

генерации одного или нескольких вариантов разбора (или же в фиксации неграмматичности предложения). В нашем случае ЛП должен упорядочить допустимые цепочки и выбрать один вариант фразы (который и предъявляется как результат распознавания) из нескольких цепочек-кандидатов, любая из которых может быть неграмматична. Один из возможных механизмов решения этой проблемы — "островной анализ". Под "островом" понимается грамматически содержательный фрагмент предложения (именная группа, глагол-предикат с частично заполненными валентностями и др.).

Если цепочка почти полностью покрывается островами, а набор островов и "разрывов" удовлетворяет определенным эвристическим критериям, то такую цепочку можно признать допустимой и приписать ей соответствующий грамматический штраф.

Решение многокритериальной задачи выбора единственного варианта распознавания осуществляется на основе этих штрафов, а также следующей информации: штрафов, поступивших от ФР; информации о результатах локальных проверок грамматической структуры и семантической сочетаемости слов; статистических данных о лексической сочетаемости слов и частотности отдельных слов; статистических данных о частотности синтаксической структуры островов, предложений или фразы в целом.

Описанные разработки ведутся в рамках договора о сотрудничестве между факультетом вычислительной математики и кибернетики МГУ им. М.В. Ломоносова и фирмой Accent, Inc. (США). В работе принимают участие более десяти специалистов (инженеры знаний, эксперты-филологи, программисты) из России (МГУ, Институт Востоковедения РАН), США и Великобритании.

Реализация ЛП и необходимых инструментальных средств (формирование, тестирование и сопровождение лингвистической базы знаний; планирование и анализ результатов экспериментов) выполнена сотрудниками факультета ВМиК МГУ В.Г. Абрамовым, И.Г. Головиным и В.Н. Пильшиным. Язык реализации — Visual C++, версия 2.0, операционная среда — Windows NT, версия 3.5.

## О разработке процедур автоматического решения задач

А.С. Подколзин

Разработана компьютерная система, имитирующая поведение человека при решении математических задач. Обучение системы проводилось для таких разделов, как элементарная алгебра, тригонометрия и дифференциальное исчисление: при этом был достигнут достаточно высокий процент решаемых задач средней сложности. В основе решателя лежит новый принцип организации процессов решения задач, обеспечивающий практически квази-градиентное его функционирование и позволяющий преодолеть эффект перебора.

Разработка компьютерных решателей задач является важным направлением в математической кибернетике и теории интеллектуальных систем. Такие решатели составляют основу самых различных экспертных и интеллектуальных систем (см., например, [7, 11, 12, 13]), используемых для автоматизации инженерных расчетов в технике, управления сложными технологическими и динамическими процессами, обработки информации при проведении научных исследований, а также систем компьютерного обучения. Создание решателей стимулировало проведение как исследований в конкретных предметных областях (компьютерная алгебра, вычислительная геометрия, дискретная оптимизация и др.) ориентированных на развитие используемого при решении задач математического аппарата, так и исследований математических моделей процесса решения задач в целом [1, 3, 5, 9].

Можно выделить два основных подхода, использовавшихся при разработке систем автоматического решения задач. Первый из этих подходов, связанный с созданием экспертных систем в конкретных предметных областях, основан на применении древовидной классификации задач из рассматриваемой области и накоплении библиотеки алгоритмических процедур, решающих задачи различных типов согласно их классификации. Каждая такая процедура определяет целенаправленный процесс преобразования входных данных ("постановка задачи") при помощи некоторого списка извлеченных из теории рассматриваемой предметной области утверждений ("знаний"), дающих возможность видоизменить описание задачи с уменьшением ее сложности характеