^h \in \Pi^h$ есть только одно отношение из ${}^{11}\ddot{R}^n$. Пример состояния разработки метода решения π^u после завершения перечисленных выше работ дан на рис. 6.9. На нем показано, что для решения однородной задачи π_1^h разработана одна модель $\dot{m}_3^a|_1^{An}$ с использованием аналитического (An) метода $m_3^a|_1^{An}$, для решения однородной задачи π_2^h разработаны две модели $\dot{m}_1^a|_2^{St}, \dot{m}_2^a|_2^{St}$, обе с использованием двух разных статистических (St) методов $m_1^a|_2^{St}, m_2^a|_2^{St}$ и т.д.

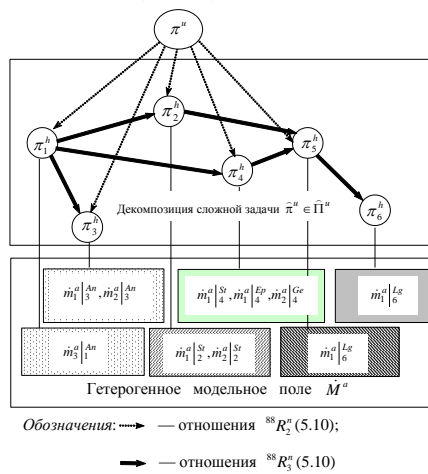


Рис. 6.9 — Пример декомпозиции сложной задачи и гетерогенное модельное поле

процедура оценки и выбора автономной модели для каждой пары $(\pi_i^h, {}^{11}\ddot{R}^n \pi_j^h)$. Эта процедура использует систему с базой знаний

$KB^{\dot{m}^x}$ (6.19), созданную на этапе 6 — «Спецификация областей релевантности методов» ПС-методологии (рис. 6.1) и реализована следующим образом. Априори, на этапах 6 и 9 (рис. 6.1) каждая \dot{m}^a должна быть оценена экспертами на предмет формирования областей релевантности. На основе экспертных оценок настраивается нечеткая система

Поскольку вход, состояние и выход функциональной ГиИС неоднородны, т.е. имеет место комбинация значений переменных из разных классов, далее должны быть установлены отношения ${}^{88}R_4^n$ (5.10). После того, как ${}^{88}R_4^n$ заданы и, поскольку одной задаче π^h в декомпозиции $\hat{\pi}^u$ может соответствовать более одной модели \dot{m}^a , в гибридизацию (см. ниже п. 6, 10, 11 алгоритмов 6.3, 6.4) включена

или экспертная система, на вход которой при конструировании ГиИС подаются идентификатор задачи π^h (5.8) и состояние ${}^2\hat{\mathbf{x}}_3^n(t)$ из схемы концептуальных моделей крупнозернистых функциональных ГиИС (5.22), а на выходе получается необходимая оценка. Модель из ранжированного таким образом списка, для каждой подобласти гетерогенного модельного поля, может быть выбрана пользователем во время инициализации ГиИС по априори заданным им предпочтительным интервалам оценок или случайным образом. Наконец, он может и не согласиться с компьютерными оценками и самостоятельно принять решение о применении той или иной модели.

Приведенный ниже алгоритм 6.3 использует рассмотренную выше процедуру и конструирует метод решения неоднородной задачи π^u в символической форме. Для этого декомпозиция $\hat{\pi}^u$, гетерогенное модельное поле \dot{M}^a , таблица гибридных стратегий и некоторая другая информация (см. ниже «дано») преобразуются в список Ls^m , в котором знаки однородных задач заменены на знаки выбранных автономных моделей, каждой модели поставлен в соответствие интерпретатор: $I^a, I^s, I^e, I^f, I^{n1}, I^{n2}, I^g, I^p$ (разд. 3) и определены интерфейсы для обмена информацией. Список Ls^m — исходная информация для работы интерпретатора ГиИС [1].

Тогда может быть сформулирована задача конструирования функциональной ГиИС, релевантной $\hat{\pi}^u$ для декомпозиций в виде графа-дерева.

Дано: декомпозиция $\hat{\pi}^u = (\Pi^h, {}^{88}I_{3_1}^n) | {}^{88}I_{3_1}^n \in {}^{88}R_3^n$ в виде графа-дерева, гетерогенное модельное поле \dot{M}^a , таблица гибридных стратегий T^u для декомпозиции $\hat{\pi}^u$, которые каждой тройке $\pi_\gamma^h, {}^{88}I_{3_1}^n, \pi_\zeta^h \in \hat{\pi}^u | \gamma, \zeta = 1, \dots, N_h, \gamma \neq \zeta$ взаимно однозначно сопоставляют множество (в алгоритме 6.3 однозначное) отношений интеграции $\{\dot{\pi}^n\} | \dot{\pi}^n \in {}^{11}\dot{R}^n$, т.е. $\pi_\gamma^h, {}^{88}I_{3_1}^n, \pi_\zeta^h \leftrightarrow \{\dot{\pi}^n\}$, множество отношений ${}^{88}R_4^n$ соответствия целей и исходных данных однородных задач из $\Pi^h = \{\pi_1^h, \dots, \pi_{N_h}^h\}$, а также целей и исходных данных π^u , множество интерпретаторов автономных моделей $I^{\dot{M}^a} = \{I^a, I^s, I^e, I^f, I^n, I^k, I^g, I^p\}$ и множество межмодельных интерфейсов $\{\rho^u\}$. Кроме того, установлены соответствия моделей

$\dot{m}^a \in \dot{M}^a$ и интерпретаторов $I^v \in I^{\dot{M}^a} \mid v = \{a, s, e, f, n, k, g, p\}$, т.е. задан метод решения π^h , а также $\dot{m}^a|_y^h$ и ${}^{99}\rho^u$, т.е. задан алгоритм преобразования представления информации с одного языка описания модели в другие.

Найти: список Ls^m , состоящий из троек $(\dot{m}_x^a \text{ }^{11};^n|_y^x \dot{m}_y^a)|_\zeta^y$, где $x, y = 1, \dots, N_m$, причем каждой модели \dot{m}^a сопоставлен интерпретатор автономного метода I^v , т.е. $\dot{m}^a \rightarrow I^v$, а каждому отношению интеграции ${}^{11};^n|_y^x$ сопоставлены интерфейсы ${}^{99}\rho^u|_y^x$ и (или) ${}^{99}\rho^u|_x^y$.

Обозначения: E_j^l — матрица «модель-модель», строки которой соответствуют моделям задачи π_ζ^h , а столбцы моделям задачи π_γ^h ; $E^l = \{E_1^l, \dots, E_{N_\phi}^l\}$ — множество матриц E_j^l для множества троек $\{(\pi_\gamma^h \text{ }^{88};^n|_{\zeta_1}^h \pi_\zeta^h)\}$ из $\tilde{\pi}^u$, обрабатываемое на l -м шаге алгоритма; N_E — количество l -х шагов алгоритма, определяемое размерами дерева «по вертикали»; Ls^* — вспомогательный список для обработки троек, у которых задача π_γ^h — корень графа-дерева.

Алгоритм 6.3 (фрагмент алгоритма изображен на рис. 6.10):

- 1) начало, $l=1$;
- 2) выбрать из $\tilde{\pi}^u$ множество троек $\Pi_\beta^h = \{(\pi_\gamma^h \text{ }^{88};^n|_{\zeta_1}^h \pi_\zeta^h)\} \mid \beta = 1, \dots, N_{\Pi^h}$, в которых каждая π_ζ^h находится в листе $\tilde{\pi}^u$;
- 3) $j=1, E^l = \emptyset$;
- 4) выбрать очередную j -ю пару из Π_β^h ;
- 5) если π_γ^h — корень, то перейти к п. 6, иначе перейти к п. 7;
- 6) инициализировать для j -й пары список Ls_j^* , помечая его элементы \mathcal{E}_ω^j моделями задачи π_ζ^h . Используя экспертную или нечеткую систему, получить оценки моделей из $\{\dot{m}_\zeta^a\}$, выбрать элемент с *max* значением оценки и занести ${}^{11};^n|_y^x \dot{m}_y^a \mid y = 1, \dots, N_m$ в список Ls^u . Перейти к п. 8;

- 7) инициализировать для j -й пары матрицу E_j^l , помечая ее строки q и столбцы ω моделями \dot{m}^a задач π^h (столбцы помечаются $\{\dot{m}_\gamma^a\}$, а строки — $\{\dot{m}_\zeta^a\}$). Дополнить $E^l | l=1, \dots, N_E$ матрицей E_j^l , т.е. $E_j^l \in E^l$;
- 8) $j = N_{PH} + 1$? Если «нет», то $j = j + 1$ и перейти к п. 4, иначе удалить листья из $\hat{\pi}^u$ и проверить, что в $\hat{\pi}^u$ остался только корень? Если «да», то $l = 1$ и перейти к п. 9, иначе $l = l + 1$ и перейти к п. 2;
- 9) $j = 1$;

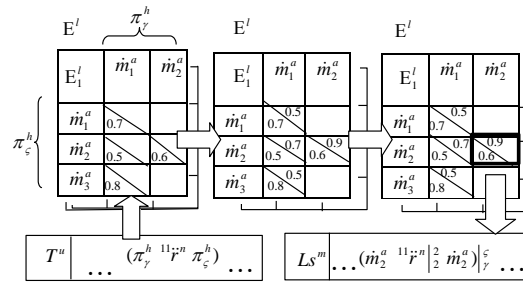


Рис. 6.10 — Фрагмент алгоритма синтеза функциональной ГИС

- 10) выбрать $E_j^l | j=1, \dots, N_{PH}$ из $E^l = \{E_1^l, \dots, E_{N_{PH}}^l\}$. Используя T^u , активировать только те элементы $\varepsilon_{q\omega}^j$ в E_j^l , для которых между \dot{m}_q^a и \dot{m}_ω^a существует отношение $^{11};^n$ (на рис. 6.10 это показано перечеркиванием клетки матрицы E_j^l). Если $\{\dot{m}_q^a\}$ уже оценивались, то занести эти оценки в знаменатель $\downarrow \varepsilon_{q\omega}^j$ дробей-элементов $\varepsilon_{q\omega}^j$ и перейти к п. 11, иначе, используя экспертную или нечеткую системы, получить оценки моделей $\{\dot{m}_q^a\}$ задачи π_ζ^h , заноса оценки в знаменатели $\downarrow \varepsilon_{q\omega}^j$ дробей-элементов $\varepsilon_{q\omega}^j$;

- 11) моделировать решения π_c^h на каждой из $\{m_q^a\}$, и полученные значения выходов модели m_q^a использовать в качестве входов модели m_ω^a . Используя экспертную или нечеткую системы, получить оценки моделей $\{m_\gamma^h\}$ задачи π_γ^h , заноса оценки в числители $\hat{\varepsilon}_{q\omega}^j$ дробей-элементов $\varepsilon_{q\omega}^j$. По формуле $\max \sum_{q\omega} ((\hat{\varepsilon}_{q\omega}^j + \downarrow \varepsilon_{q\omega}^j) - \downarrow \varepsilon_{q\omega}^j - \uparrow \varepsilon_{q\omega}^j)$ выбрать адекватную ${}^2\hat{\mathbf{x}}_3^n(t)$ пару моделей $(m_x^a \overset{11}{r} m_y^a) | x, y = 1, \dots, N_m$ и занести ее в список Ls^u ;
- 12) $j = N_{PH} + 1$? Если «нет», то $j = j + 1$ и перейти к п. 10, иначе к п. 13;
- 13) $l = N_E + 1$? Если «нет», то $l = l + 1$ и из подмножеств моделей, сопоставленных задачам $\{\pi_\gamma^h\}$, удалить модели в списке Ls^u , которых нет в $\{m_x^a\}$, и перейти к п. 9, иначе перейти к п. 14;
- 14) дополнить Ls^u списком интерпретаторов и интерфейсов. Инициализировать ГиИС в соответствии с Ls^u ;
- 15) конец.

Для декомпозиций типа «сеть» условия предыдущей задачи сохраняются, за исключением того, что декомпозиция $\hat{\pi}^u$ задается не деревом, а сетью — ориентированным графом без петель. Кроме этого на множестве $\Pi^h = \{\pi_1^h, \dots, \pi_{N_h}^h\}$ задан порядок, т.е. известна очередность решения π^h . Порядок задан следующим образом. Построено множество $In = \{In^1, \dots, In^{NT}\}$, где $In^1 = \{in_1, \dots, in_{m^1}\}$ — множество нижних индексов задач π^h , решаемых в первую очередь, ..., $In^{NT} = \{in_1, \dots, in_{m^{NT}}\}$ — множество нижних индексов задач π^h , решаемых в последнюю очередь.

Алгоритм 6.4:

- 1) начало, $l = 1$, $In^l = In^1$;
- 2) выбрать из $\hat{\pi}^u$ множество троек $\Pi_\beta^h = \{(\pi_\gamma^h \overset{88}{r} \pi_\zeta^h) | \beta = 1, \dots, N_{PH}\}$, в которых каждая π_ζ^h имеет очередность из In^l ;

- 3) $j = 1, E^l = \emptyset$;
- 4) выбрать очередную j -ю пару из Π_ϕ^h ;
- 5) инициализировать для j -й пары матрицу E_j^l , помечая ее строки q и столбцы ω моделями \dot{m}^a задач π^h (столбцы помечаются $\{\dot{m}_q^a\}$, а строки — $\{\dot{m}_\omega^a\}$). Дополнить $E^l | l = 1, \dots, N_E$ матрицей E_j , т.е. $E_j^l \in E^l$;
- 6) $j = N_{PH} + 1$? Если «нет», то $j = j + 1$ и перейти к п. 4, иначе — к п. 7;
- 7) $j = 1$;
- 8) выбрать $E_j^l | j = 1, \dots, N_{PH}$ из $E^l = \{E_1^l, \dots, E_{N_{PH}}^l\}$. Используя T^u , активировать только те элементы $\varepsilon_{q\omega}^j$ в E_j^l , для которых между \dot{m}_q^a и \dot{m}_ω^a есть отношение ${}^{11}r^n$. Используя экспертную или нечеткую системы, получить оценки моделей $\{\dot{m}_q^a\}$ задачи π_ζ^h , заноса оценки в знаменатели $\downarrow \varepsilon_{q\omega}^j$ дробей-элементов $\varepsilon_{q\omega}^j$;
- 9) моделировать решения π_ζ^h на каждой из $\{\dot{m}_q^a\}$ и полученные значения выходов \dot{m}_q^a использовать в качестве входов \dot{m}_ω^a . Используя экспертную или нечеткую, получить оценки моделей $\{\dot{m}_l^h\}$ задачи π_γ^h , заноса оценки в числители $\uparrow \varepsilon_{q\omega}^j$ дробей-элементов $\varepsilon_{q\omega}^j$. По формуле $\max_{q\omega} \sum ((\uparrow \varepsilon_{q\omega}^j + \downarrow \varepsilon_{q\omega}^j) - \downarrow \varepsilon_{q\omega}^j - \uparrow \varepsilon_{q\omega}^j)$ выбрать адекватную ${}^2\widehat{\mathbf{x}}_3^n(t)$ пару моделей $(\dot{m}_x^a \quad {}^{11}r^n \quad \dot{m}_y^a) | x, y = 1, \dots, N_m$ и занести ее в список Ls^u ;
- 10) $j = N_{PH} + 1$? Если «нет», то $j = j + 1$ и перейти к п. 8, иначе — к п. 11;
- 11) $l > NT$? Если «нет», то $l = l + 1$ и перейти к п. 2, иначе перейти к п. 12;
- 12) дополнить список Ls^u списком интерпретаторов и интерфейсов. Инициализировать ГиИС в соответствии с Ls^u ;
- 13) конец.

6.10. Свойства проблемно-структурной методологии функциональных гибридных интеллектуальных систем

Проблемно-структурная методология — не быстрый, трудоемкий, но верный путь решения сложных задач, заведомо гарантирующий, что создаваемые продукты будут интегрируемы в систему с новыми качествами, релевантными сложностям обработки и управления, присущими предприятиям XXI века.

Проблемно-структурная методология — это методология, когда ничто из разработанного «не выбрасывается», а аккумулируется и накапливается в интересах перехода на все более качественный уровень познания окружающего нас мира. Это адаптивная методология, которая никак не может завести увлеченного разработчика в тупик ограничений какого-либо одного метода моделирования. Более того, ПС-методология даст ему возможность управлять плюсами и сглаживать минусы методов при разработке информационных интеллектуальных систем, подстраивающихся к неоднородностям практических задач.

Проблемно-структурная методология все более активно используется для организации и эксплуатации интеллектуального управления в системах поддержки принятия решений, когда фундаментальных аналитических знаний недостаточно и необходима их коррекция статистическими закономерностями, узкопрофессиональными знаниями, зависимостями, скрытыми в эмпирических данных, выраженных лингвистически. В этой связи можно указать и еще одно свойство ПС-методологии: ее целенаправленное, осознанное и регулярное применение создает условия для преобразования накапливаемых эвристических знаний в новые, еще не открытые аналитические закономерности.

Сформулируем преимущества проблемно-структурной методологии:

- 1) ПС-методология дает принципиальную возможность решать без упрощения и идеализации сложные задачи;
- 2) ПС-методология приводит к самоорганизующимся, социальным моделям интеллектуального управления, каждый элемент которых развивается, получая данные и знания от другого элемента, что приводит к снижению затрат на разработку и эксплуатацию систем обработки информации и управления;
- 3) ПС-методология позволяет разрабатывать релевантные оригиналу модели, что подтверждается экспериментами;
- 4) ПС-методология позволяет из имеющегося в распоряжении разработчика материала создавать и экспериментировать с множеством методов,

которые будут адаптироваться к непрерывным изменениям в составе и структуре неоднородных задач. ПС-методология направляет эволюцию методов моделирования в русло, определенное естественными законами природы и развития общества, а не искусственными индивидуальными, узкоограниченными научными школами;

5) ПС-методология «сотрудничает» с перспективными направлениями моделирования решения задач человеком — распределенным интеллектом, распределенным решением задач, системами поддержки принятия решений, интеллектуальными агентами, мягкими вычислениями, что создает хорошую основу для взаимного использования идей социального, системного интеллекта;

6) ПС-методология — путь решения проблем не только сегодняшнего дня междисциплинарных схем коррекции точных аналитических знаний эмпирическими эвристиками для интеллектуального управления, но и возможность точного решения проблем в будущем за счет обнаружения новых, еще не известных фундаментальных законов развития и поведения больших социально-природно-производственных систем, где тесно переплетаются природа, человек и созданная им техника и технология;

7) ПС-методология позволяет соединять в ГиИС объективные (по крайней мере, относительно объективные) и субъективные подходы при разработке КСППР. Объективные, точные знания сдерживают безудержную фантазию интеллекта при решении творческих задач, а субъективные, относительно более легко извлекаемые, знания аппроксимируют нелинейности, неоднородности, которыми изобилует внешний мир.

6.11. Принципы разработчика функциональных гибридных интеллектуальных систем

Гибридизация в информатике — процесс чрезвычайно сложный и непредсказуемый. Сложный он, поскольку интегрирует в себе многоэтапную работу с информацией о сложных субъективных сущностях — задачах, которые выступают в качестве оригинала, о широком спектре разнообразных методов моделирования — прототипов, выступающих в качестве носителей первичных родительских признаков, комбинируемых в гибридах, и, наконец, с информацией о результатах гибридизации, т.е. тех целях, которые ставят и стремятся достичь разработчики.

Носитель задачи — субъект (специалист, эксперт) — зачастую не в состоянии без специальных приемов понять задачу, определить границы, отделяющие ее от других задач, осознать ситуацию (условия), в которых решается данная задача, выбрать или выработать метод ее решения. Все это приводит к тому, что казавшийся ранее относительно простым и выполнимым процесс постановки задачи обработки информации

и управления сегодня, переплетаясь с извлечением знаний из экспертов, чем занимается инженерия знаний, становится «камнем преткновения» на начальной стадии проектов разработки интеллектуальных систем.

Сложность гибридизации состоит еще и в том, что, несмотря на первые результаты [1, 65, 66], информатика еще не выработала до конца взгляда на низкоуровневое представление метода как сущности делимой, редуцируемой на составные части, и не выявила закономерностей, которым подчиняется наследование методами-потомками, конструируемыми из родительского «генетического материала», тех или иных существенных с точки зрения цели гибридизации первичных признаков.

Сложность гибридизации состоит и в зачастую «туманно» формулируемых целях проектирования. Теоретическая неясность того, «что?», «какой гибрид?» собираются создать, что он собой представляет, какова его модель, изначально перечеркивает все запланированные результаты. В силу популярности и «моды» на гибриды, заведомо неправильного применения специфической терминологии становится все труднее отличить «истинные гибриды» от подделок.

Сложность гибридизации состоит еще и в том, что в ее результате получается сложный программный продукт. Программные проекты часто прерываются, выходят за рамки сроков и бюджета или приводят к некачественным результатам. Разработка программного обеспечения по-прежнему — искусство, а не наука.

Непредсказуемость гибридизации объясняется тем, что часто допускаемые ошибки при разработке моделей с использованием методов математики, теории принятия решений, исследования операций, ИИ переходят и в гибриды. Поэтому потомки-результаты могут оказаться потенциально «слабее», чем методы-родители. Отсутствуют оценки вычислительной сложности гибридных алгоритмов, устойчивости решений, вырабатываемых в системах управления с гибридными моделями. Наконец, не следует забывать, что производительность ГиИС потенциально ниже в сравнении с производительностью интеллектуальных систем, работающих в режиме эксплуатации автономных моделей.

Сложность и непредсказуемость гибридизации объясняется еще и тем, что она была и остается ремеслом «генетика-информатика» в его уникальной мастерской в сравнении с инженерной наукой. Здесь уместно процитировать статью К. Бюрера «От ремесла к науке: поиск основных принципов разработки программного обеспечения» [253, с.1]: «Когда я смотрю на изумительные европейские готические кафедральные соборы, я часто спрашиваю себя, кем были архитекторы, спроектировавшие и создавшие их? Как они научились создавать постройки, которые не только прекрасны, но и достаточно прочны, чтобы в течение веков противостоять силам природы? Как они смогли построить эти зда-

ния без компьютеров, гидравлических инструментов и современных строительных материалов?

Ясно, что средневековые архитекторы обладали выдающимися знаниями в области проектирования и строительства зданий — знаниями, благодаря которым они признаны мастерами своего дела. Они были мастерами, но они не заслужили звания инженеров.

Почему? Причина проста. Хотя средневековые архитекторы обладали глубокими познаниями о способах постройки высоких зданий, они ничего не знали о законах физики. Они знали, что форма арочного потолка должна быть конической, но они никогда не слышали о дифференциальных уравнениях. Знание средневековых архитекторов было основано исключительно на опыте, полученном из проб и ошибок, и не имело никакого научного основания. Короче говоря, средневековые архитекторы знали, как делать, но не знали, почему надо делать именно так. Проектирование и строительство зданий было ремеслом, а средневековые мастера архитектуры были ремесленниками, а не инженерами.

Различие между средневековым архитектором и современным строительным инженером заключается в понимании инженером причин архитектурных правил. Даже не имея прикладного опыта, он может вывести эти правила из законов физики и, таким образом, доказать надежность своего проекта до начала строительных работ. В отличие от инженера, средневековый архитектор должен медленно и мучительно накапливать свои знания путем проб и ошибок. Постепенно он может стать мастером и передать эти знания своему ученику.

Проектирование зданий и сооружений неожиданно развилось из ремесла в инженерную дисциплину, когда оно стало строиться на научном фундаменте, основанном на законах физики. Зрелость сама по себе не превращает ремесло в инженерную дисциплину, инженерная наука требует сдвига парадигмы. Ремесло основано на пробах и ошибках, тогда как инженерная наука базируется на системе основных принципов (часто прикладных законах физики), из которых можно вывести всю систему знаний. Пробы и ошибки являются признаком ремесла, основные принципы являются признаком инженерной науки».

Признаки становления информатики как инженерной науки можно заметить еще в 1952 — 1953 гг., когда А. А. Ляпунов впервые прочитал курс «Принципы программирования» для студентов и аспирантов кафедры вычислительной математики МГУ [254].

Выработке основных принципов, которые должны быть заложены в автоматизированные системы управления на стадиях разработки, внедрения и функционирования, посвящены работы академика В.М. Глушкова [255] и его соратников.

В настоящее время даже экспресс-анализ ИНТЕРНЕТ-источников показывает, что в информатике уделяется самое серьезное внимание выработке научных принципов.

Одна из первых попыток выработать требования к любой символично - коннекционистской гибридной модели предпринята в 1993 г. [256]. Однако ориентация этих требований на дигибриды [1], отрыв от решения практических задач требуют выработки иных принципов, которые и формулируются ниже, для одного из классов гибридных интеллектуальных систем — функциональных ГиИС.

Обобщая содержание разд. 5,6, можно сформулировать несколько принципов, которые должны лежать в основе методологии и технологии разработки функциональных ГиИС. Руководствуясь этими принципами при гибридизации, мы значительно более уверены в том, что конечный результат, т.е. функциональная ГиИС, будет построена с меньшими трудозатратами и будет содержать меньше ошибок, чем если бы предлагаемые принципы разработчиком игнорировались.

1. Принцип неоднородности. При переходе к решению практических задач, возникающих не в искусственно созданном мире игровых задач, а в реально сложившемся и эволюционирующем мире, разработчик неизбежно сталкивается с многообразием парадигм, методов и переменных в науке, дисциплин в обучении, мнений и моделей внешнего мира на практике, фаз управления, целей решения задач, отношений на знаниях. Следствие такого многообразия — неоднородность сложных задач, требующая от разработчика отказа от попыток применить для имитации решения подобных задач автономных методов.

2. Принцип плюрализма. Нет одного-единственного, окончательно разработанного метода для объяснения или решения сложной, неоднородной задачи. Тем не менее, существует некоторое подмножество уже разработанного множества методов и моделей, которое может быть использовано для моделирования ее решения.

3. Принцип системного анализа сложной задачи. Решению сложной задачи должен предшествовать системный анализ ее свойств, состава и структуры, что позволяет сделать более отчетливыми границы однородных областей и подобрать релевантные этим областям автономные методы. Вывод о применении для решения задачи гибридов делается по результатам ее системного анализа. Решение сложных задач должно быть рассмотрено на более чем одной страте, что приводит к множеству простых моделей для решения одной и той же подзадачи, но с использованием многоаспектности, разнообразия точек зрения.

4. Принцип конструирования. Метод решения сложной задачи синтезируется из методов, моделей, модулей, инструментальных средств и технологий всякий раз заново, когда возникает необходимость решения

сложной задачи. Приступая к конструированию, необходимо знать плюсы и минусы методов и инструментариев-кирпичиков, из которых строятся функциональные ГиИС, надежность знаний о подзадаче, цену решения подзадачи тем или иным методом или инструментарием.

5. Принцип приоритета знаний. Функциональная ГиИС должна быть построена таким образом, чтобы первый приоритет в решении сложной задачи отдавался точным знаниям и жестким вычислениям и только второй — эвристическим знаниям и мягким вычислениям. При этом использование эвристик должно рассматриваться как коррекция решения, полученного на точных знаниях. Если точные знания отсутствуют, то возможно применение одних эвристик, однако тогда должна быть поставлена цель получения таких точных знаний.

6. Принцип постепенности. Прежде чем разрабатывать ФГиИС, необходимо накопить опыт и знания построения или использования автономных моделей. Это позволит избежать ошибок в автономных моделях, которые неизбежно перейдут в ГиИС и снизят ее качество.

7. Первый принцип наследования. Чтобы ФГиИС унаследовала сумму плюсов методов-прототипов, необходимо, чтобы степень зернистости и интерфейсы, реализующие отношения интеграции, были установлены как на уровне декларативных, так и процедурных представлений.

8. Второй принцип наследования. Архитектура функциональных ГиИС наследуется как от состава и структуры сложной задачи, так и от плюсов и минусов автономных методов. Отказ от релевантности проблеме ведет к хорошим, но бесполезным гибридам. Отказ от учета плюсов и минусов автономных методов ведет к снижению качества ГиИС.

9. Принцип самоорганизации. Функциональная ГиИС должна обучаться и извлекать знания из одного элемента для совершенствования других элементов. Это обеспечивает адаптацию ФГиИС к новым условиям и снижает трудозатраты на ее эксплуатацию.

10. Принцип полноты. Функциональная ГиИС должна строиться с использованием как можно большего числа классов автономных методов: аналитических, статистических, символьных, коннекционистских и эволюционных.

11. Принцип снижения производительности. Функциональная ГиИС может использоваться там, где не существенно снижение производительности вычислений по сравнению с автономными моделями.

Данные принципы не претендуют на универсальный и строгий характер математических или логических правил. Эти принципы сформулированы в результате исследований опыта разработки ГиИС в мировой практике и личного двадцатилетнего опыта авторов. Использование этих принципов позволит делать ГиИС там, где это целесообразно, и

такими, какими они должны быть, и работающими так, чтобы не разочаровывать своих создателей.

7. ТЕХНОЛОГИЯ РЕШЕНИЯ СЛОЖНЫХ ЗАДАЧ МЕТОДАМИ ФУНКЦИОНАЛЬНЫХ ГИБРИДНЫХ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

7.1. Понятие проблемно-структурной технологии функциональных гибридных интеллектуальных систем

Важная функция науки — порождение возможных технологий (одного из факторов выработки новых знаний) [257]. Современная наука заинтересована в создании системы правил человеческой деятельности по производству таких результатов, которые сами по себе, в рамках действия стихийных природных процессов, появиться не могут.

В [257] рассмотрено несколько определений понятия «технология».

1. Технология (по Д. Гелбрейту) — систематизированное применение научного (организованного) знания для решения практических задач.

2. Технология (по С.С. Гусеву) — некоторый способ человеческого отношения к окружающей действительности, порожденный практической ориентированностью познания.

3. Технология (по В.П. Каширину) — прогрессирующая и управляемая человеком природно-социальная совокупность процессов целенаправленного изменения различных форм вещества, энергии, информации, протекающая в различных системах с их специфическими законами строения и функционирования.

4. Технология (по А.И. Ракитову) — охватывает: инструментальную среду; совокупность операционных процедур; систему деятельности, детерминированную инструментальной средой и влияющую на нее; систему управления соответствующей деятельностью; совокупность последствий (социальных и экономических); информационную среду, в которой эта деятельность осуществляется.

Обобщая приведенные выше определения технологии, можно определить понятие «технология функциональных ГиИС».

Технология функциональных гибридных систем (рис. 7.1) охватывает инструментальную среду, совокупность, систему операций информатики по системному анализу сложных задач и синтезу функциональных гибридных интеллектуальных систем, систему управления деятельностью разработчика и информационную среду этой деятельности.

ПС-технология разработки ФГиИС определяется как:

$${}^3X_T^n = \langle {}^3X_o^n, \Pi^o, L^S, {}^9X_3^0, L^U \rangle, \quad (7.1)$$

где ${}^3X_o^n$ — операции технологии; Π^o — логическая последовательность операций; L^S — информационный язык функциональных ГиИС (информационная среда); ${}^9X_3^0$ — множество программных средств поддержки выполнения операций (инструментальная среда); L^U — система управления деятельностью разработчика.

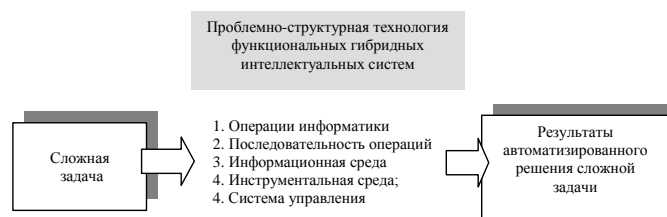


Рис. 7.1 — Понятие технологии функциональных гибридных интеллектуальных систем

Технологию, построенную в соответствии с проблемно-структурной методологией функциональных гибридных интеллектуальных систем (разд. 6), будем называть проблемно-структурной технологией (ПС-технологией). Цель ПС-технологии — получение результатов автоматизированного решения сложных в соответствии с проблемно-структурной методологией функциональных ГиИС.

Определим основные составляющие ПС-технологии.

Совокупность операций — операционные процедуры, детализирующие этапы деятельности разработчика и экспертов и уточняющие операции по входу, выходу, содержанию, файлам данных и знаний.

Логическая последовательность операций — взаимосвязанная совокупность действий разработчика по преобразованию информации об объектах-оригиналах и объектах-прототипах в объекты-результаты.

Инструментальная среда — программное обеспечение для выполнения операций системного анализа сложных задач, синтеза и эксплуатации функциональных ГиИС.

Информационная среда — информационный язык, знаки которого обозначают отношения, ресурсы, свойства, действия, их иерархии, структуры, ситуации, однородные и неоднородные задачи, автономные

методы, крупно- и мелкозернистые элементы функциональных ГиИС, функциональные ГиИС, их поведение, операции информатики и их логическую последовательность.

Система управления деятельностью разработчика — программное обеспечение для управления деятельностью разработчика функциональных ГиИС в соответствии с их жизненным циклом (рис. 6.1).

Ниже рассмотрены вопросы, раскрывающие содержание всех пяти составляющих ПС-технологии.

7.2. Информационная среда технологии

Приведем несколько определений.

Язык искусственный — семиотическая система для автоматической и автоматизированной переработки информации [143].

Язык информационный — искусственный язык, предназначенный для записи семантической информации с целью ее последующего использования в информационно-поисковых и информационно-логических системах [143].

Тогда информационный язык функциональных ГиИС — искусственный язык, предназначенный для записи семантической информации об объектах-оригиналах (сложных задачах), объектах-прототипах (автономных методах решения задач), объектах-результатах (функциональных ГиИС), их поведении, операций информатики по разработке функциональных ГиИС и логической последовательности операций.

Информационный язык функциональных ГиИС имеет пять основных предназначения (рис. 7.2). Первое назначение (на рис. 7.2 — ИЯ(1)) — средство коммуникации между членами коллектива разработчиков функциональной ГиИС,

между коллективами разработчиков (обмен опытом и знаниями) и между разработчиками и экспертами в системе поддержки принятия решений (извлечение знаний, изучение опыта, сбор и предварительная обработка информации).

Второе назначение (ИЯ(2)) — средство структурирования реального мира экспертами в системе поддержки принятия решений.

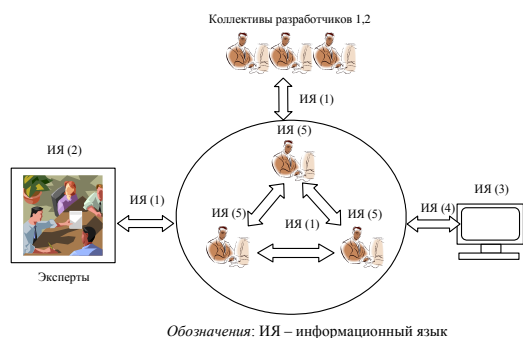


Рис. 7.2 — Предназначение информационного языка функциональных гибридных интеллектуальных систем

ного мира экспертами в системе поддержки принятия решений.

Третье назначение (ИЯ(3)) — средство компьютерного представления модели внешнего мира, данных, фактов, знаний, опыта.

Четвертое назначение (ИЯ(4)) — средство взаимодействия в человеко-машинной системе. Пятое назначение (ИЯ(5)) — средство структурирования реального мира, а также мира методов и средств информатики разработчиками функциональной ГиИС.

Множество рассмотренных предназначений информационного языка можно обобщить в три: 1) средство коммуникации между людьми; 2) средство коммуникации в человеко-машинной системе; 3) средство структурирования и представления информации.

Язык профессиональной деятельности в системах поддержки принятия решений представляет собой полиязык [1]. С одной стороны, это обусловлено присущей языку структурированностью внешнего мира. Это приводит к необходимости представления в информационном языке знаков, обозначающих ресурсы, свойства, действия, структуры, ситуации, состояния, поведение, а с учетом деятельности субъекта управления — целей, задач, планов, оценок. В работе [1] информационный язык представлен семейством таких языков описания ресурсов, операций, структур, ситуаций, состояний, поведения объекта управления, а также целей, планов и задач. В этом случае у разработчика есть набор средств-компонентов для конструирования из них метаязыка, описывающего решение сложной задачи комбинацией нескольких взаимосвязанных процессов рассуждений на разных языках.

С другой стороны, полиязыковый характер языка профессиональной деятельности — следствие эволюции систем управления в сторону многомодельных, гибридных и гибридных адаптивных систем управления [1]. Это приводит к узкой специализации управленцев-экспертов по профессиональным нишам и к тому, что информация в СППР представляется на широком спектре языков профессиональной деятельности со своими относительно независимыми задачами, лексикой, данными, знаниями, принципами [1]. Стратификация по нишам при разработке функциональных ГиИС имеет целью снизить сложность задач, разбив их на простые элементы — подзадачи, рассматриваемых в локальных подобластях мира управления — профессиональных нишах.

В разд. 6.3 сформулирован метод решения сложных задач в системах поддержки принятия решений. Суть метода состоит в том, что решение сложной задачи рассматривается как итерационный процесс на семействе относительно простых моделей-страт, что позволяет лучше понять задачу, снизить ее сложность, с последующим комбинированием нишевых (автономных) моделей в функциональную ГиИС и получением в результате наблюдений за ее поведением информации для решения задачи.

Семейства таких моделей-страт могут быть получены в аксиоматической теории Te (5.1).

На рис. 7.3 такое семейство изображено в виде многоуровневой схемы ролевых концептуальных моделей для системного анализа сложных задач и синтеза функциональных ГИИС.

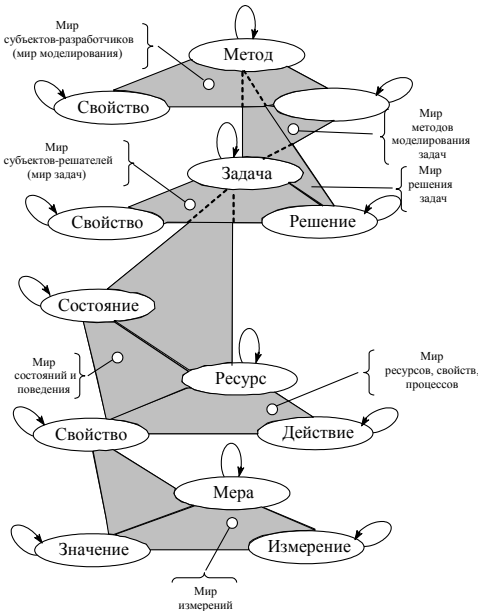


Рис. 7.3 — Многоуровневая схема ролевых концептуальных моделей для системного анализа сложных практических задач и синтеза функциональных ГИИС

Эта схема сконструирована из девяти трехролевых конструктов (разд. 5.2), часть которых (они показаны различными видами штриховки) задает межуровневое взаимодействие. В общей сложности выделено семь взаимосвязанных и взаимодействующих друг с другом схем ролевых концептуальных моделей, которые называются моделями миров, чтобы подчеркнуть определенную связь с моделями реального мира экспертов, ЛПР в СППР и разработчика.

На нижнем уровне показана модель мира измерений значений свойств-параметров. От совершенства измерений во многом зависит и ка-

чество решений в СППР. Мир измерений связан с миром ресурсов, свойств и процессов трехролевой концептуальной моделью «значение — мера — свойство», изображенной в вертикальной плоскости, что обеспечивает перевод информации в другой мир, представляющий и манипулирующий данными и знаниями уже в другой ролевой концептуальной модели.

Существование еще одного мира — состояний и поведения — завершает картину объекта управления для СППР. Может показаться странным, что этот мир настолько прост, что представлен лишь состояниями (ситуациями). На самом деле это не так. Достаточно обратить внимание не только на концепты, но и на отношения «состояние — со-

стояние». Именно они и позволяют определить состояние объекта в момент времени $t + 1$ по состоянию в момент времени t . Именно эти отношения — знания о функционировании объекта значимы для поддержки принятия решений. Применение многоуровневых схем ролевых концептуальных моделей при разработке функциональных ГИИС приводит к языку описания состояния объекта управления (рис. 7.4):

$$L^S = L_1^S \cup L_2^S \cup L_3^S \cup L_4^S \cup L_5^S, \quad (7.2)$$

где $L_1^S, L_2^S, L_3^S, L_4^S, L_5^S$ — информационные языки описания производных отношений, ресурсов—свойств—действий, иерархий ресурсов—свойств—действий, пространственных и операционно-технологических действий, ситуаций и состояний соответственно.

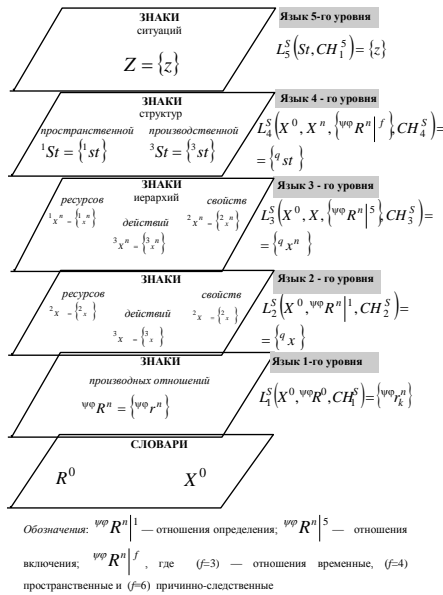


Рис. 7.4 — Иерархия информационного языка для представления ресурсов, действий, их свойств, иерархий, структур, ситуаций и состояний

На рис. 7.4 изображен «слоеный пирог» описания состояний объекта управления для системы поддержки принятия решений. В первом слое расположены словари понятий и отношений — концептуальный базис информационного языка. Над этим базисом строится семейство упорядоченных по уровням языков описания. Язык первого уровня L_1^S использует эвристические правила CH_1^S для построения из X^0 и R^0 знаков производных (составных) отношений $\psi^{\phi} r^n \in \psi^{\phi} R^n$, где ψ, ϕ выбираются в соответствии с классификацией R^0 [1], $n > 0$. На этом уровне можно получить такие знаки $\psi^{\phi} r^n$, как «иметь

длину», «быть на расстоянии 30 м от» и др. В языке второго уровня L_2^S эвристики CH_2^S используются, чтобы обозначить ресурсы, действия и свойства без учета их иерархичности с помощью множества знаков ${}^q x \in {}^q X$ отношениями определения. На третьем уровне отношениями

включения формализуют иерархии знаков второго уровня, на четвертом — пространственные, операционально-технологические структуры и, наконец, на пятом уровне — знаки-ситуации и знаки-состояния.

Субъекты-решатели (эксперты и лицо, принимающее решение) в схеме многоуровневой концептуальной модели представлены мирами задач и решения задач. Мир задач показан упрощенно, только задачами, процессами решений и их свойствами. Особый интерес вызывает спецификация задачи на множестве свойств, т.е. исследование отношений «задача-свойства». Этот вопрос решен разработкой схем ролевых концептуальных моделей «однородная задача» и «неоднородная задача» (5.7, 5.10) и введением соответствующих знаков информационного языка (7.3). Решая задачи, субъект рассуждает, используя при этом формализованные, частично формализованные и неформализованные знания, что переводит его деятельность в плоскость мысленных модельных экспериментов. Именно анализ и выявление этих методов манипулирования широким спектром знаний и представляют значительный интерес для разработки функциональных ГиИС.

Особого внимания заслуживает исследование отношений «задача-задача» и «решение-решение». Очевидно, первые связаны с открывающейся возможностью системного анализа понятия «задача», с одной стороны, и уменьшением сложности задач, применяя редукцию, с другой стороны (разд. 6). Отношения «решение-решение» отображают порядок решения задачи. Их анализ позволит ответить на вопросы о том, из каких частей состоит процесс решения задач человеком, какие подпроцессы можно выделить в этом еще недостаточно хорошо изученном явлении (разд. 2.4). В конечном итоге по этим отношениям редуцируются действия, чтобы изучить сложные процессы как систему из более простых частей. Отношения «решение-решение» исследованы в разд. 6.3.

Очевидно, что мир решений должен быть связан еще с одним миром — исполнению, реализации и решения задач соответствующими операциями в объекте управления. Мир исполнения не показан на рис. 7.3, однако это не значит, что схема его ролевых концептуальных моделей проще, чем для других миров.

Наконец, на верхнем уровне показан еще один мир — мир моделирования. Это уже мир разработки функциональных ГиИС и систем обработки информации и управления. Основные роли в этом мире играют субъекты моделирования — разработчики, владеющие некоторым количеством, а может быть, одним-единственным методом решения задач. Для понимания действий в мире методов моделирования важны отношения «метод-свойства» (разд. 5.4). Исследование этих отношений позволяет не только специфицировать метод как сущность, но и выявить

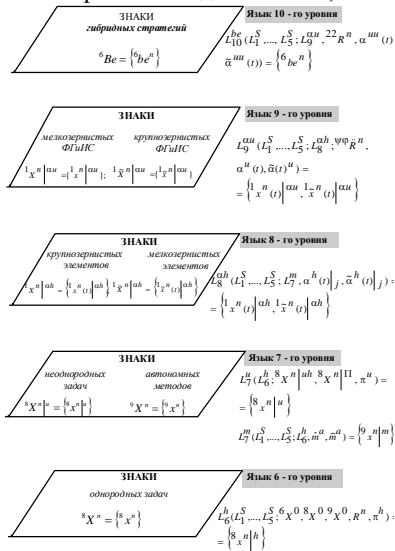
сильные и слабые стороны каждого из них, что особенно актуально для гибридизации. Мир методов моделирования задач — не застывший, а эволюционирующий мир, в котором появляются новые методы и забываются уже известные, не выдержавшие проверки временем и практикой.

В каждом из рассмотренных миров есть свои понятия, которые связаны, с одной стороны, своими денотатами с реалиями внешнего мира, а с другой стороны, установлено их соответствие со знаками информационного языка функциональных ГиИС. Такое соответствие позволяет, с одной стороны, эвристически направлять системный анализ языков профессиональной деятельности, инструмент которого — схемы многоуровневых концептуальных моделей. С другой стороны, такое соответствие делает функционирование ГиИС релевантным поведению системы поддержки принятия решений и поведению объекта управления.

На рис. 7.5 показана иерархия информационного языка функциональных ГиИС:

$$L^{ИЯ} = L^S \cup L_6^h \cup L_7^u \cup L_7^m \cup L_8^{ah} \cup L_9^{au} \cup L_{10}^{be}. \quad (7.3)$$

В иерархии на рис. 7.5 семейство информационных языков упорядочено по уровням с 6-го по 10-й (уровни 1 — 5 рассмотрены на рис. 7.4), на которых последовательно, начиная с первого, конструируются знаки,



обозначающие отношения, ресурсы, свойства, действия, их иерархии, структур, ситуаций, однородных задач, неоднородных задач и автономных методов, крупно- и мелкозернистых элементов ГиИС, функциональных ГиИС и их поведения.

Информационный язык ГиИС $L^{ИЯ}$ (7.5) — инструмент разработчика функциональных ГиИС, использующего гибридизацию для создания интеллектуальных систем. Это также язык для обмена информацией между различными коллективами проектировщиков в рамках междисциплинарной научной области «функциональные ГиИС». Это

позволяет сделать методологию и технологию гибридизации не уникальным явлением, а широко используемым инструментом решения сложных задач на практике и, что особенно важно, — предметом осмысления, оценки и развития другими учеными и создания новых, более совершенных информационных языков, новых методологий и технологий.

Язык L^{IIA} — это язык не только системного анализа неоднородных проблемных сред (6.2), т.е. сред существования и эволюции объектов-оригиналов, методов исследования операций, теории принятия решений, искусственного интеллекта, т.е. сред существования и эволюции объектов-прототипов, но и синтеза моделей-систем и методов-систем за рамками ограничений классов методов из автономного базиса (6.17), (6.18). При этом как анализ, так и синтез основываются на концептуально полной для структурирования внешнего мира тетраде «ресурс — свойство — действие — отношение» (разд. 5.2).

Язык L^{IIA} снимает многие терминологические и гносеологические проблемы, может использоваться для описания эволюционных процессов в мирах задач и методов.

Язык L^{IIA} — это единый инструментарий концептуального описания данных и аналитических, статистических, логических, лингвистических, генетических знаний и опыта.

7.3. Операции технологии

В табл. 7.1 — 7.21 приведено описание двадцати одной операции ${}^3X_o^n \in {}^3X_o^n$ (7.1) ПС-технологии разработки функциональных ГиИС (${}^3X_o^n$), в ходе которых извлекается, представляется и преобразовывается информация об объектах-оригиналах (сложных задачах) и объектах-прототипах (автономных методах) в объекты-результаты (функциональные ГиИС).

Операции пронумерованы, названы, привязаны к этапам проблемно-структурной методологии (рис. 6.1), описано их содержание, входная и выходная информация. Каждая операция рассмотрена и с точки зрения системы управления деятельностью разработчика — L^U (7.1), т.е. отношений следования: какой схемой ролевых концептуальных моделей такая деятельность направляется, какие операции предшествуют специфицируемой операции и какие операции следуют за ней в логической последовательности действий разработчика (разд. 7.4). Кроме этого, при описании входной и выходной информации указано на используемые модели, базы данных и знаний, таблицы, списки, фрейм-представления,

с примерами которых подробнее можно познакомиться в [1]. С обозначениями, использованными в табл. 7.1 — 7.21, можно познакомиться в списке сокращений.

Таблица 7.1 — Описание операции идентификации свойств сложной задачи

<p><i>Операция №1</i> <i>Название:</i> «Идентификация свойств сложной задачи» <i>Этап проблемно-структурной методологии:</i> «Идентификация неоднородной задачи» <i>Система управления деятельностью разработчика:</i> 1) операция является началом работ; 2) обработка информации направляется частью 1 (до серой заливки) схемы ролевых концептуальных моделей «неоднородная задача» (5.10)</p>		
Содержание операции	Входная информация	Выходная информация
<p>Анализ, увеличение уровня понимания свойств сложной задачи и создание в памяти компьютера ее модели «неоднородная задача»</p>	<ol style="list-style-type: none"> 1. Цель решения задачи; 2. Исходные данные для решения задачи; 3. Условия решения задачи; 4. Фаза, на которой решается задача (учет, контроль, анализ, нормирование, прогнозирование, планирование, регулирование, организации); 5. Класс переменных, которыми необходимо и достаточно манипулировать при решении задачи (детерминированные, стохастические, нечеткие или четкие лингвистические, генетические); 6. Класс задачи (размещения, расписания, маршрутизации, совершенствования и др. (5.10); 7. Идентификатор сложной задачи (5.8); 8. Личностные характеристики ЛПР; 9. Оценки экспертов по результатам решения или подзадач из состава сложной задачи; 10. Оценки результатов операций, исполняющих решение сложной задачи 	<p>В языке описания неоднородных задач L_7^u (7.3), как подмножестве информационного языка технологии ФГиИС, по части первой схемы π^u (5.10) конструируется знак $8 X^n u \in 8 X^n u$ (5.11), рис. 7.5</p>

Таблица 7.2 — Описание операции редукции сложной задачи

<p><i>Операция № 2</i> <i>Название:</i> «Редукция сложной задачи» <i>Этап проблемно-структурной методологии:</i> «Редукция неоднородной задачи» <i>Система управления деятельностью разработчика:</i> 1) операция выполняется после операции 1 — «идентификация свойств сложной задачи»; 2) обработка информации направляется частью 2 (показано серой заливкой) схемы ролевых концептуальных моделей «неоднородная задача» (5.10)</p>		
Содержание операции	Входная информация	Выходная информация
<p>Анализ, увеличение уровня понимания состава и структуры сложной задачи и создание в памяти компьютера ее модели «неоднородная задача»</p>	<ol style="list-style-type: none"> 1. Состав сложной задачи (множество подзадач); 2. Множество декомпозиций сложной задачи 	<p>Знак $8 X^n u \in 8 X^n u$ (5.11), рис. 7.5</p>

Таблица 7.3 — Описание операции спецификации подзадачи сложной задачи

<p><i>Операция № 3</i> <i>Название:</i> «Спецификация подзадачи сложной задачи» <i>Этап проблемно-структурной методологии:</i> «Спецификация однородной задачи» <i>Система управления деятельностью разработчика:</i> 1) операция выполняется после операции 2 — «Редукция сложной задачи»; 2) обработка информации направляется схемой ролевых концептуальных моделей «однородная задача» (5.7)</p>		
Содержание операции	Входная информация	Выходная информация
<p>Анализ, увеличение уровня понимания подзадачи из состава сложной задачи и создание в памяти компьютера ее модели «однородная задача»</p>	<ol style="list-style-type: none"> 1. Цель решения задачи; 2. Исходные данные для решения задачи; 3. Условия решения задачи; 4. Фаза, на которой решается задача (учет, контроль, анализ, нормирование, прогнозирование, планирование, регулирование, организация); 5. Класс переменных, которыми необходимо и достаточно манипулировать при решении задачи (детерминированные, стохастические, нечеткие или четкие лингвистические, генетические); 6. Класс задачи (размещения, расписания, маршрутизации, совершенствования и др. (5.7)); 7. Идентификатор подзадачи (5.8); 8. Свойства среды разработки, проблемной среды, проекта, среды экспертов, измерений, информации, плана решения задачи; 9. Отношение включения подзадачи в сложную задачу; 10. Отношения с другими подзадачами их состава сложной задачи 	<p>В языке описания однородных задач L_6^h как подмножестве информационного языка ПС-технологии ФГиИС по схеме π^h (5.7) конструируется знак $8_x^n h \in 8_X^n h$ (5.9), рис. 7.5</p>

Таблица 7.4 — Описание операции установления соответствия неоднородной задачи и однородных задач

<p><i>Операция № 4</i> <i>Название:</i> «Установление соответствия неоднородной и однородных задач» <i>Этап проблемно-структурной методологии:</i> «Разработка таблицы гибридных стратегий» <i>Система управления деятельностью разработчика:</i> операция выполняется после завершения операции 3. Операция выполняется до начала выполнения операции 5</p>		
Содержание операции	Входная информация; БД, БЗ	Выходная информация
<p>Установление соответствия между исходными данными и целями неоднородной задачи (5.10) и целями и исходными данными однородных задач (5.7) из декомпозиции $\hat{\pi}^u$ (5.10). Работы выполняются для каждой декомпозиции $\hat{\pi}^u \in \hat{\Pi}^u$ (5.10) и извлечения из ЛППР и экспертов и представления компьютера знаний о том, каким однородным задачам передаются как исходные данные ${}^6D^h$ (5.7) исходные данные неоднородной задачи ${}^6D^u$ (5.10) и какие цели решения однородных задач ${}^6G^h$ (5.7) — цели решения неоднородной зада-</p>	<p>Фрейм декомпозиции неоднородной задачи</p>	<p>Фрейм декомпозиции, пополненный информацией о соответствии исходных данных и целей неоднородной задачи и целей и исходных данных для однородных задач</p>

чи ${}^6G^u$ (5.10)		
---------------------	--	--

Таблица 7.5 — Описание операции спецификации связей неоднородной задачи

<p><i>Операция № 5</i> <i>Название:</i> «Спецификация связей неоднородной задачи» <i>Этап проблемно-структурной методологии:</i> «Спецификация связей неоднородных задач» <i>Система управления деятельностью разработчика:</i> операция выполняется после завершения операции 4, если решается несколько взаимосвязанных по «входу-выходу» неоднородных задач, либо если некоторые сложные задачи уже решаются (эксплуатируются) в автоматизированном режиме и настоящая задача требует предварительного решения одной или нескольких других задач. Операция выполняется до начала выполнения операции 6</p>		
Содержание операции	Входная информация Базы данных, Базы знаний	Выходная информация
<p>Установление соответствия «выход ${}^6G^h$ (5.7) одной неоднородной задачи — вход ${}^6D^h$ (5.7) другой неоднородной задачи»</p>	<p>1. Множество знаков неоднородных задач в информационном языке L_7^u (5.11), рис. 7.5; 2. Информация от экспертов об установлении отношений «задача — задача» ${}^{88}R_4^n$ (5.10) на множестве сложных задач; 3. Информация об установлении соответствия «выход одной задачи — вход другой задачи» (цель — исходные данные) на множестве сложных задач</p>	<p>1. Модель мира неоднородных задач ${}^5W^u$ (6.14); 2. Таблица связей неоднородных задач (СН — таблица)</p>

Таблица 7.6 — Описание операции выбора класса методов решения подзадачи сложной задачи

<p><i>Операция № 6</i> <i>Название:</i> «Выбор класса метода решения подзадачи сложной задачи» <i>Этап проблемно-структурной методологии:</i> «Выбор автономных методов» <i>Ограничения:</i> операция выполняется для шести классов базисных методов (табл. 5.1); <i>Система управления деятельностью разработчика:</i> 1) операция выполняется после завершения операции 5; 2) обработка информации направляется схемой (5.7); 3) операция может выполняться и после операции 3 — «Спецификация подзадачи сложной задачи»; 4) операция выполняется для каждой подзадачи из каждой декомпозиции сложной задачи; 5) операция выполняется до операции 8 — «Проверка однородности сложной задачи»</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Установление соответствия на множествах подзадач из состава сложной задачи и классов базисных методов. 1) Функции разработчика — ответы на вопросы по спецификатору однородной задачи; выбор режима консультаций; принятие решения о выборе класса базисных методов; принятие решения о выборе конкретного метода внутри базисного класса; 2) Функции компьютера: обработка информации из</p>	<p>1. Спецификатор $({}^2Q^h)$ однородной задачи: свойства среды разработки, проблемной среды, проекта, среды экспертов, плана решения задачи, параметры измерений, действий и информации; 2. БД вопросов к разработчику для конструирования спецификатора ${}^2Q^h$; 3. БЗ «класс задач — класс методов»; 4. Режим консультаций: 1) обобщенные оценки «класс методов - значение оценки»; 2) детализированные оценки «группа свойств спецификатора - класс методов - значение оценки релевантности»;</p>	<p>Фрейм-представление знака неоднородной задачи, в котором знак однородной задачи пополнен концептуальной моделью выбранного в ходе выполнения данной операции класса методов и идентификатором конкретного метода (или нескольких методов) внутри класса базисных методов</p>

БД и БЗ; выдача консультаций в виде балльных оценок	5. Идентификатор конкретного метода внутри класса базисных методов, выбранного разработчиком	
---	--	--

Таблица 7.7 — Описание операции кластеризации подзадач сложной задачи

<p><i>Операция № 7</i> <i>Название:</i> «Кластеризация подзадач сложной задачи» <i>Этап проблемно-структурной методологии:</i> «Спецификация однородных задач» <i>Система управления деятельностью разработчика:</i> операция выполняется после завершения операции 6 и до начала выполнения операции 8</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Выявление среди специфицированных подзадач из состава сложной задачи групп (кластеров) подзадач, схожих на свойствах спецификатора $2Q^h$:</p> <p>1) Функции разработчика — анализ кластеров на множестве спецификаторов однородных задач; 2) Функции компьютера: поиск кластеров на множестве спецификаторов однородных задач из состава неоднородной задачи</p>	<p>Множество спецификаторов $(2Q^h)$ однородных задач из состава неоднородной задачи</p>	<p>Фрейм-представление неоднородной задачи, пополненное информацией «кластер — однородная задача»</p>

Таблица 7.8 — Описание операции проверки неоднородности сложной задачи

<p><i>Операция № 8</i> <i>Название:</i> «Проверка неоднородности сложной задачи» <i>Этап проблемно-структурной методологии:</i> «Проверка неоднородности» <i>Система управления деятельностью разработчика:</i> операция выполняется после завершения операций 6 и 7, до начала выполнения операции 9. Если сложная задача удовлетворяет модели «неоднородная задача», то выполняется операция 9. Если сложная задача не удовлетворяет модели «неоднородная задача», то дальнейшие работы по ПС-технологии функциональных ГиИС должны быть прекращены</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Выполняется, чтобы повысить у разработчика степень уверенности в том, что исходная задача может быть отнесена к модели «неоднородная задача». Решение о неоднородности сложной задачи принимает разработчик по результатам операций 6 и 7.</p> <p>Рекомендации для принятия решений. Признаки прекращения работ по ПС-технологии: 1) в результате операции 6 однородные задачи из состава неоднородной задачи получили близкие значения балльных оценок в пользу одного и того же класса базисных методов; 2) в результате выполнения операции 7 кластеры отсутствуют либо выявлен только один кластер. Иначе нет причин отказываться от продолжения работ по ПС-технологии функциональных ГиИС</p>	<p>Фрейм-представление неоднородной задачи</p>	<p>Решение разработчика о том, удовлетворяет или нет сложная задача модели «неоднородная задача»</p>

Таблица 7.9 — Описание операции разработки и тестирования автономной модели

<p><i>Операция № 9</i> <i>Название:</i> «Разработка и тестирование автономной модели» <i>Этап проблемно-структурной методологии:</i> «Разработка автономных моделей» <i>Система управления деятельностью разработчика:</i> операция выполняется после завершения операции 8, до начала выполнения операции 10. Если в ходе выполнения данной операции будет создано функционально избыточное множество моделей, то обязательно выполнение и операции 10 - «Спецификация области релевантности модели»</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Разработка и тестирование модели, алгоритма и компьютерной программы решения подзадачи из состава сложной задачи. Операция выполняется по методологиям элементов ФГИИС [1]. Для организации «спора моделей» в ходе синтеза функциональной ГИИС (операция 15) рекомендуется разработка множества моделей, алгоритмов и программ, что увеличивает релевантность синтезируемых ФГИИС сложным задачам</p>	<ol style="list-style-type: none"> 1. Фрейм-представление неоднородной задачи; 2. Модель «однородная задача» моделируемой подзадачи из состава сложной задачи; 3. Информация для моделирования в соответствии с технологией работ по выбранному в результате операции методу (ам); 4. Информация, необходимая для тестирования модели (лей) 	<ol style="list-style-type: none"> 1. Модель в виде компьютерной программы решения подзадачи сложной задачи; 2. Функционально избыточное множество моделей в виде программ, разработанных одним методом, либо разными, но из одного класса базисных методов. 3. Список моделей (ГМ - список); 4. Список входных и выходных параметров по каждой модели (П - список)

Таблица 7.10 — Описание операции разработки области релевантности автономной модели

<p><i>Операция № 10</i> <i>Название:</i> «Разработка области релевантности автономной модели» <i>Этап ПС- методологии:</i> «Разработка областей релевантности автономных моделей» <i>Система управления деятельностью разработчика:</i> операция выполняется после завершения операции 9, если в ходе работ по выполнению операции 9 было разработано функционально избыточное множество моделей, алгоритмов и программ. Операция выполняется до начала выполнения операции 11</p>		
Содержание операции	Входная информация Базы данных, Базы знаний	Выходная информация
<p>Для моделей, разработанных экспертами операцией 9, выбираются существенные входные параметры, для которых задаются интервалы значений, в которых модель релевантна процессам и явлениям в моделируемой системе (6.19). При этом настраивается нечеткая система для операции 15</p>	<ol style="list-style-type: none"> 1. Список моделей (ГМ - список) разработанных операцией 7; 2. Список входных и выходных параметров по каждой модели (П - список); 3. Оценки экспертов, решающих подзадачи сложной задачи, о значимости входных параметров модели; 4. Оценки экспертов, решающих подзадачи сложной задачи интервалов релевантности модели оригиналу 	<ol style="list-style-type: none"> 1. Таблица опроса экспертов (Э - таблица); 2. Список существенных параметров для данной модели (ПС — список); 3. Список пар «модель — интервалы» (МИ — список); 4. База знаний KB^{nlx} (6.19) нечеткой системы, используемой в операции 15

Таблица 7.11 — Описание операции разработки интерфейсов автономных моделей

<p><i>Операция № 11</i> <i>Название:</i> «Разработка интерфейсов автономной модели» <i>Этап проблемно-структурной методологии:</i> «Разработка интерфейсов автономных моделей» <i>Система управления деятельностью разработчика:</i> операция выполняется после завершения операции 10. Операция выполняется до начала выполнения операции 12</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Для каждой модели из множества моделей, разработанных посредством операции 9 и с учетом множества отношений интеграции (указанных в таблице гибридных стратегий), связывающих данную модель с другими автономными моделями, разрабатываются интерфейсные компьютерные программы, реализующие гибридизацию на уровнях «данные-данные», «знания-знания»</p>	<ol style="list-style-type: none"> 1. Список моделей (ГМ - список); 2. Список входных и выходных параметров по каждой модели (П - список); 3. Таблица гибридных стратегий (Т-таблица); 4. Базы данных (база знаний), используемые автономными моделями 	<ol style="list-style-type: none"> 1. Компьютерные программы интерфейсов автономных моделей; 2. Список интерфейсов автономных моделей (ИМ - список)

Таблица 7.12 — Описание операции разработки таблицы гибридных стратегий

<p><i>Операция № 12</i> <i>Название:</i> «Разработка таблицы гибридных стратегий» <i>Этап проблемно-структурной методологии:</i> «Разработка таблицы гибридных стратегий» <i>Система управления деятельностью разработчика:</i> операция выполняется после завершения операции 11. Операция выполняется до начала выполнения операции 13</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Анализ декомпозиции неоднородной задачи и замена отношений декомпозиции ${}^{88}R_3^n$ (5.10), которые установлены между однородными задачами (см. операцию 1), на отношения интеграции $\Phi \ddot{R}^n$ (6.15) автономных моделей: извлечения, включения, дополнения, сравнения, аргументации, обобщения, управления. Одно отношение декомпозиции может заменяться одним или несколькими отношениями интеграции. При этом автономные модели упорядочиваются в одну или несколько интегрированных моделей (6.18)</p>	<ol style="list-style-type: none"> 1. Фрейм декомпозиции неоднородной задачи; 2. Список отношений интеграции; 3. Список моделей (ГМ — список) 	<p>Таблица гибридных стратегий (Т-таблица): «декомпозиция неоднородной задачи — однородная задача — однородная модель — интегрированная модель»</p>

Таблица 7.13 — Описание операции разработки технологических компонентов

<p><i>Операция № 13</i> <i>Название:</i> «Разработка технологических компонентов» <i>Этап проблемно-структурной методологии:</i> «Разработка таблицы гибридных стратегий» <i>Система управления деятельностью разработчика:</i> операция выполняется после завершения операции 12. Операция выполняется до начала выполнения операции 14</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Разрабатываются технологические компоненты (разд. 5.5), поддерживающие на программном уровне работу группы функциональных элементов — автономных моделей из состава интегрированной модели (6.15). Необходимость в создании технологических элементов возникает при редукции сложной практической задачи по информационно-управляющим отношениям (6.21). В этом случае возникает ситуация, когда в интегрированной модели используются отношения управления $(\Phi \ddot{R}^n 7)$ на рис. 6.3 и одна технологическая компонента может управлять работой нескольких автономных моделей</p>	<ol style="list-style-type: none"> 1. Фрейм декомпозиции неоднородной задачи; 2. Список моделей (ГМ - список); 3. Список входных и выходных параметров по каждой модели (П - список); 4. Компьютерные программы автономных функциональных моделей 	<ol style="list-style-type: none"> 1. Фрейм декомпозиции, пополненный информацией о технологическом модельном поле (ТМ- поле); 2. Компьютерные программы технологических компонентов 3. Список технологических компонентов (ТМ - список)

Таблица 7.14 — Описание операции выбора стратегии из таблицы гибридных стратегий

<p><i>Операция № 14</i> <i>Название:</i> «Выбор стратегии из таблицы гибридных стратегий» <i>Этап проблемно-структурной методологии:</i> «Выбор стратегии из таблицы гибридных стратегий» <i>Система управления деятельностью разработчика:</i> операция выполняется после завершения операции 13. Операция выполняется до начала выполнения операции 15</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Эксперт выбирает их множества интегрированных моделей (6.15), включенных в таблицу гибридных стратегий конкретную модель, отображающую одну из декомпозиций сложной задачи $\hat{\pi}^u \in \hat{\Pi}^u$ (5.10). Содержание данной операции позволяет преодолеть свойство изменчивости в составе и структуре сложной задачи относительно времени и обеспечить выбор той или иной интегрированной модели для последующего синтеза</p>	<ol style="list-style-type: none"> 1. Таблица гибридных стратегий; 2. Фрейм декомпозиции неоднородной задачи 	<ol style="list-style-type: none"> 1. Одна их интегрированных моделей, выбранная из таблицы гибридных стратегий. 2. Отредактированный фрейм декомпозиции неоднородной задачи

функциональной ГиИС		
---------------------	--	--

Таблица 7.15 — Описание операции выбора альтернативной автономной модели по области ее релевантности

<p><i>Операция № 15</i> <i>Название:</i> «Выбор альтернативной автономной модели по области ее релевантности» <i>Этап проблемно-структурной методологии:</i> «Инициализация функциональной ГиИС по выбранной стратегии (синтез метода решения неоднородной задачи)» <i>Система управления деятельностью разработчика:</i> операция выполняется, если выполнялась операция 10 — «Разработка области релевантности модели». Операция выполняется после завершения операции 14. Операция выполняется до начала выполнения операции 16</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>На основании информации, получаемой от пользователя о режиме принятия решений, осуществляется (или не осуществляется) выбор конкретной автономной модели по области ее релевантности. Содержание операции отражает плюрализм мнений экспертов при гибридизации с целью разработки функциональных ГиИС</p>	<ol style="list-style-type: none"> 1. Отредактированный фрейм декомпозиции неоднородной задачи; 2. Настроенная по результатам выполнения операции 10 нечеткая система; 3. Список пар «модель — интервалы» (МИ — список); 4. Режим принятия решений: МАКСИМУМ (выбор осуществляется по результатам работы нечеткой системы); ПОЛЬЗОВАТЕЛЬ — (выбор осуществляется пользователем); ОТКЛЮЧИТЬ (принятия решений о выборе автономной модели по области ее релевантности нет) 	<p>Фрейм-представление неоднородной задачи, в котором идентифицированы конкретные автономные модели из состава интегрированной модели, удовлетворяющие условиям решения сложной задачи</p>

Таблица 7.16 — Описание операции инициализации функциональной гибридной интеллектуальной системы

<p><i>Операция № 16.</i> <i>Название:</i> «Инициализация функциональной гибридной интеллектуальной системы» <i>Этап проблемно-структурной методологии:</i> «Инициализация функциональной ГиИС по выбранной стратегии (синтез метода решения неоднородной задачи)» <i>Система управления деятельностью разработчика:</i> Операция выполняется после завершения операции 15. Операция выполняется до начала выполнения операции 17.</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Выполняется автоматически, без участия пользователя с целью завершения формирования в памяти компьютера символического представления функциональной ГиИС. Символьное представление функциональной ГиИС получается из фрейма представления неоднородной задачи идентификацией ссылок на ком-</p>	<ol style="list-style-type: none"> 1. Фрейм-представление неоднородной задачи; 2. Список моделей (ГМ - список); 3. Список технологических компонент (ТМ - список); 4. Список интерфейсов автономных моделей (ИМ 	<p>Символьное представление функциональной ГиИС (объекта-результата)</p>

пьютерные программы автономных моделей, технологических компонент, интерфейсов автономных моделей	- список)	
---	-----------	--

Таблица 7.17 — Описание операции моделирования решения сложной задачи и анализа результатов

<p><i>Операция № 17</i> <i>Название:</i> «Моделирование решения сложной задачи и анализа результатов» <i>Этап ИС-методологии:</i> «Машинные эксперименты и интерпретация результатов» <i>Система управления деятельностью разработчика:</i> 1) операция выполняется после завершения операции 16. Операция выполняется до начала выполнения операции 19 или 21; 2) операция может выполняться одновременно с операцией 18 - «Координация функциональных компонентов»</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Выполняется автоматически под контролем пользователя для получения результатов решения сложной задачи моделированием ФГИИС по ее символическому представлению, полученному после операции 16. Результаты интерпретируются. Пользователь может формировать или нет множество альтернативных решений (операция 19)</p>	<p>1. Символьное представление функциональной ГИИС (объекта-результата); 2. Компьютерные программы автономных моделей, технологических компонентов и интерфейсов автономных моделей; 3. БД и БЗ автономных моделей; 4. Режим работы: КООРДИНАЦИЯ (выполняется операция 18 «Координация функциональных компонентов»); ОТКЛЮЧИТЬ (операция 18 не выполняется); 5. Режим формирования результатов: ФОРМИРОВАТЬ (выполняется операция 19 — «Формирование множества альтернативных решений сложной задачи»); ОТКЛЮЧИТЬ (операция 19 не выполняется)</p>	<p>1. Результаты решения сложной задачи (цели неоднородной задачи G^u (5.10)) для выбранной из таблицы гибридных стратегий (Таблицы) интегрированной модели и принятого решения по выбору автономных моделей с учетом их областей релевантности</p>

Таблица 7.18 — Описание операции координации функциональных компонентов

<p><i>Операция № 18</i> <i>Название:</i> «Координация функциональных компонентов» <i>Этап ИС-методологии:</i> «Моделирование решения сложной задачи» <i>Система управления деятельностью разработчика:</i> 1) операция выполняется одновременно с операцией 17 — «Моделирование решения сложной задачи и анализ результатов»; 2) самостоятельно операция не выполняется</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Имитация взаимодействия ЛПР в сложной задаче и консультирующих его экспертов, решающих ее подзадачи. При этом процесс функционирования ФГИИС разбивается на этапы. Между этапами выполняются анализ решения автономных задач (имитация работы экспертов) и перераспределение ресурсов, выделенных ЛПР. Выполняется с целью компенсации несоответствия, возникающего при</p>	<p>1. Фрейм декомпозиции неоднородной задачи; 2. Результаты решения одной или нескольких однородных задач в соответствии с декомпозицией неоднородной задачи; 2. Новые значения исходных данных и условий в моделях однородных задач, устанавливаемые ЛПР или техноло-</p>	<p>1. Измененные знаки однородных задач в информационном языке L_6^h (5.9); 2. Измененные значения входных параметров автономных моделей</p>

редукции сложной задачи (6.20) ÷ (6.22) во множество взаимосвязанных подзадач (декомпозицию) $\hat{\pi}^u \in \hat{\Pi}^u$ (5.10).	гическим компонентом «Лицо, принимающее решения»	
--	--	--

Таблица 7.19 — Описание операции формирования множества альтернативных решений сложной задачи

<p><i>Операция № 19</i> <i>Название:</i> «Формирование множества альтернативных решений сложной задачи» <i>Этап проблемно-структурной методологии:</i> «Формирование множества альтернатив» <i>Система управления деятельностью разработчика:</i> Операция выполняется после завершения операции 17. Операция выполняется до начала выполнения операции 20</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Формируется список результатов решения сложной задачи (PP — список). Данную операцию целесообразно выполнять, если пользователь предполагает получить в ходе моделирования более 7—10 альтернативных решений, выбрать из которых одно решение ему затруднительно. При формировании списка он упорядочивается по множеству декомпозиций сложной задачи $\hat{\Pi}^u$ (5.10) и по используемым автономным моделям в зависимости от их областей релевантности (6.19)</p>	<ol style="list-style-type: none"> 1. Результаты моделирования решения сложной задачи, получаемые в операции 18; 2. Фрейм декомпозиции неоднородной задачи; 3. Список пар «модель — интервалы» (МИ- список) 	<p>Результаты решения сложной задачи упорядоченные: 1) по декомпозициям $\hat{\Pi}^u$ (5.10) «Декомпозиция — результаты»; 2) по областям релевантности автономных моделей «Модель — результат» в PP - список</p>

Таблица 7.20 — Описание операции совершенствования множества альтернативных решений сложной задачи и анализа результатов

<p><i>Операция № 20</i> <i>Название:</i> «Совершенствование множества альтернативных решений сложной задачи и анализа результатов» <i>Этап ИС-методологии:</i> «Совершенствование множества альтернатив и анализ результатов» <i>Система управления деятельностью разработчика:</i> Операция выполняется после завершения операции 19. Операция выполняется до начала выполнения операции 21</p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
<p>Решается задача совершенствования множества альтернатив с помощью генетического алгоритма. При этом множество альтернатив рассматривается как популяция индивидуумов, эволюция которой в соответствии с функцией приспособленности и соответствующими значениями параметров настройки имитируется генетическим алгоритмом. В результате получается информация в виде усовершенствованного (сокращенного до одной или нескольких) множества альтернатив. ЛППР изучает альтернативы, принимает</p>	<ol style="list-style-type: none"> 1. Результаты решения сложной задачи (PP - список); 2. Функция (приспособленности) оценки; 3. Параметры настройки генетического алгоритма; 4. Операция в объекте управления, реализующая результаты решения сложной задачи; 5. Оценки результатов 	<ol style="list-style-type: none"> 1. Принятая альтернатива решения сложной задачи; 2. Информация об организованной для выполнения альтернативы операции, занесенная в знак неоднородной задачи; 3. Оценки экспертов, решавших однородные задачи, занесенные в знак неоднородной задачи;

решение, организует операцию по выполнению решения и дает оценки экспертам по итогам решения ими однородных задач	решения однородных задач экспертами	родной задачи
---	-------------------------------------	---------------

Таблица 7.21 — Описание операции эксплуатации и модернизации

<p><i>Операция № 21</i> <i>Название: «Эксплуатация и модернизация»</i> <i>Этап проблемно-структурной методологии: «Эксплуатация и модернизация»</i> <i>Система управления деятельностью разработчика: Операция выполняется после завершения операции 20 или 17. Операция — конечная в проблемно-структурной технологии</i></p>		
Содержание операции	Входная информация Базы данных Базы знаний	Выходная информация
Внесение изменений в базы данных и базы знаний функциональных элементов (автономных моделей). Совершенствование автономных моделей, технологических компонентов и интерфейсов автономных моделей. Внесение изменений в таблицу гибридных стратегий, настройка областей релевантности автономных моделей. Изменение списка экспертов, привлечение для решения подзадач сложной задачи	<ol style="list-style-type: none"> 1. Базы данных и знаний функциональных и технологических элементов; 2. Компьютерные программы автономных моделей, технологических компонентов и интерфейсов моделей; 3. Таблица гибридных стратегий (Т-таблица); 4. Списки моделей (ГМ - список), технологических компонентов (ТМ - список), интерфейсов моделей (ИМ - список); 5. Список существенных параметров для данной модели (ПС - список); 6. Список пар «модель — интервал» (МИ - список); 7. База знаний (KB^{mx} (6.19)) нечеткой системы, используемой в операции 15 	Отредактированные базы данных и знаний, компьютерные программы, списки моделей

7.4. Последовательность действий разработчика и пользователя функциональных гибридных интеллектуальных систем

Второй существенный компонент ПС-технологии (7.1) связан с установлением отношений следования на множестве операций:

$$\Pi^O = ({}^3X_o, {}^{33}R^n | 6), \quad (7.4)$$

где ${}^{33}R^n | 6$ — отношения следования (входят в класс отношений «действие-действие» (рис. 5.2)).

Установление отношений следования позволяет увязать операции информатики в единую последовательность действий, осмысленную и апробированную с точки зрения достижения цели — разработки функциональной ГиИС для решения в автоматизированном режиме сложной,

неоднородной задачи (рис. 7.4). Логическая последовательность действий — $\Pi^O(7.4)$ ПС-технологии изображена на рис. 7.6.

Она представлена ориентированным графом, вершины которого обозначают операции, справа от которых даны их названия. Дуги графа обозначают отношение «следовать за». Детально эти отношения рассмотрены в разделе «Система управления деятельностью разработчика» табл. 7.1 — 7.21.

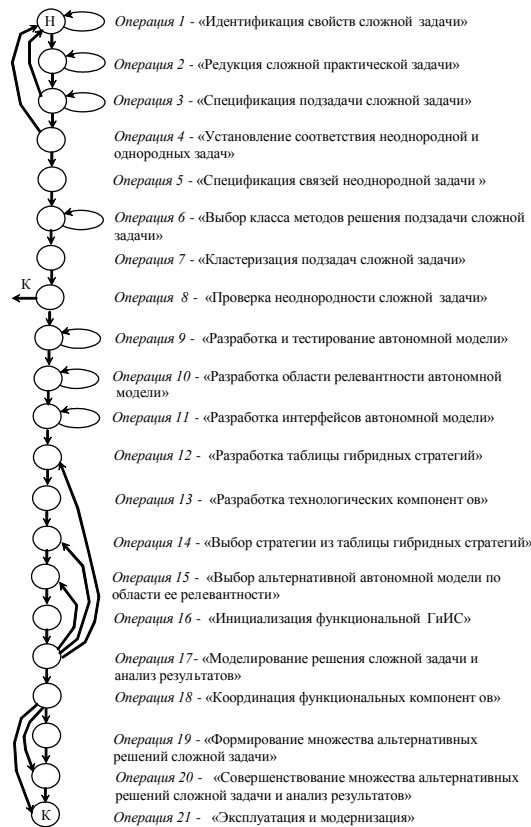


Рис. 7.6 — Логическая последовательность операций разработки функциональных гибридных интеллектуальных систем

Петли при вершинах графа обозначают возможность многократного повторения соответствующей операции. На графе не показаны возмож-

ные «откаты назад» из-за допущенных в ходе разработки ошибок. Буквой «Н» показано начало работ, буквой «К» обозначены альтернативы окончания работ. Таких альтернатив две: первая — операция 21; вторая — операция 8, если проверка показала, что исходная задача нерелевантна модели «неоднородная задача» (5.10).

В логической последовательности, изображенной на рис. 7.6, можно выделить три части. Часть 1 — «Системный анализ сложной задачи» охватывает операции 1 — 8 и выполняется с целью глубокого проникновения участников СППР и разработчиков в существо решаемой задачи и извлечения из экспертов информации (данных и знаний) для моделирования. Работы по системному анализу заканчиваются проверкой и установлением факта наличия у исходной задачи свойства неоднородности. Если такового отсутствует, нет смысла переходить к работам из второй части. Часть 2 — «Синтез функциональной ГиИС и моделирование решения сложной задачи» охватывает операции 9 — 20. Работы на этом этапе носят итерационный характер, что отражает ситуацию, когда интегрированная модель постепенно уточняется и адаптируется коллективом разработчиков к изменчивым составу и структуре сложной задачи и условиям ее решения. Часть 3 «Эксплуатация и модернизация» охватывает одну операцию — 21. Однако это не означает, что нельзя детализировать работы и этого этапа.

7.5. Инструментальная среда технологии

Для апробации операций ПС-технологии функциональных ГиИС на сложных задачах из двух неоднородных проблемных сред [1]: «Поддержка принятия решений в био-производственной системе» и «Поддержка принятия конструкторских решений по системам судовой автоматизации машинных отделений транспортных судов иностранной постройки» были разработаны программные средства (табл. 7.22) из множества ${}^9X_3^0$ (7.1).

Ниже дано краткое описание выполнения операций из левого столбца табл. 7.22 некоторыми программными средствами из правого столбца табл. 7.22. При этом основное внимание уделено средствам поддержки системного анализа сложных задач.

Средства редукции и идентификации сложных задач. Для редукции и идентификации сложных задач и их подзадач, т.е. выполнения операций 1 — 4, разработан комплекс программ САПФГИС, имеющий модульную структуру: главный модуль для управления другими модулями; модуль работы с графическими элементами (позволяет добавлять, удалять, перемещать подзадачи, а также добавлять и удалять от-

ношения между подзадачами); модуль работы с базой данных (позволяет выполнять функции, связанные с редактированием информации в базе данных); модуль редактирования сведений о задаче (позволяет рассматривать новые или работать с уже созданными задачами); модуль редактирования сведений о декомпозиции (позволяет создавать новые декомпозиции или работать с созданными); модуль редактирования сведений о подзадаче и отношении (позволяет редактировать сведения о подзадаче или отношении); модуль формирования документов (позволяет формировать документы-отчеты).

Программное обеспечение апробировано на примере сложной задачи планирования урожаев и агротехнологических мероприятий (разд. 7.6).

Средства установления соответствия объектов-оригиналов и объектов-прототипов. Для выбора класса методов решения подзадач сложной задачи, т.е. выполнения операции 6, разработано, как видно из табл. 7.22, несколько программ.

Таблица 7.22 — Программная апробация операций ПС-технологии

Номер, название операции	Приложение
1 — «Идентификация свойств сложной задачи»	1) Программа U-СПЕЦИФИКАТОР; 2) Система автоматизированного проектирования ФГИС — САПФГИС
2 — «Редукция сложной задачи»	1) Программа U-РЕДЬЮСЕР; 2) Комплекс программ САПФГИС
3 — «Спецификация подзадачи сложной практической задачи»	1) Программа H-СПЕЦИФИКАТОР; 2) Комплекс программ САПФГИС
4 — «Установление соответствия неоднородной и однородных задач»	Пакет программ АГРО
5 — «Спецификация связей неоднородной задачи»	
6 — «Выбор класса методов решения подзадачи сложной задачи»	1) Программы НМ-КОНСУЛЬТАНТ-1, 2) Автоматизированная система выбора класса методов обработки информации — АС ВКМОИ
7 — «Кластеризация подзадач сложной задачи»	Комплекс программ анализа сложной задачи ПА СПЗ
8 — «Проверка неоднородности сложной задачи»	Комплекс программ анализа сложной задачи ПА СПЗ
9 — «Разработка и тестирование автономной модели»	
10 — «Разработка области релевантности автономной модели»	Пакет программ АГРО
11 — «Разработка интерфейсов автономной модели»	Программа ИНТЕРФЕЙСЫ; пакет программ АГРО
12 — «Разработка таблицы гибридных стратегий»	Пакет программ АГРО
13 — «Разработка технологических компонентов»	Пакет программ АГРО
14 — «Выбор стратегии из таблицы гибридных стратегий»	ГИБРИДИЗАТОР
15 — «Выбор альтернативной автономной модели по области ее релевантности»	Пакет программ АГРО
16 — «Инициализация функциональной ГИИС»	Программа ГИБРИДИЗАТОР
17 — «Моделирование решения сложной задачи и анализ результатов»	Пакет программ АГРО

18 — «Координация функциональных компонентов»	«Гибридная система планирования»
19 — «Формирование множества альтернативных решений сложной задачи»	
20 — «Совершенствование множества альтернативных решений сложной задачи и анализ результатов»	
21 — «Эксплуатация и модернизация»	Пакет программ АГРО

Ниже рассмотрены результаты экспериментов с программой НМ-КОНСУЛЬТАНТ-1,2, в основе которой лежит алгоритм 6.1 из разд. 6. Эта программа написана на Delphi 5.0, использует обмен данными через DDE с программой на Си-подобном языке и экспертной системой, разработанной в оболочке KAPPA-PC. Экспертная система работает с результатами других программ [1].

Основная особенность программы — двухуровневая схема обработки результатов. На первом, обобщенном уровне результаты выдаются для каждой из анализируемых подзадач из декомпозиции Π^h по всем шести классам базисных методов в баллах. Чем больше количество баллов, «набранных» классом методов, в этом случае оценка ближе к единице, тем выше может быть у пользователя степень уверенности в том, что задаче релевантны методы из этого класса.

Однако достаточно часто балльные оценки бывают близкими друг к другу, что затрудняет принятие решений. В этом случае программа предлагает консультации на другом, уже детализированном уровне. Для этого, по запросу пользователя, для каждой исследуемой подзадачи выдаются не только общие оценки, но и баллы, полученные классом методов по каждой из шести групп свойств спецификатора ${}^2O^h$ (5.7).

Очевидно, что при этом у пользователя появляются более широкие возможности для сравнительного анализа с учетом того, чем в спецификации задачи он считает возможным пренебречь, а что для данного проекта — обязательные ключевые моменты.

С программой ГИМ-НМ-КОНСУЛЬТАНТ-1,2 были проведены эксперименты, суть которых состояла в следующем. В языке описания однородных задач L_6^h специфицировались заведомо корректные, с точки зрения базы знаний KB^m , знаки-эталоны, т.е. такие знаки ${}^8x^n|_j^h$, в отношении которых заранее известно, в какой класс задач ${}^8X^n|_{ha}$, решаемых подмножеством инструментариев ${}^qM^a|_q = \{АН, ИСМ, ЭС, НС, ИНС, ГА\}$, они будут отнесены.

Затем в параметры знаков-эталонов вносился шум, что имитировало ошибки разработчика функциональных ГиИС в ходе установления соответствия «задача-методы». Экспериментаторов интересовали качество консультаций и «чувствительность» системы с базой знаний и вероят-

ностной нейронной сети к ошибкам пользователя. На оба поставленных вопроса помогают ответить графики, изображенные на рис. 7.7.

По осям OX на всех графиках отложено количество зашумленных позиций в знаках-эталонах, а по оси OY — бальность в долях единицы

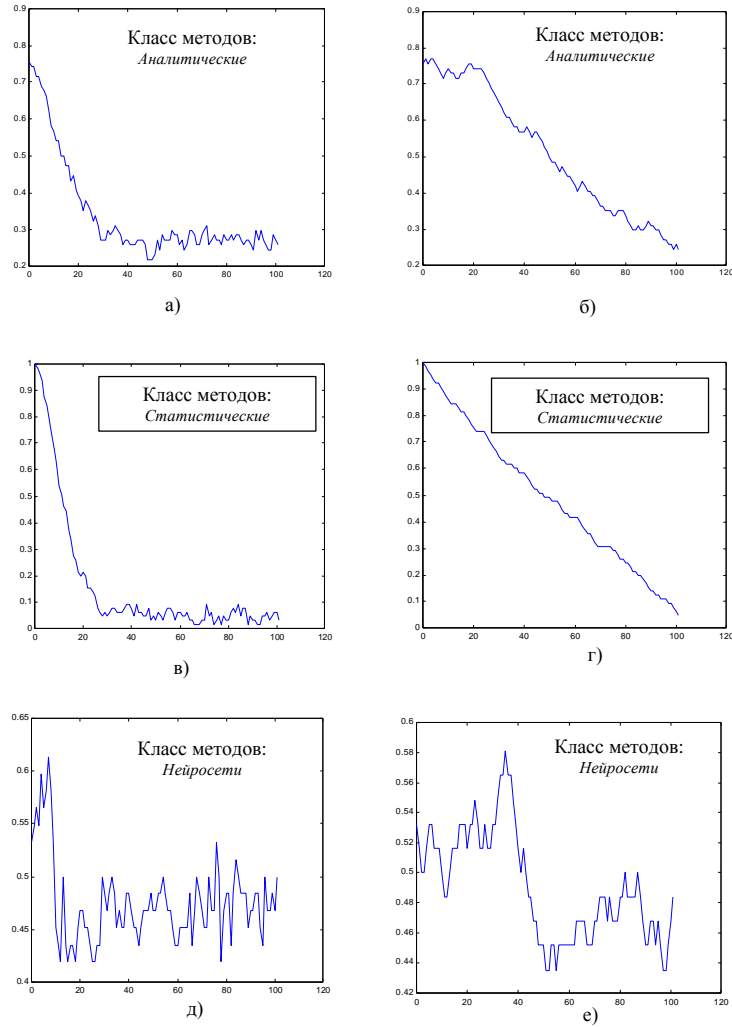
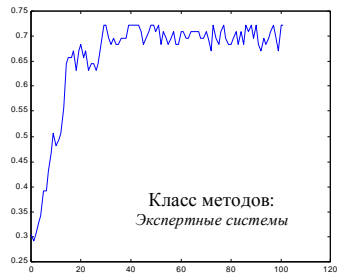


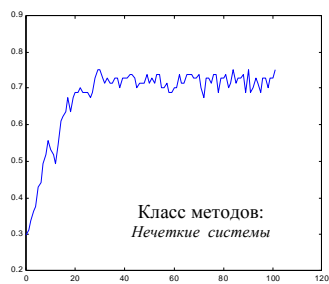
Рис. 7.7 — Результаты экспериментальных исследований системы с базой знаний для выбора класса методов



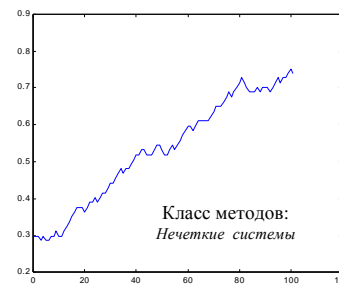
ж)



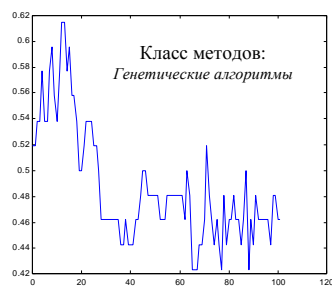
з)



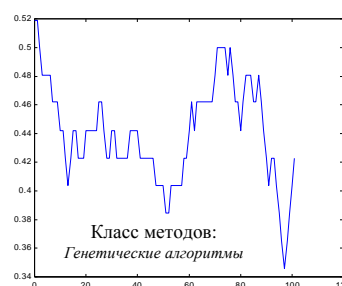
и)



к)



л)



м)

Рис. 7.7 — Результаты экспериментальных исследований системы с базой знаний для выбора класса методов (продолжение)

отнесения специфицированной задачи к тому классу методов, наименование которого выделено на поле графика курсивом. На рис. 7.7, *a, в, д, ж, и, л* показаны результаты одной из шести серий экспериментов, суть которых сводится к следующему. В качестве шаблона-эталона для исследования был взят эталон, которому заведомо соответствует класс имитационных статистических моделей. На графике рис. 7.7, *в* это показано заключением в прямоугольную рамку наименования класса базисных методов. Затем в шаблон-эталон вносился шум, причем количество зашумленных позиций нарастало. При этом вновь рассчитываемое количество зашумленных позиций увеличивалось на единицу по сравнению с предыдущим, а ранее внесенный шум сохранялся. На графике рис. 7.7, *в* видно, что с увеличением количества шума в эталоне значения оценок резко падают, а после примерно 30 ошибок значение оценки стабилизируется и становится равным 0,1, что говорит о высокой чувствительности системы с базой знаний.

Графические зависимости на рис. 7.7, *a, д, ж, и, л* показывают, как влияет количество увеличивающегося шума в эталоне «статистические методы» на балльные оценки отнесения знака-шаблона к остальным пяти классам базисных методов. Анализ пары графиков на рис. 7.7, *в* и рис. 7.7, *a* показывает, что система с базой знаний относит эти два класса методов в один кластер. Анализ графиков рис. 7.7, *в*, а также рис. 7.7, *ж, и* показывает, что в базе знаний существует еще один кластер (экспертные и нечеткие системы), который достаточно «удален» от кластера аналитико-статистических методов.

Средства кластеризации подзадач и проверки неоднородности сложной задачи. Для кластеризации подзадач и проверки неоднородности исходной задачи, т.е. выполнения операций 7, 8 из табл. 7.22, создан комплекс программ ПА СПЗ.

Разработанное программное обеспечение использует в качестве исходной информации результаты идентификации и построения декомпозиций сложной задачи и позволяет эксперту, проектировщику специфицировать множество подзадач, а затем на основании спецификации выделить кластеры похожих подзадач. Можно рассчитать оценки, характеризующие применимость каждого из классов автономных методов для решения каждого кластера подзадач, и при затруднении эксперта или проектировщика определить, можно ли данную задачу отнести к неоднородной.

Программное обеспечение имеет модульную архитектуру. Первый модуль работает с базой данных (занесение, удаление, редактирование вопросов, классов методов, параметров вопросов, возможных ответов на вопросы, ответов на вопросы для каждой подзадачи). Второй модуль

выбирает класс автономного метода (формирование векторов параметров конечных подзадач, разбиение векторов параметров подзадач на кластеры по алгоритму ISODATA, определение векторов — представителей кластеров, задание обучающего примера, построение и обучение вероятностной нейронной сети, расчет класса методов обработки информации, вывод пользователю оценок релевантности каждого класса для каждого кластера подзадач). Третий модуль проверяет задачи на неоднородность (выявление «плохо» определенных задач, т.е. задач, для которых на большинство вопросов пользователь не дал никакого ответа; проверка возможности использования каждого класса автономного метода при решении всех конечных подзадач; вывод пользователю информации, может ли быть задача решена в рамках одного класса методов). Четвертый модуль формирует отчеты.

Программное обеспечение было апробировано на примере системного анализа сложной задачи планирования урожаев и агротехнических мероприятий в био-производственной системе (разд. 7.7).

Первоначально была построена декомпозиция задачи. Она соответствует декомпозиции, изображенной на рис. 7.9,б. Декомпозиция содержит девять однородных подзадач из табл. 7.24. После построения декомпозиции каждая однородная задача была специфицирована. Далее выполнялась кластеризация (операция 7) и операции 6, 8. В результате анализа однородные подзадачи были объединены в четыре кластера и рассчитаны оценки релевантности классов методов (табл. 7.23).

Таблица 7.23 — Результаты анализа задачи «Планирование урожаев и агромероприятий» комплексом программ ПА СПЗ

Номер кластера	Код подзадач	Численные оценки релевантности
1	ПЭК, АГР3, ИНЖ, ГР2Р(П1), АГР4	Аналитические методы — 95% Имитационное статистическое моделирование — 23,8% Искусственные нейронные сети — 35,2% Экспертные системы — 35,1% Нечеткие системы — 83,3% Генетические алгоритмы — 3,83%
2	АГР7, АГР2В	Аналитические методы — 0,05% Имитационное статистическое моделирование — 0,26% Искусственные нейронные сети — 0,26% Экспертные системы — 0,03% Нечеткие системы — 0,02% Генетические алгоритмы — 0,57%
3	АГР1(П1),	Аналитические методы — 5,82% Имитационное стат. моделирование — 9,58% Искусственные нейронные сети — 29,1% Экспертные системы — 28,9% Нечеткие системы — 74,3% Генетические алгоритмы — 1,67%
		Аналитические методы — 11,2% Имитационное статистическое моделирование — 11,4%

4	АГР2С(П)	Искусственные нейронные сети — 56,3% Экспертные системы — 86,7% Нечеткие системы — 62,4% Генетические алгоритмы — 5,2%
---	----------	---

Подзадачи из кластера 2 плохо специфицированы, оценки методов низкие и не превышают 2%. Подзадачи кластера 1 могут быть решены аналитическими методами, а кластера 3 — методами нечетких систем, кластера 4 — методами экспертных систем или нечетких систем.

7.6. Пример решения сложной задачи планирования урожаев и агротехнических мероприятий в био-производственной системе

После завершения полевых работ в агрофирмах планируются урожаи и агротехнические мероприятия на весенне-летний период будущего года. Эта задача решается в процессе коллективной творческой деятельности с привлечением широкого спектра аналитических, статистических, логико-лингвистических и других знаний, распределенных по профессиональным нишам: экономики, бухгалтерского учета, агрономии, мелиорации на разных уровнях от конкретного сельскохозяйственного поля агрофирмы до района и области.

Проведенные исследования [258] показали актуальность планирования урожаев и агротехнических мероприятий, в первую очередь отдельного сельскохозяйственного поля с его уникальной агроэкосистемой. Особые проблемы при решении данной задачи возникают в зонах рискованного растениеводства, к которым относится и Калининградская область. Это связано с необходимостью резкого снижения ошибки прогнозирования, достигающей порой 40% от объема урожая. Последствия подобных просчетов в планировании могут колебаться в весьма широких пределах от частичного сглаживания неспрогнозированными заранее благоприятными погодными условиями в период возделывания и уборки урожая до полного банкротства агрофирмы, например, из-за неспособности вернуть банковские кредиты.

Повышение качества планирования урожая позволяет снизить затраты на аготехмероприятия, себестоимость продукции и увеличить объем прибыли сельскохозяйственного предприятия.

Особенности био-производственной системы. В био-производственных системах сельского хозяйства сочетаются социально-экономические и естественные процессы воспроизводства. В качестве основных средств производства и одновременно объектов труда выступают почва, обладающая определенным плодородием, и растения. Зеленые растения используют плодородие почвы, воздушную среду, осадки, ветер, солнце, чтобы превращать неорганические вещества в органические.

ские, т.е. создавать белки, крахмал, сахар и другие вещества, входящие в состав растениеводческих продуктов.

Измеряя состояние почвы, растений, воздуха, солнечной активности и других факторов, прогнозируя метеоусловия и развитие растений, а также имея в своем распоряжении некоторые ресурсы, субъект управления в био-производственной системе разрабатывает агротехнические мероприятия для достижения определенных целей: объема урожая, рентабельности и прибыльности растениеводства.

Био-производство обладает сезонностью, здесь труднее, чем в других отраслях применять технику и технологию возделывания и измерений параметров агроэкосистемы, совершенствовать организацию труда.

Кроме рассмотренных особенностей следует обратить внимание на то, что объект управления (людские ресурсы не рассматриваются) представляет сложную, многоуровневую, самоорганизующуюся систему управления на уровнях: клетки, организма, оперативного и стратегического управления организмом [134].

Модели, применяемые на фазе прогнозирования в подобных системах, чрезвычайно сложны и состоят, по существу, в предвосхищении будущего поведения организма в результате его взаимодействия с внешней средой. Эти модели должны включать как родовой опыт организма, так и специфику его поведения в конкретных ситуациях, что содержит элементы случайности, неопределенности и неполноты. Применение теории управления в биологии и медицине всегда было затруднено из-за особенностей живых систем [11].

В связи с рассмотренными особенностями, моделирование био-производственных систем должно происходить на основе компромисса между дедуктивными математическими моделями, создаваемыми биологами для описания модели родового поведения организма в условиях внешней среды, с индуктивными, эвристическими моделями поведения организма в конкретных ситуациях. Если для математических моделей характерен учет зональных и региональных закономерностей жизни организма, то эвристические модели — более детализированы и должны учитывать особенности конкретного хозяйства, занимающегося растениеводством и, что особенно важно, — конкретного поля и участка поля как основной среды жизни организма.

Один из возможных путей построения компромиссных моделей — это коррекция фундаментальных аналитических зависимостей эвристическими знаниями в ГиИС.

Системный анализ задачи планирования урожаев и агротехнических мероприятий. Био-производство — сложная, многоаспектная и стратифицированная система (рис. 7.8). В ней трудятся специалисты разных профессий, имеющих свое уникальное представление и понима-

ние процессов сельскохозяйственного производства. Однако производственно-экономическое, инженерно-технологическое и биологическое представления наиболее существенны для коллективного решения задачи планирования урожаев и агротехнических мероприятий. Анализ фаз учета, контроля, анализа, планирования, прогнозирования и регулирования на стратифицированной модели БПС позволил выделить функции управления, автоматизация которых дает наибольший экономический эффект.

На агрострате, как наиболее тонком и детальном представлении БПС, специалистами из службы главного агронома решаются сложные задачи планирования и прогнозирования. В результате решения задач планирования формируется множество планов агротехмероприятий возделывания сельскохозяйственных культур. Планы определяют возможные варианты влияния на биопроцессы. При разработке планов используются информация об исходном состоянии агроэкосистемы, производственных ресурсах агрофирмы, прогнозах воздействий внешней среды и профессиональные знания субъекта управления. Планы должны удовлетворять некоторым целевым установкам.

Прогнозирование выполняется для предсказания состояния агроэкосистемы при реализации плана.

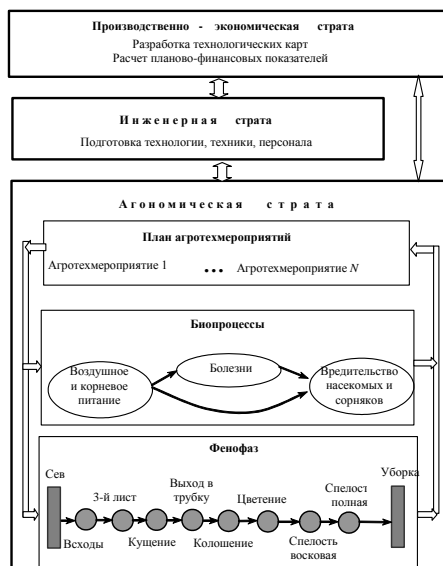


Рис. 7.8 — Концептуальная модель био-производственной системы

На инженерной страте специалисты инженерной службы занимаются подготовкой персонала, техники и технологии для выполнения планируемых в будущем году агротехмероприятий.

На производственно-экономической страте, более абстрактном представлении БПС, присущем профессиям экономиста и бухгалтера, в соответствии с планом агромероприятий составляются технологические карты, определяющие объем и последовательность выполнения операций по возделыванию культур, и рассчитываются планово-финансовые показатели производства сельскохозяйственной продукции.

Между стратами существует обмен информацией в ходе планирования производства. Так план агромероприятий, прогнозы урожая передаются на вышерасположенную страту и становятся основой выполняемых там расчетов. В свою очередь, результаты этих расчетов выступают в качестве оценок качества агромероприятий и необходимости корректировки прогноза урожая.

Сложная задача «планирование урожаев и агротехнических мероприятий», в соответствии с хорошо известным и уже отработанным в сельском хозяйстве подходе [259], состоит из девяти задач (коды ПЭК, ИНЖ и АГР1 — АГР7, см. табл. 7.24), расчеты по которым должны быть выполнены на различных стратах в определенной последовательности, чтобы исходная задача считалась решенной. Результаты анализа подзадач, возникающих при планировании урожаев и агротехнических мероприятий, а также моделей и методов их решения экспертами, а также инструментариями, известными в России и за рубежом, сведены в табл. 7.24.

Задачи планирования и прогнозирования на производственно-экономической страте успешно решаются системой AgroTool (Всесоюзный НИИ и проектно-технологический институт по разработке, внедрению и эксплуатации АСУ в агропромышленном производстве ВАСХНИЛ, Всесоюзный НИИ экономики сельского хозяйства ВАСХНИЛ и др.), экспертными системами AFAES, MES, ISARM и системой принятия решений «Озимая пшеница» [1] — в рамках одного представления знаний.

Решение задач планирования и прогнозирования главным агрономом слабо представлено в рассматриваемых программах. Это объясняется многими причинами. Прежде всего, неполнотой знаний о биопроцессах. Однако эти рамки оказываются слишком узкими для разнообразия профессиональных знаний на этой страте. Так, например, для решения задачи АГР1 нужны логико-лингвистические знания о формировании индекса листовой поверхности. Чтобы получать планируемые урожаи, для задачи АГР2 актуальны аналитические, статистические, логико-лингвистические и эволюционные знания о связях развития и адаптации фотосинтетического аппарата сельскохозяйственных растений в разных условиях внешней среды со скоростью роста посева в онтогенезе. Для АГР3 важны аналитические и логико-лингвистические знания об интенсивности поступления элементов питания и накоплении сухого вещества сельскохозяйственной культурой за период вегетации и т.д.

Задачи АГР1 — АГР7, ИНЖ успешно решаются пока только экспертами. Как показал анализ, создатели пакета СИМОНА (Система имитационного моделирования динамики агроэкосистем Агрофизиче-

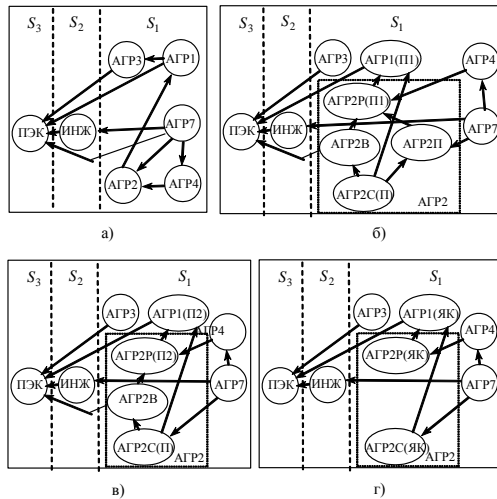
ского института ВАСХНИЛ), пакета AgroTool, экспертных систем и других подобных программ не рассматривают эти задачи в силу специфичности и привязанности к производственным и социальным условиям конкретной агрофирмы, агроэкосистемам поля и отдельных его участков. В то же время крупные производители сельскохозяйственной продукции настоятельно обращают внимание на актуальность автоматизированного решения именно этих задач.

Редукция сложной задачи планирования урожаев и агротехнических мероприятий. Исследования состава сложной задачи выполнялись параллельно с построением и изучением ее декомпозиций, в результате чего выбор был остановлен на структуре, изображенной на рис. 7.9,а. На нем видно, что для исследований из множества задач, приведенного в табл. 7.24, выбрано семь проблем на всех трех стратах. Отношения между подзадачами на рис. 7.9,а — это отношения обмена информацией, показывающие, в каком порядке справа налево решаются подзадачи (связи от страты S_q к страте S_j , где $q > j$ не показаны).

Таблица 7.24 — Методы и модели для решения задачи планирования урожаев и агротехмероприятий в био-производственной системе

Страта	Эксперт	Подзадача (код)	Методы и модели
Производственно-экономическая	Экономист	<i>Разработка агротехнологий, прогнозирование финансов, анализ плановых альтернатив (ПЭК)</i>	Функциональные зависимости Алгебраические уравнения
	Бухгалтер		
Инженерная	Инженер	<i>Разработка средств механизации производства (ИНЖ)</i>	Продукции
Агрономическая	Главный агроном	<i>Прогнозирование урожаев с/х культур (АГР1)</i>	Линейная регрессия Продукции
		<i>Прогнозирование продуктивности посевов (АГР2); Прогнозирование роста, развития посева (АГР2Р); Прогнозирование водного режима посева (АГР2В); Прогнозирование состояния пигментного комплекса (АГР2П); Разработка структуры урожая (АГР2С)</i>	Линейная регрессия Продукции Система дифференциальных уравнений
		<i>Прогнозирование доз удобрений и микроэлементов (АГР3)</i>	Функциональные зависимости Алгебраические уравнения Продукции
		<i>Прогнозирование повреждения с/х культур болезнями (АГР4)</i>	Продукции
		<i>Прогнозирование эффекта от мелиоративных работ (АГР5)</i>	Системы дифференциальных уравнений
		<i>Прогнозирование распространения болезней, вредителей и сорняков (АГР6)</i>	Системы дифференциальных уравнений Продукции
		<i>Разработка агротехнологий возделывания с/х культур (АГР7)</i>	Продукции

Обозначения: курсив — подзадачи, выбранные для исследований; с/х — сельскохозяйственных



Обозначения: S_1, S_2, S_3 — агрономическая, инженерная, производственная страты соответственно; (П), (ЯК) — пшеница и ячмень, картофель соответственно; (П1), (П2) — различные почвенно-климатические условия возделывания пшеницы

Рис. 7.9 — Декомпозиции сложной задачи планирования урожаев и агротехмероприятий

Спецификация проблем из декомпозиции на рис. 7.9,а показала, что все они, кроме АGR2, могут быть отнесены к однородным подзадачам. Задача АGR2, так же, как и исходная, должна рассматриваться как «задача-система». Ее структура показана внутри пунктирных прямоугольников на рис. 7.9,б — 7.9,г. Она зависит от возделываемой культуры: рис. 7.9,б, 7.9,в — для озимой пшеницы и рис. 7.9,г — для озимой ржи, ярового ячменя и картофеля. Наличие нескольких вариантов декомпозиции для

озимой пшеницы объясняется исследованиями различных лимитирующих факторов, в частности, почвенно-климатических условий.

Анализ свойств однородных задач, выбор автономных методов и разработка гетерогенного модельного поля. После построения и анализа декомпозиций задачи планирования урожаев и агротехмероприятий были выбраны методы решения ее подзадач, для чего исследовались их спецификации.

Для расчетов на производственно-экономической и инженерной страте достаточно манипулировать данными с помощью СУБД.

Для решения задач на агрономической страте применен широкий спектр методов. Для подзадачи АGR1 из-за отсутствия у экспертов соответствующих знаний выбраны нейро-нечеткие системы. Однако в агрофирмах и НИИ сельского хозяйства имелся богатый статистический материал, из которого такие знания могли быть извлечены автоматически.

Еще две статические модели прогнозирования урожаев озимой пшеницы и ржи в условиях засухи взяты из [260] и представляют собой регрессионные уравнения, разработанные аналитическими методами.

В сельскохозяйственной науке и практике накоплен значительный опыт решения задач АГР2Р, АГР2В, АГР3 и АГР4. Имеются проверенные практикой аналитические модели, которые были откорректированы с учетом специфических агроклиматических условий Калининградской области. По задаче АГР2П эксперты не имели необходимый объем знаний, но существовала богатая статистика, на которой была обучена искусственная нейронная сеть. По задаче АГР2С эксперты располагали опытом и знаниями для применения ЭС.

В итоге гетерогенное модельное поле [1] для планирования урожаев сельскохозяйственных культур для агрофирм Калининградской области насчитывает 19 моделей (табл. 7.25) и охватывает возделывание четырех культур — озимой пшеницы, озимой ржи, ярового ячменя и картофеля.

Таблица 7.25 — Гетерогенное модельное поле для синтеза ГИИС при планировании урожаев и агротехмероприятий

Подзадача (код)	Характеристики модели	
	Назначение (количество моделей)	Метод, представление
Разработка агротехнологий, прогнозирование финансов, анализ плановых альтернатив (ПЭК, ИНЖ)	Накопление, хранение, систематизации и выдача информации по технологии возделывания с/х культур (1)	Реляционная модель
Прогнозирование урожаев с/х культур (АГР1)	Статические модели расчета урожая (4)	Нейронечеткая система
	Статические модели расчета урожайности озимой пшеницы в условиях засухи (2)	Регрессионные уравнения
Прогнозирование роста, развития посева (АГР2В)	Динамические модели прогнозирования роста, развития с/х культур для различных агроклиматических, в том числе и стрессовых условий (4)	Аналитические функции
Прогнозирование водного режима посева (АГР2Р)	Динамическая модель прогнозирования водного режима посева с/х культур (1)	Аналитические функции
Прогнозирование состояния пигментного комплекса (АГР2П)	Статическая модель состояния пигментного комплекса озимой пшеницы (1)	Искусственная нейронная сеть
Разработка структуры урожая (АГР2С)	Динамические модели планирования элементов урожая с/х культур (4)	Производственная экспертная система
Прогнозирование доз удобрений и микроэлементов (АГР3)	Модель расчета доз удобрений и микроэлементов по характеристикам почв и прогнозируемой урожайности (1)	Система алгебраических уравнений
Прогнозирование повреждения с/х культур болезнями (АГР4)	Статическая модель прогнозирования повреждения с/х культур болезнями (1)	Функциональные зависимости

Обозначения: с/х — сельскохозяйственных

Синтез функциональных гибридных интеллектуальных систем над гетерогенным модельным полем и результаты экспериментов.

Для автоматизации трудоемких процедур синтеза функциональных ГиИС над гетерогенным модельным полем был разработан пакет программ АГРО [1] для ПЭВМ. Гибриды синтезировались в соответствии с алгоритмом 6.3 (разд. 6). В ходе моделирования было синтезировано и исследовано семь функциональных ГиИС.

Экспериментальные данные (1990 — 2000 гг.) о фактических урожаях и некоторых других существенных для планирования параметров были предоставлены агрономическими службами ООО им. Ладушкина, ООО «Борьба» и другими, Калининградским научно исследовательским институтом сельского хозяйства, Государственным центром агрохимической службы «Калининградский» и охватили полный спектр агроклиматических, агротехнических и микроклиматических условий возделывания сельскохозяйственных культур в Калининградской области.

Для определения возможностей гетерогенного модельного поля, синтезируемых ФГиИС и особенностей их поведения при различных агроклиматических и агротехнических условиях возделывания, использовался метод исследования динамических моделей экосистем [261].

Оценка качества моделей, входящих в состав гетерогенного модельного поля, для решения подзадач прогнозирования развития посева, прогнозирования водного режима посева и разработки структуры урожая, выполнялась по методике математического моделирования биогеохимических циклов в травяных экосистемах [261], для чего рассчитывались относительная средняя погрешность δ_{cp} ($0 \leq \delta_{cp} \leq 100, \%$) и коэффициент несовпадения Тейла U ($0 \leq U \leq 1$). По этой методике результаты моделирования считаются приемлемыми, если $\delta_{cp} \leq 25\%$, а значения U близки к нулю.

Относительная средняя погрешность (δ_{cp}) и коэффициент несовпадения Тейла (U) составили: для моделей структуры урожая — $\delta_{cp} = 10$ — 23 %, U не превышает 0,076 %; для моделей прогнозирования водного режима посева, моделей прогнозирования роста, развития посева — $\delta_{cp} = 10$ — 13 %, U не превышает 0,026 %, — что можно считать удовлетворительными результатами, сравнимыми с точностью расчета по моделям [259, 261 — 263].

Оценка погрешности и синхронности колебаний фактической и вычисленной с помощью ГиИС урожайности (рис. 7.10) выполнялась на основе оценки погрешности вычислений и синхронности колебаний

урожайности на модели прогнозирования урожаев культур [261]. Из рис. 7.10 видно, что эти колебания синхронны, максимальная погрешность проведенных экспериментов не превышала 15 % от фактических значений. «Уверенная» синхронность колебаний урожайности и низкая относительная средняя погрешность достигаются благодаря совместному использованию ГИИС и моделей оценки релевантности (6.19).

В среднем, по серии из 60 экспериментов с данными за 12 лет, ошибка прогноза урожаев не превысила 1,7 ц/га по всем четырем культурам. Как видно из рис. 7.10, функциональные ГИИС, разработанные по ПС-методологии, релевантно отображают моделируемые процессы и явления, что положительно сказывается на синхронности колебаний динамики и погрешности при прогнозировании урожаев сельскохозяйственных культур.

Лабораторные результаты полностью подтвердились в ходе опытной эксплуатации системы АГРО в ООО им. Ладушкина с августа 1998 по август 1999 г. на площади 300 га и с августа 1999 по август 2000 г. на площади 350 га.

Установлено также, что использование ГИИС, синтезированных с помощью пакета АГРО, позволяет:

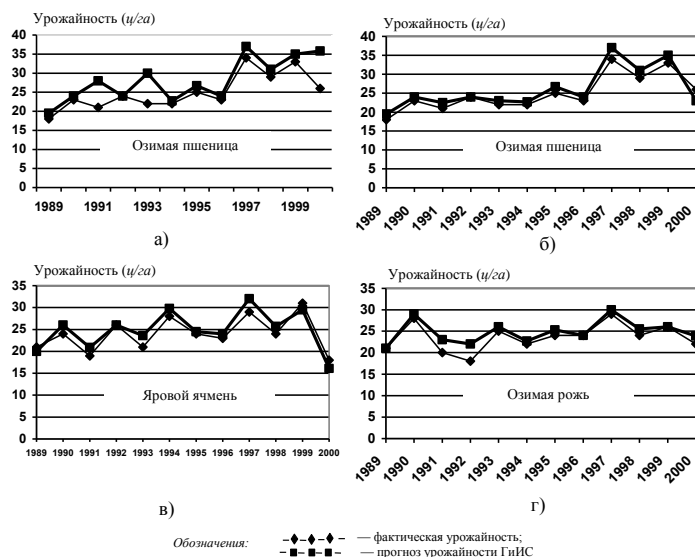


Рис. 7.10 — Результаты экспериментов

1) снизить ошибки прогнозов (по годовичному циклу производства) значений урожаев сельскохозяйственных культур до 5 — 10 % при благо-

приятных агроклиматических условиях возделывания и до 7 — 14 % при сложных условиях;

2) увеличить на 5 %, по нижней границе, урожайность сельскохозяйственной культур;

3) на 2 — 6 % снизить простои рабочих по организационным причинам и на 3 — 7 % простои техники и оборудования; на 3 — 11 % снизить расход горюче-смазочных материалов, удобрений, семян и т.д.;

4) получать дополнительную прибыль 12 — 15 %, а в отдельных случаях и до 25 — 30 % на один га; в 8 — 10 раз снизить затраты труда на подготовку технологической документации.

ЗАКЛЮЧЕНИЕ

Настоящая работа посвящена повышению эффективности интеллектуальных систем обработки информации и управления, в частности компьютерных систем поддержки принятия решений, путем разработки моделей, методологии и технологии автоматизированного проектирования функциональных гибридных интеллектуальных систем с учетом человеческого фактора.

На этом, одном из возможных путей решения сложных задач, получены несколько результатов.

К первому мы относим необходимость смещения акцентов с традиционного для науки и практики взгляда на сложную задачу, как на неделимый объект. Напротив, рассматривать ее следует как неоднородную, стратифицированную сущность, как задачу-систему,

требующую, прежде всего, тщательного анализа и углубленного понимания. Хочется надеяться, что неоднородность, как НЕ-фактор, усложняющий создание интеллектуальных систем, не только показана, но может быть и преодолена с меньшим количеством ошибок.

Второй результат — это выработка моделей, как нам кажется, полезных разработчику автоматизированных систем обработки информации и управления. Это, прежде всего, модели сложной задачи и ее подзадач, модель метода решения подзадач и модель функциональной ГиИС. Все эти объекты представлены гетерогенными системами, что открывает возможность рассматривать их не как «застывшие изваяния», а как эволюционирующие, развивающиеся сущности. Эти модели специфицируют объекты-оригиналы, объекты-прототипы и объекты-результаты гибридизации, внося в нее механизмы целеполагания. Эти механизмы послужат надежным ориентиром и спокойной гаванью Вашему кораблю, отправившемуся в мало изведанный путь решения сложных задач.

Третий результат — разработка ПС-методологии решения сложных задач методами функциональных гибридных интеллектуальных систем. Это наш вклад в «Практическое руководство по решению сложных задач в системах поддержки принятия решений». Хотелось бы и книгу назвать именно так. Но, думается, это было бы преждевременно. Не все здесь еще ясно, остаются и «белые пятна» в методах выполнения отдельных действий разработчика.

Наконец, четвертый результат — создание технологии решения сложных, неоднородных задач методами функциональных ГиИС. В таком детализированном виде она публикуется впервые. Однако даже и такая детализация, конечно, не исключит возможного недопонимания наших идей, моделей, необходимых действий, их последовательности и не ответит на вопрос: «Как написать программную систему?». Тем не менее, предлагаемая технология снимет многие проблемы на начальном этапе проекта, позволит накопить опыт, сэкономит Ваше время и снизит трудозатраты, т.е. сделает разработку эффективной.

В 2007 — 2009 гг. в ИПИ РАН исследования по методологии и технологии гибридизации будут продолжены.

ЛИТЕРАТУРА

1. Колесников А.В. Гибридные интеллектуальные системы. Теория и технология разработки / Под ред. А.М. Яшина. – СПб.: СПбГТУ, 2001. – 711 с.
2. Непомнящий Е.Г. Экономика и управление предприятием: Конспект лекций. – Таганрог: ТРТУ, 1997.
3. Поспелов Д.А. Принципы ситуационного управления // Известия АН СССР. Серия «Техническая кибернетика». – 1971. №2. – С.10 – 17.
4. Поспелов Д.А. Ситуационное управление: теория и практика. – М.: Наука, 1986. – 288 с.
5. Клыков Ю.И. Ситуационное управление большими системами. – М.: Энергия, 1974. – 134 с.
6. Вороновский Г.К. и др. Генетические алгоритмы. Искусственные нейронные сети и проблемы виртуальной реальности. – Харьков: Основа, 1997. – 112 с.

7. McCulloch W.S., and Pitts W.H. A logical calculus of the Ideas immanent in Neuruous Activity// Bull. of Mathematical biophysics. – 1943. №1. – P.115 – 119.
8. Hebb D. Organisation of Behaviour. – New York: Wiley, 1949.
9. Венда В.Ф. Системы гибридного интеллекта: Эволюция, психология, информатика. – М.: Машиностроение, 1990. – 448 с.
10. Гельфандбейн Я.А., Колесников А.В., Рудинский И.Д. Семиотико–статистическая модель поведения транспортного узла // Вопросы кибернетики: сб. научн. тр. Вып. 100. – М.: Научный Совет по комплексной проблеме «Кибернетика» при Президиуме АН СССР, 1983. – С. 40 – 49.
11. Прангишвили И.В. Доклад на юбилейном заседании, посвященном 60–летию ИПУ РАН. – http://www.ipu.ru/publ/p1_p3.htm.
12. Виханский О.С., Наумов А.И. Менеджмент: Учебник.– 3-е изд. – М.: Гардарики, 2002.–528 с.
13. Тарасов В.Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. – М.: Эдиториал УРСС, 2002. – 352 с.
14. Bell D. The Coming of Post–industrial Society. A Venture in Social Forecasting. – N.Y.: Basic Books, Inc., 1973.
15. Алексеева И. Возникновение идеологии информационного общества // Информационное общество. – 1999. – Вып. 1.
16. Цвылев Р.И., Постиндустриальное развитие. Уроки для России. – М.: Наука, 1996.
17. Кастельс М. Информационная эпоха: экономика, общество и культура. – М.: ГУ ВШЭ, 2000.
18. Рассел С., Норвиг П. Искусственный интеллект: современный подход, 2-е изд.: Пер. с англ. – М.: Вильямс, 2006.– 1408 с.
19. БСЭ. Т.21. – М.: Советская энциклопедия, 1971. – С. 7.
20. Ожегов С.И. Словарь русского языка. – М.: Советская энциклопедия, 1972. – С.556.
21. Ларичев О.И. Теория и методы принятия решений, а также Хроника событий в Волшебных странах: Учебник. Изд. 2-е, перераб. – М.: Логос, 2003. – 392 с.
22. Самсонова М.В., Ефимов В.В. Технология и методы коллективного решения проблем: Учебное пособие. – Ульяновск: УлГТУ, 2003. – 152 с.
23. Перегудов Ф.И., Тарасенко Ф.П. Введение в системный анализ: Учебное пособие для вузов. – М.: Высшая школа, 1989. – 367 с.
24. Акоф Р., Сасиени М. Основы исследования операций. – М.: Мир, 1971. – 536 с.
25. Пойа Д. Математическое открытие. Решение задач: основные понятия, изучение и преподавание. – М.: Наука, 1970. – 452 с.
26. Лэсдом Л.С. Оптимизация больших систем. – М.: Наука, 1975. – 432 с.
27. Russell S.J., Norvig P. Artificial Intelligence. A Modern Approach. – Prentice Hall, 1995. – 932 p.

28. Самюэль А. Некоторые исследования возможности обучения машин на примере игры в шашки // Вычислительные машины и мышление / Под ред. Э. Фейгенбаума и Дж. Фельдмана. – М.: Мир, 1967. – С. 71 – 111.
29. Попов Э.В., Фоминых И.Б. и др. Статические и динамические экспертные системы. – М.: Финансы и статистика, 1996. – 320 с.
30. Kolesnikov A., and Yashin A. Hybrid Simulation of Stratified Systems // Mathematical Modelling and Analysis MMA 99: Pr. of the 4th Intern. Conf. – Vilnius, 1999. – P. 51.
31. Kolesnikov A., Yashin A. Hybrid Modelling in Stratified Decision Support Systems –1 // Mathematical Modelling and Analysis. V.5. – Vilnius: Technika, 2000. – P. 108 – 118.
32. Kolesnikov A., Yashin A. Hybrid Modelling in Stratified Decision Support Systems – 2 // Mathematical Modelling and Analysis. V. 6. №1. – Vilnius: Technika, 2001. – P. 97 – 105.
33. БСЭ. Т. 7. – М.: Советская энциклопедия, 1971. – С. 53.
34. БСЭ. Т. 6. – М.: Советская энциклопедия, 1971. – С. 439.
35. Александров Е.А. Основы теории эвристических решений. – М.: Сов. радио, 1975. – 320 с.
36. Ларичев О.И. Наука и искусство принятия решений. – М.: Наука, 1979. – 200 с.
37. Ларичев О.И. Теория и методы принятия решений. – М.: Логос, 2000. – 296 с.
38. Забейко М.И. Data Mining & Knowledge Discovery in Databases: Предметная область, задачи, методы и инструменты // Шестая национальная конференция по искусственному интеллекту с международным участием КИИ 98: сб. научн. тр. Т. 2. – Пушкино, 1998. – С. 592 – 600.
39. Нильсон Н. Принципы искусственного интеллекта. – М.: Радио и связь, 1985. – 376 с.
40. Гладун В.П. Каким должен быть интеллектуальный компьютер // Международ. научно–практическая конференция KDS – 2001: сб. научн. тр. Т.1.– СПб.: Северо–западный государственный заочный технический университет, 2001. – С. 136 – 143.
41. Емельянов В.В., Зафиров Э.Г. Гибридные системы на базе генетических оптимизационных алгоритмов и интеллектуального имитационного моделирования // Седьмая национальная конференция по искусственному интеллекту с международным участием КИИ 2000: сб. научн. тр. – М.: Физматлит, 2000. – С. 780 – 788.
42. Тарасов В.Б. Системно–организационный подход в искусственном интеллекте // Известия РАН. Теория и системы управления. – 1997. – №3. – С. 6 – 13.
43. Тарасов В.Б. Искусственная жизнь и нечеткие эволюционные многоагентные системы – основные теоретические подходы к построению интеллектуальных организаций // Известия РАН. Теория и системы управления. – 1998. – №5. – С. 12 – 23.

44. Нариньяни А.С. Искусственный интеллект: стагнация или новая перспектива // Шестая национальная конференции по искусственному интеллекту с международным участием КИИ 98: сб. научн. тр. Т.1. – Пущино, 1998. – С. 15 – 29.
45. Кун Т. Структура научных революций / Пер. с англ. И.З. Налетова. – М.: Прогресс, 1977. – 288 с.
46. Миронова Т.С., Плесневич Г.С. Гибридные модели знаний // Известия РАН. Серия «Техническая кибернетика». – 1994. – №2. – С. 56 – 70.
47. Горюнов И. Как соединить мозг человека с компьютером? Эту проблему решают российские ученые // Поиск, 2006. – <http://www.pisknews.ru>
48. Волкова В.Н., Денисов А.А. Основы теории систем и системного анализа. – СПб.: СПбГТУ, 1999. – 512 с.
49. Николаев В.И., Брук В.М. Системотехника: методы и приложения. – Л.: Машиностроение, 1985. – 199 с.
50. Пешель М. Моделирование сигналов и систем. – М.: Мир, 1981. – 302 с.
51. Шеннон Р. Имитационное моделирование систем – искусство и наука. – М.: Мир, 1978. – 418 с.
52. Аксенов В.В., Салмин И.Д. Постановка задачи коммивояжера с временными окнами и ее решение. – <http://library.mephi.ru/data/scientific-sessions/2006/t2/2-1-26.doc>.
53. Интервью с директором института мозга и человека РАН С.В. Медведевым // Секретные материалы XX века. – 2001. – N19 (63). – С. 10 – 11.
54. Роль информации в современном обществе. – http://www.ime.ru/disnsce/demo/e100_01.htm
55. ИНТЕРНЕТ–страница фирмы Interface Ltd.: программные продукты Rational Software. – <http://www.interface.ru>.
56. Г. Хакен. Тайны природы. Синергетика: наука о взаимодействии. – Москва – Ижевск, 2003. – 25 с.
57. Antsaklis J., Stiver J., Lemmon M.D. Hybrid System Modeling and Autonomous Control Systems // In Hybrid Systems/ R.L. Grossman, A. Nerode, A.P. Ravin, and H. Rischel, Eds., Vol. 736 of Lecture Notes in Computer Science. New York: Springer-Verlag, 1993.
58. Back A., Guckenheimer J., and Myers M. A. Dynamical Simulation Facility for Hybrid Systems // Technical Report 92-6, Mathematical Science Institute. Cornell University, 1992.
59. Brockett R.W. Hybrid Models for Motion Control Systems// Technical Report CICS-P-364. Center for Intelligent Control Systems. Massachusetts Institute of Technology, 1993.
60. Nerode A., and Kohn W. Models for Hybrid Systems: Automata, Topologies, Stability // Technical Report 93-11. Mathematical Science Institute, Cornell University, 1993.
61. Tavernini L. Differential Automata and their discrete simulators// Nonlinear Analysis, Theory, Methods, and Applications. – 1987. –11(6). – P.665 – 683.
62. Экспертные системы: состояние и перспективы / Под. Ред. Д.А. Поспелова. – М.: Наука, 1989.

63. Борисов А.Н. Принципы построения и реализации гибридных экспертных систем // Гибридные экспертные системы в задачах проектирования сложных технических объектов. Матер. краткосрочного научно-технического семинара «Гибридные экспертные системы в задачах проектирования сложных технических объектов»: сб. тез. докл. – СПб: Дом НТП, 1992. – С. 6 – 9.
64. Рыбина Г.В. Интегрированные экспертные системы: современное состояние, проблемы и тенденции // Известия РАН. Теория и системы управления. – 2002. – №5. – С. 111 – 126.
65. Ярушкина Н.Г. Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004. – 320 с.
66. Гаврилов А.В. Гибридные интеллектуальные системы. – Новосибирск: Изд-во НГТУ, 2003. – 164 с.
67. Мациевский С.В., Толстень О.В. Нечеткие системы: Учебное пособие. – Калининград: Изд-во РГУ им. И.Канта, 2006. – 256 с.
68. Эрлих А.И. Проблемы моделирования в интеллектуальных системах // Вестн. МГТУ. – 1/94. – Системы искусственного интеллекта: спец. выпуск. – М.: МГТУ, 1994.
69. Рыбина Г.В. Современные экспертные системы: тенденции к интеграции и гибридизации // Приборы и системы. Управление, контроль, диагностика. – 2001. – №8.
70. Рыбина Г.В. Проектирование систем, основанных на знаниях. – М: МИФИ, 1997.
71. Поспелов Г.С. Искусственный интеллект – основа новой информационной технологии. – М.: Наука, 1988.
72. Goonatilake S., and Khebbal S. Intelligent Hybrid Systems // The First Singapore Int. Conference on Intelligent Systems: proceedings. – 1992. – P. 356 – 364.
73. Hilario M. An overview of strategies for neurosymbolic integration // The IJCAI Workshop on Connectionist –Symbolic Integration: From Unified to Hybrid Approaches. Montreal. – 1995. – P. 234 – 250.
74. Medsker L.R. Hybrid Intelligent Systems. – Kluwer Academic Publ., 1995.
75. Kasabov N., and Kozma R. Hybrid Intelligent Adaptive Systems: a Framework and a Case Study on Speech Recognition // Intelligent Systems. 1998. – V.13. – P. 455 – 466.
76. Sun R. Robust Reasoning: Integrating rule-based and similarity-based Reasoning // Artificial Intelligence . – 1995. –75(2). – P. 214 – 295.
77. Кузин Л.Т. Основы кибернетики: в 2 т. – Т. 1.– Математические основы кибернетики. – М.: Энергия, 1973. – 504 с.
78. Кузин Л.Т. Основы кибернетики: В 2 т. – Т. 2. – Основы кибернетических моделей. – М.: Энергия, 1979. – 582 с.
79. Поспелов Г.С., Ириков В.А. Программно-целевое планирование и управление (введение). – М.: Советское радио, 1976. – 440 с.

80. Борисов А.Н. и др. Диалоговые системы поддержки принятия решений на баз мини-ЭВМ: Информационное, математическое и программное обеспечение. – Рига: Зинате, 1986. – 195 с.
81. Ларичев О.И. Объективные модели и субъективные решения. – М.: Наука, 1987.– 144 с.
82. Искусственный интеллект. В 3 кн. – Кн. 2. – Модели и методы / Под ред. Д.А. Пospelова. – М.: Радио и связь, 1990. – 304 с.
83. Толковый словарь по искусственному интеллекту / Авторы–составители А.Н. Аверкин, М.Г. Гаазе–Рапопорт, Д.А. Пospelов. – М.: Радио и связь, 1992. – 256 с.
84. Трахтенгерц Э.А. Компьютерный анализ в динамике принятия решений // Приборы и системы управления. – 1997. – №1. – С. 49–56.
85. Рабинович З.Л., Воронков Г.С. Представление и обработка знаний во взаимодействии сенсорной и языковой нейросистем человека // Управляющие системы и машины: международный научный журнал Института кибернетики им. В.М. Глушкова НАН Украины. – 1998. – №2. – С. 3 – 11.
86. Donskoy V.I. Case-, Knowledge-, and Optimisation – Based Hybrid Approach in AI // The 11 Int. Conference on Industrial and Engineering Applications of AI and Expert Systems IEA–98–AIE: proceedings. – Benicassim. Spain, 1998. – P. 520 – 527.
87. Lobo F.G., and Goldberg D.E. Decision Making in a Hybrid Genetic Algorithms // Report No. 96009. – Illinois: University of Illinois. Department of General Engineering. PliGAL, 1996.
88. Колесников А.В., Клачек П.М.. Неоднородное моделирование для интеллектуального управления биопроизводственной системой // Интеллектуальные системы и информационные технологии управления ИСИТУ–2000–IS&ITC: сб. тр. международной конференции. – СПб.: СПбГТУ, 2000. – С. 359 – 363.
89. Кулагин О.А. Принятие решений в организациях // ИПКПРО ОГПУ / Банк педагогической информации. – http://bank.orenipk.ru/Text/t12_5.htm.
90. Тайны мышления: Популярная психология / Пер. с англ. – М.: Мой мир Гмбх&Ко. КГ, 2005. – 176 с.
91. Трахтенгерц Э.А. Методы генерации, оценки и согласования решений в распределенных системах поддержки принятия решений // Автоматика и телемеханика. – 1995. – №4. – С. 3 – 52.
92. Power D.J. Decision Support Systems: Concepts and Resources for Managers.– Quorum Books, Greenwood Publishing, 2002. – 272 p.
93. Козелецкий Ю. психологическая теория решений. – М.: Прогресс, 1979.
94. Корнилова Т.В. Психология риска и принятия решений. – М.: Аспект Пресс, 2003.
95. Кочетков В.В. Индивидуальные психологические проблемы принятия решений. – М.: Наука, 1993.
96. Логика конечнзначных предикатов на основе неравенств: Учебное пособие. – СПб.: С.– Петерб. университет, 2000. – 268 с.

97. Вентцель Е.С. Исследование операций: задачи, принципы, методология. – М.: Наука, 1988. – 208 с.
98. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. – СПб.: Питер, 2000. – 384 с.
99. Налимов В.В. Вероятностная модель языка. О соотношении естественных и искусственных языков. – М.: Наука, 1979.
100. Трахтенгерц Э.А. Компьютерная поддержка переговоров при согласовании управленческих решений. Серия «системы и приборы управления». – М.: СИНТЕГ, 2003. – 284 с.
101. Голубков Е.П. Сущность и характерные особенности управленческих решений // Менеджмент в России и за рубежом. – 2003. – №2.
102. Райков А.Н. Ситуационная комната для поддержки корпоративных решений.– http://www.osp.ru/os/1999/07-08/179889/_p2.html.
103. Райков А.Н. Тенденции развития систем поддержки принятия решений // Информационное общество. – 2004. – Вып. 4. – С. 36 – 40.
104. Harrison E.F, Pelletier M.A. Сущность управленческого решения.– <http://b-news.narod.ru>.
105. Eom S.B. Decosion support systems research: reference disciplines and a cumulative tradition//The International Journal of Management Science. – 23. – 5, October 1995. – P. 511 – 523.
106. Inmon W.H. Building the Data Werehouse 3RD Edition. – Wiley&Sons, 2002. – 432 p.
107. Рассом Ф. Тенденции программного обеспечения в области визуализации для бизнес-пользователей. – <http://www.intersoft.ru>.
108. Плотинский Ю.М. Визуализация информации. – М.: Изд. МГУ, 1994.
109. Системы поддержки принятия решений. – <http://www.oviont.ru>.
110. Горский П. Уточнение понятия «система поддержки принятия решений».– http://www.gorsky.ru/Articles/DSS_term.html.
111. Системы поддержки принятия решений.– <http://www.intuit.ru/departament/database/datamiming/17>.
112. Интернет университет информационных технологий. Лекция “Data Mining”. – <http://www.intuit.ru>.
113. Корнеев С.В. Системы поддержки принятия решений в бизнесе // Сети и бизнес. – №6. – 2005. – <http://www.management.com.ua>.
114. Егоров А.И. Математические задачи управления системами // Междунар. конференция «Интеллектуальное управление: новые интеллектуальные технологии в задачах управления ICIT 99»: сб. трудов. – М.: Наука, 1999. – С. 160 – 163.
115. Ерофеев А.А., Поляков А.О. Интеллектуальные системы управления. – СПб.: СПбГТУ, 1999. – 256 с.
116. Колесников А.В. Источники сложности моделирования принятия решений в условиях риска и пути ее преодоления // Международная конференция, посвященная 40-летию пребывания КГТУ на Калининградской земле и 85-летию высшего рыбохозяйственного образования в России. ч. 4: сб. тез. докл – Калининград: КГТУ, 1999. – С. 146 – 148.

117. Месарович М., Мако Д., Такахара И. Теория иерархических многоуровневых систем. – М.: Мир, 1973. – 344 с.
118. Ларичев В.М., Поляков А.О. Информодинамика или путь к Миру открытых систем. – СПб.: Изд-во СПбГТУ, 1999.
119. Поляков А.О., Юсупов Р.М., Ерофеев А.А. Интеллектуальные системы и информационные технологии управления. – <http://www.inftech.webservis.ru>.
120. Захаров В.Н. Интеллектуальные системы управления: Основные понятия и определения // Известия РАН. Теория и системы управления. – 1997. – №3. – С. 138 – 145.
121. Андрейчиков А.В., Андрейчикова О.Н. Интеллектуальные информационные системы: Учебник. – М.: Финансы и статистика, 2004. – 424 с.
122. Вагин В.Н., Еремеев А.П. Некоторые базовые принципы построения интеллектуальных систем поддержки принятия решений реального времени // Известия РАН. Теория и системы управления – 2001. – № 6. – С. 114 – 123.
123. Экспертные системы. принципы работы и примеры / Под ред. Р. Форосайта. – М.: Радио и связь, 1987.
124. Построение экспертных систем // Пер. с англ. / Под ред. Ф. Хейса-Рота, Д. Уотермана, Д. Лената. – М.: Мир, 1987. – 441 с.
125. Аристотель. Сочинения. В 4 т. – Т. 2. – М.: Мысль, 1978.
126. Волкова В.Н., Темников Ф.Е. Методы формализованного представления (отображения) систем // Текст лекций. – М.: ИПКИР, 1974. – 114 с.
127. Hertzmann C.S. Symbolical Reasoning about Numerical Data: A Hybrid Approach // Applied Intelligence. – 1997. – №7. – Р. 339 – 354.
128. Поспелов Д.А., Кузнецов О.П. Знания и рассуждения в гуманитарных науках // Новости искусственного интеллекта. – 1996. – №2. – С. 93 – 98.
129. Микони С.В. Модели и базы знаний: Учебное пособие. – СПб.: Петербургский гос. университет путей сообщения, 2000. – 155 с.
130. Шишаков М.Л. Гибридная экспертная система концептуального проектирования. – Минск: Изд-во АН республики Беларусь, ин-т технической кибернетики, 1992.
131. Представление и использование знаний / Под ред. Х. Уэно, М. Исудзука. – М.: Мир, 1989. – 220 с.
132. Управление, информация, интеллект / Под ред. А.И. Берга и др. – М.: Мысль, 1976. – 384 с.
133. Захаров В.Н., Ульянов С.В. Нечеткие модели интеллектуальных регуляторов и систем управления. IV. Имитационное моделирование // Известия РАН. Серия «Техническая кибернетика». – 1994. – №5. – С. 168 – 210.
134. Афанасьев В.Г. Мир живого: системность, эволюция и управление. – М.: Политиздат, 1986. – 334 с.
135. Поспелов Д.А. Метафора, образ и символ в познании мира // Новости искусственного интеллекта. – 1998. – №1. – С. 94 – 114.
136. БСЭ. Т.18. – М.: Советская энциклопедия, 1971. – С.1377 – 1378.
137. Люгер Дж.Ф. Искусственный интеллект: стратегии и методы решения сложных проблем. – М.: Изд. дом «Вильямс», 2003. – 864 с.

138. Воинов А.В. Моделирование интуитивных рассуждений эксперта методами психосемантики и вывода // *Новости искусственного интеллекта.*–1998.– №2. – С.130 – 142.
139. БСЭ. Т.1. – М.: Советская энциклопедия, 1971. – С.1677 – 1678.
140. Goldberg A. *Natural and Artificial Creative Processes in Nature, Science and Engineering.* – Tel–Aviv: Pilies Studio, 2002.
141. Словарь иностранных слов / Под ред. И.В. Лехина и Ф.Н. Петрова. – М.: Гос. изд. иностранных. и национальных словарей, 1955. – 856 с.
142. Советов Б.Я., Яковлев С.А. Моделирование систем. – М.: Высшая школа, 1985. – 271 с.
143. Энциклопедия кибернетики. В 2 т. – Киев: АН УССР, гл. ред. Украинской советской энциклопедии, 1974.
144. Бусленко Н.П. Моделирование сложных систем. – М.: Наука, 1978. – 400 с.
145. Колесников А.В., Петухов О.А. Моделирование систем. – Л.: Северо–западный заочный политехнический институт, 1981.– 72 с.
146. Уотермен Д. Руководство по экспертным системам. – М.: Мир, 1989. – 184 с.
147. Поспелов Д.А. Продукционные модели // *Искусственный интеллект.* В 3 кн. Кн. 2. Модели методы: Справочник / Под. ред. Д.А. Поспелова. – М.: Радио и связь, 1990. – С. 49 – 56.
148. Medsker L.R. *Hybrid Neural Network and Expert Systems.* – Boston/Dordrecht/London: Kluwer Academic Publishers, 1994. – 240 p.
149. Newell A., Simon H. *Human Problem Solving.* – Englewood Cliffs, NJ: Prentice Hall, 1972.
150. Джексон П. Введение в экспертные системы: Пер. с англ.: Учебное пособие. – М.: Изд. дом «Вильямс», 2001.– 624 с.
151. Джарратанно Дж., Райли Г. Экспертные системы: принципы разработки т программирование.: Пер. с англ.– М.: ООО «И.Д. Вильямс», 2007. – 1152 с.
152. Поспелов Д.А. Логико–лингвистические модели в системах управления. – М.: Энергоиздат, 1981. – 232 с.
153. Nonavar V. *Symbolic Artificial Intelligence and Numeric Artificial Neural Networks: Towards A Resolution of Dichotomy* // *Technical Report TR 94–14.* – Iowa: Iowa State University, 1994.
154. Прикладные нечеткие системы // Пер. с япон. / Под ред. Т. Тэрано, К. Асаи, М. Сугэно. – М.: Мир, 1993. – 368 с.
155. Глухов Д.О. Экспертная система на нечетких продукционных правилах для обследования сложного объекта // *Междун. конференции по мягким вычислениям и измерениям SCM–98: сб. докл.– Т.2.* – СПб, 1998. – С. 174 – 176.
156. Нечеткие множества в моделях управления и искусственного интеллекта // Аверкин А.Н., Батыршин И.З., Блишун А.Ф. / Под ред. Д.А. Поспелова. – М.: Наука, 1986. – 312 с.
157. Rutkowska D., Pilinski M., Rutkowski L. *Sieci neuronowe, algoritmy genetyczne i systemy rozmyte.* – Warszawa–Lodz: Wyd. naukowe PWN, 1997. – 411 s.

158. Мацневский С.В. Нечеткие множества: Учебное пособие.– Калининград: Изд-во КГУ, 2004.– 176 с.
159. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы: Пер. с польск. И.Д. Рудинского.– М.: Горячая линия – Телеком, 2004. – 452 с.
160. Кобринский Б.А. Символьно–образный подход в искусственном интеллекте // Седьмая национальная конференция по искусственному интеллекту с международным участием КИИ–2000: сб. докл.– Т. 2. – М.: Наука, 2000. – С. 601 – 608.
161. Фоминых И.Б. Интеграция нейронных и символьно–логических моделей в интеллектуальных технологиях // Седьмая национальная конференция по искусственному интеллекту с международным участием КИИ–2000: сб. докл.– Т. 2. – М.: Наука, 2000.– С. 588 – 596.
162. Медведев В.С., Потемкин В.Г. Нейронные сети. MATLAB 6 / Под. общ. ред. В.Г. Потемкина.– М.: ДИАЛОГ–МИФИ, 2002. – 469 с.
163. Specht D.F. Probabilistic Neural Networks for Classification, Mapping or Associative Memory // IEEE International Conference on Neural Network: proceedings.– 1988. – Vol. 1. – P. 525 – 532.
164. Specht D.F. Probabilistic Neural Networks // IEEE Trans. Neural Networks.– 1990. – Vol. 3 (1)(12).– P. 109 – 118.
165. Круглов В.В., Дли М.И., Голунов Р.Ю. Нечеткая логика и искусственные нейронные сети: Учеб. пособие. – М.: Изд. Физико–математической литературы, 2001. – 224 с.
166. Уоссермен Ф. Нейрокомпьютерная техника. – М.: Мир, 1992. – 126 с.
167. Осовский С. Нейронные сети для обработки информации / Пер. с польского И.Д. Рудинского.– М.: Финансы и статистика, 2002. – 344 с.
168. Holland J.H. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Application to Biology, Control, and Artificial Intelligence. – Michigan: University of Michigan, 1975.
169. Goldberg D.E. Genetic Algorithm in Search and Machine Learning. – Addison–Wesley Publishing Company, Inc., 1989.
170. Kirkpatrick S., Toulouse G. Configuration space analysis of traveling salesman problems. J. de Phys. – 1985. – V.46. – P. 1277 – 1292.
171. Tanese R. Parallel Genetic Algorithms for Hypercube // The Second Int. Conf. on Genetic Algorithms: proceedings / J.J.Grefenstette (Editor). – Lawrence Erlbaum Associates. – 1987. –177.– P. 400 – 408.
172. Manderick B, and Spiessers P. Fine–Grained Parallel Genetic Algorithms // The Third Int. Conf. on Genetic Algorithms: proceedings / J.D.Schffer (Editor).– Morgan Kauffman. – 1989.– 428.– P. 456 – 500.
173. Horn J., Nafpliotis N., and Goldberg David E. Multiobjective Optimization Using the Niche Pareto Genetic Algorithm // Technical Report №93005. University of Illinois at Urbana–Champaign.– IlliGAL, 1993.
174. Смит С. Адаптивные обучающиеся системы // Экспертные системы. Принципы работы и примеры: Пер. с англ. / Под ред. Р. Форсайта.– М.: Радио и связь, 1987. – С. 158 – 177.

175. Schank R.C., and Abelson, R.P. Scripts, Plans, Goals and Understanding. – Erlbaum, Hillsdale, New Jersey, 1977.
176. Schank R. Dynamic memory: a theory of reminding and learning in computers and people. – Cambridge University Press, Cambridge, 1982.
177. Kolodner J. L. Maintaining Organization in a Dynamic Long–Term Memory// Cognitive Science. –1983. – 7(iv). – P. 243 – 80.
178. Kolodner J. L. Reconstructive Memory: A Computer Model// Cognitive Science. – 1983. – 7(iv). – P. 281 – 28.
179. Simpson R. L. A Computer Model of Case–Based Reasoning in Problem Solving: An Investigation in the Domain of Dispute Mediation // Technical Report GIT–ICS–85/18.– Georgia Institute of Technology, School of Information and Computer Science, Atlanta, 1985.
180. Hammond K.J. CHEF: A Model of Case–Based Planning // American Association for Artificial Intelligence, AAAI–86, August 1986: proceedings. – Philadelphia, PA, 1986.
181. Sycara E. P. Resolving adverbial conflicts: An approach to Integrating Case–Based and Analytic Methods // Technical Report GIT–ICS–87/26.– Georgia Institute of Technology, School of Information and Computer Science, Atlanta, 1987.
182. Richter A.M., and Weiss S. Similarity, uncertainty and case–based reasoning in PATDEX // In, Automated reasoning, essays in honour of Woody Bledsoe.– Kluwer R.S. Boyer (ed.), 1991. – P. 249 – 265.
183. Sharma S., and Sleeman D. REFINER: A Case–Based Differential Diagnosis Aide for Knowledge Acquisition and Knowledge Refinement // European Working Session on Learning EWSL 88: proceedings. – 1988.– P. 201 – 210.
184. Porter B.W., and Bareiss E.R. PROTOS: An experiment in knowledge acquisition for heuristic classification tasks // The First International Meeting on Advances in Learning (IMAL): proceedings.– Les Arcs, France, 1989 – P. 159 – 74.
185. Кузнецов О.П. Быстрые процессы мозга и обработка образов // Новости искусственного интеллекта. – 1998. – №2. – С.17 – 130.
186. Пригожин И. От существующего к возникающему: Время и сложность в физических науках. 2–е изд.: Пер. с англ.– М.: Едиториал УРСС, 2002.
187. Хакен Г. Информация и самоорганизация: Микроскопический подход к сложным системам: Пер. с англ. – М.: Мир, 1991.
188. Smith R.G. The Contract Net Protocol: High Level Communication and Control in a Distributed Problem Solver // IEEE Transactions on Computers.– 1980. – Vol. 29. – №12. – P. 1104 – 1113.
189. Городецкий В.И. Многоагентные системы: основные свойства и модели координации поведения // Информационные технологии и вычислительные системы. – 1998. – №1. – С. 22 – 34.
190. Rosenshein J., Zlotkin P. Rules of Encounter: Designing Conventions for Automated Negotiation Among Computers. – Cambridge MA: MIT Press, 1994.
191. Castelfranchi C., Werner E. Artificial Social Systems. – Berlin: Springer, 1991.

192. Курейчик В.М. Генетические алгоритмы. Состояние. Проблемы. Перспективы // Изв. РАН. Теория и системы управления. – 1999.– №1.– С. 144 – 160.
193. Поляязыковые семиотические модели для математического обеспечения АСУ, АСУТП и САПР в рыбной промышленности // Отчет о НИР «Информационное, математическое и программное обеспечение АСУ, АСУТП и САПР в рыбной промышленности».– N гос. рег. 01860015101.– инв. N 02860109459.– Калининград: КТИРПиХ, 1986.
194. Erman L., Hayes–Roth F., Lesser V., Reddy D. The HEARSAY–2 Speech Understanding System: Integrating Knowledge to Resolve Uncertainty // ACM Computers Surveys. – 1980. – Vol.12.
195. Hewitt C. Viewing Control Structures as Patterns of Message Passing// Artificial Intelligence. – 1977. – Vol. 8. – №3. – P. 323 – 364.
196. Lenat D. BEINGS: Knowledge as Interacting Experts // The 1975 IJCAI Conf.: proceedings. – 1975.– P. 126 – 133.
197. Хорошевский В.Ф. Поведение интеллектуальных агентов: модели и методы реализации // 4-й Международный семинар по прикладной семиотике, семиотическому и интеллектуальному управлению: сб. научн. тр. – Переславль–Залесский: РАИИ, 1999. – С. 5 – 20.
198. Witsenhausen H.S. A Class of Hybrid–State Continuous Time Dynamic Systems // IEEE Trans. on Automatic Control. – 1966. – 2.– V. 11. – P. 161 – 167.
199. Branicky M.S., Bonkar V.S., and Mitter S.K. A Unified Framework for Hybrid Control: Background, Model and Theory // The 33 IEEE Conference on Decision and Control: proceedings. – Lake Buena Vista, 1994.
200. Puri A.,Varaija P. Decidability of Hybrid Sustersms with Rectangular Differential Inclusions, Computer–Aided Verification, LN CS 818.– Springer–Verlag, 1994.
201. Walsh G. On Race Conditions for Networked Control Systems // The 30 CISS: proceedings.– Princeton, 1996. – P. 411– 415.
202. Praehofer P., and Zeigler B. Automatic Abstraction of Event–Based Control Models from Continuous Base Models // IEEE Trans. Systems, Man and Cybernetics. – 1995. – V. 1– P. 456 – 477.
203. Осипов Г.С. Динамика в системах, основанных на знаниях // Изв. РАН. Теория и системы управления. – 1998. – №5.– С. 24 – 28.
204. Рыбин В.М. и др. Интегрированные экспертные системы для проектирования информационно–измерительных систем и анализа систем управления сложными объектами // IV национальной конф. с международным участием КИИ–94: сб. научн. тр.– Т. 2.– Рыбинск: АИИ, 1994.
205. Федунев Б.Е. Проблемы разработки бортовых оперативно — советующих экспертных систем для антропоцентричных объектов // Изв. РАН. ТИСУ. – 1996. – № 5.
206. Гельфандбейн Я.А., Рудинский И.Д., Новожилова Н.В. Гибридные много-модельные системы. Вопросы реализации // Известия АН СССР Серия «Техническая кибернетика». – 1991. – С. 74 – 183.

207. Medsker L.R., 1996. Hybrid Intelligent Systems // International Journal of Computational Intelligence and Organizations. – 1996. – Vol. 1.– P. 10 – 20.
208. Intelligent Hybrid Systems / Ed. S. Goonatilake, and S. Khebbal. – Wiley & Sons, 1995.
209. McGarry K., Wermter S., and MacIntyre J. Hybrid Neural Systems: From Simple Coupling to Fully Integrated Neural Networks // Neural Computing Surveys. – 1999.– 2.– P. 62 – 93.
210. Edelman G. Neural Darwinism: The Theory of Neural Group Selection. – New York: Basic Books, 1987.
211. Gallant S.I. Neural Network learning and Expert Systems. – Cambridge. Massachusetts: The MIT Press, 1993.
212. Sun R. Robust Reasoning: Integrating rule-based and similarity-based Reasoning // Artificial Intelligence. – 1995. – 75(2).– P. 214 – 295.
213. Benitez J., Castro J., and Requena J.I. Are Artificial Neural Networks Black Boxes // IEEE Transactions on Neural Networks. – 1997. – 8(5). – P. 1156 – 1164.
214. Ultsch A., Mantyk R., and Halmans G. Connectionist knowledge acquisition tool: CONKAT // Artificial Intelligence Frontiers in Statistics: AI and Statistics 111 / Ed. J. Hand. – Chapman and Hall, 1993. – P. 256 – 263.
215. Jang J.-S.R., and Sun C.T. Functional Equivalence between Radial Basis Function Networks and Fuzzy Inference Systems // IEEE Transactions Neural Networks. – 1993. – 4.– P. 156 – 158.
216. Omlin C.W., and Giles C.L. Extraction and insertion of Symbolic Information in Recurrent Neural Networks // Artificial Intelligence and Neural Networks: Steps Towards Principled Integration / Eds. V. Honavar, and L. Uhrs. – San Diego: Academic Press, 1994. – P. 271 – 299.
217. Fu L. Mapping Rule-Based Systems into Neural Architectures // Knowledge-Based Systems. – 1990. – 3(1). – P. 48 – 56.
218. Wu X., and Hughes J.G. HKBCN – A Hybrid Intelligent System for Knowledge Revising // Asia-Pacific Software Engineering Conference and International Computer Science Conference ICSC'97: proceedings. – Hong Kong, 1997. – P. 106 – 114.
219. Towell G., and Shavlik J. Knowledge-Based Neural Networks // Artificial Intelligence. – 1994. – V. 69 (70). – P. 119 – 165.
220. Hruska S., Kuncicky D., and Lacher R. Hybrid Learning in Expert Networks // The Int. Joint Conf. on Neural Networks: proceedings. – 1992. – V2. – P. 117 – 120.
221. Pop E., Hauward R., and Diederich J. RULENEG: Extracting Rules from a Trained ANN by Step-wise Negation // Technical report. – QVT NRC, 1994.
222. Graver M.W., and Shavlik J.M. Using Sampling and Queries to Extract Rules from Trained Neural Networks // The 11 International Conference on Machine Learning: proceedings, 1993.– P. 37 – 45.
223. Pratt L.Y., and Christensen A.N. Relaxing the Hyperplane Assumption in the Analysis and Modification of Back-Propagation networks // Cybernetics and

- Systems – 94 World Scientific / Ed. Robert Trappl. – Singapore, 1994. – P. 1711 – 1718.
224. Web–page of MultiLogic Inc. – <http://www.multilogic.com>.
225. Fuzzy CLIPS v 6.04. Institute for Information Technology. National Research Council of Canada. – <http://ai.iit.nrc.ca/fuzzy/fuzzy.html>.
226. Takagi H., Suzuki N., Koda T., and Kojima Y. Neural Networks Designed on Approximate Reasoning Architecture and their Applications // IEEE Transactions Neural Networks. – 1992.– 5.– V. 3.– P. 752 – 760.
227. Vuorimaa P. Fuzzy Self–Organizing map // Fuzzy Sets and Systems. – 1994. – V. 66. – P. 223 – 231.
228. Pedrycz W. Fuzzy neural Networks with Reference Neurones as Pattern Classifiers // IEEE Transactions Neural Networks. – 1992. – 5.– V. 3.– P. 770 – 775.
229. Gupta M.M., and Rao D.H. On the Principles of Fuzzy Neural Networks // Fuzzy Sets and Systems. – 1994. – V. 61.– P. 1 – 18.
230. Меркурьева Г.В., Меркурьев Ю.А. Экспертные системы имитационного моделирования (Обзор) // Известия РАН. Серия «Техническая кибернетика». – 1991.– №3.– С. 156 – 173.
231. Емельянов В.В., Ясиновский С.И. Интеллектуальная имитационная среда PDO–IDS для моделирования сложных систем и процессов // Шестая национальная конференция по искусственному интеллекту с международным участием КИИ 98: сб. научн. тр. – Пушкино, 1998. – С. 510 – 517.
232. Kragelund L.V. Solving a Timetabling Problem Using Hybrid Genetic Algorithms // Software–Practice and Experience. – 1997.– V. 1.– 27(10).– P. 1121 – 1134.
233. Толковый словарь по искусственному интеллекту / Авторы–составители А.Н. Аверкин, М.Г. Гаазе–Рапопорт, Д.А. Поспелов. – М.: Радио и связь, 1992. – 256 с.
234. Колесников А.В. Концептуальная модель – инструмент познания внешнего мира в системах искусственного интеллекта // Человек техногенной цивилизации на рубеже двух тысячелетий: Монография в соавторстве с Ивановым В.Е., Каракозовой Э.В. и др.– Раздел 3.– Глава 2. – Калининград: КГТУ, Калининградское отделение МАИ, 2000. – С. 152 – 175.
235. Аверкин А.Н., Блишун А.Ф., Гаврилова Т.А., Осипов Г.С. Приобретение и формализация знаний // Искусственный интеллект. В 3 кн. Кн. 2. Модели методы: Справочник / Под ред. Д.А. Поспелова. – М.: Радио связь, 1990.– С. 65 – 76.
236. Лелюк В.А. Концептуальное проектирование систем с базами знаний. – Харьков: Основа, 1990. – 144 с.
237. Lapin K. Configuration of Technical Drawings: Conceptual Model of the Application Domain // The 4 International Conference “Mathematical Modelling and Analysis MMA 99”: proceedings. – Vilnius, 1999. – P. 54.
238. Ribikauskas A., Caplinskas A., and Vasilecas O. Conceptual Models of the Quality Control Process // The 4 Intern. Conf. “Mathematical Modelling and Analysis MMA 99”: proceedings. – Vilnius, 1999. – P. 63.

239. Уемов А.И. Вещи, свойства, отношения. – М.: Институт философии АН СССР, 1963. – 184 с.
240. Уемов А.И. Системный подход и общая теория систем. – М. Мысль, 1978. – 272 с.
241. Берзтис А.Т. Структуры данных. – М.: Статистика, 1974. – 408 с.
242. Осипов Г.С. Приобретение знаний интеллектуальными системами: Основы теории и технологии. – М.: Наука, 1997. – 112 с.
243. Кириков И.А., Колесников А.В., Пономарев В.Ф. Об одном подходе в семиотическом моделировании состояния транспортных систем // Вопросы кибернетики.– Вып. 68.– М.: Научный Совет по комплексной проблеме «Кибернетика» при Президиуме АН СССР, 1980. – С. 109 – 130.
244. Кудаков А.В., Речинский А.В., Щукин Д.В., Яшин А.М. Математические основы систем искусственного интеллекта: Учебное пособие. Ч.1. Экспертные системы. – СПб.: СПбГТУ, 1998. – 104 с.
245. Лорьер Ж.–Л. Системы искусственного интеллекта. – М.: Мир, 1991. – 568 с.
246. Кириков И.А., Колесников А.В. Языки описания схем и сценариев для представления знаний в АСУ транспортными узлами // 9 всесоюзный симпозиум по кибернетике. Т.1. «Представление знаний»: сб. научн. тр. – М.: Научный Совет по комплексной проблеме «Кибернетика» при Президиуме АН СССР, 1981.– С. 114 – 116.
247. Колесников А.В. Мультиязыковый подход к представлению знаний в диалоговых системах оперативного управления транспортными узлами // Сб. докладов Республиканский научно–технический семинар «Проблемно–ориентированные диалоговые комплексы»: сб. докл. – Кишинев: Молдавское региональное отделения Научного Совета по проблеме «Искусственный интеллект» Комитета по системному анализу при Президиуме АН СССР, 1983. – С. 82 – 84.
248. Кузин С.В. Методология, метод, теория с позиций криминалистики // Правоведение.– 2001.– №2.
249. Колесников А.В. Методология и технология разработки функциональных гибридных интеллектуальных систем // Интегрированные модели и мягкие вычисления в искусственном интеллекте. 2-й Международный научно–практический семинар: сб. научн. тр.– М.: Физматлит, 2003. – С.161– 167.
250. Декарт Р. Избранные произведения. – М.: Госполитиздат, 1950. – 712 с.
251. Поспелов Г.С., Ириков В.А. Программно–целевое планирование и управление (введение). – М.: Советское радио, 1976. – 440 с.
252. Stein B., and Curatolo D. Selection of Numerical Methods in Specific Simulation Applications // The 11 International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems IEA–98–AIE: proceedings.– V. 1.– Benicassim. Spain, 1998. – P. 918 – 927.
253. К. Бюрер. От ремесла к науке: поиск основных принципов разработки программного обеспечения». – <http://www.interface.ru>, 2006.
254. Д.А. Поспелов. Становление информатики в России. – <http://newasp.omskreg.ru/intellect/f30.htm>, 2006.

255. Глушков В.М., Матвеев М. Т., Михалевич М., Карнаух В. Т. Общие принципы создания республиканской автоматизированной системы управления народным хозяйством УССР (РАСУ). – Киев.: Изд-во ВЦ Госплана, 1973.
256. Sun R., and Bookman L. How Do Symbols and Networks Fit Together // AI Magazine. – 1995. – V.14. – № 2. – P. 20 – 23.
257. Козлов В.Н. Математика и информатика: Учебное пособие. – СПб.: Изд-во СПбГТУ, 2001. – 266 с.
258. Колесников А.В., Клачек П.М. Интеллектуальная система поддержки принятия решений по возделыванию сельскохозяйственных культур // Международная конференция по мягким вычислениям и измерениям (SCM 98): сб. докл.– Т. 2. – СПб., 1998. – С. 227 – 230.
259. Бондаренко Н.Ф. и др. Моделирование продуктивности агроэкосистем. – Л.: Гидрометеиздат, 1982. – 264 с.
260. Полевой А.Н. Теория и расчет продуктивности сельскохозяйственных культур. – Л.: Гидрометеиздат, 1983. – 175 с.
261. Гильманов Т.Г. Математическое моделирование биогеохимических циклов в травяных экосистемах. – М.: МГУ, 1978. – 272 с.
262. Полуэктов Р.А., Пых Ю.А., Швытов И.А. Динамические модели экологических систем.– Л.: Гидрометеиздат, 1980.– 88 с.
263. Полуэктов Р.А. Динамические модели агроэкосистем. – Л.: Гидрометеиздат, 1991.– 311 с.

Научное издание

Колесников Александр Васильевич
Кириков Игорь Александрович

**Методология и технология решения сложных задач
методами функциональных гибридных
интеллектуальных систем**

Редактор Г.А. Смирнова

Оригинал–макет подготовлен в ИПИ РАН

Подписано в печать 15.03.06

Тираж 100 экз. Заказ №

Издательство ИПИ РАН
119333, Москва, ул. Вавилова, д.44, корп.2
Лицензия ИД № 06392 от 05.12.2001.

Для заметок

© А.В. Колесников, И.А. Кириков, 2007
ISBN 978-5-902030-55-3 © Институт проблем информатики РАН, 2007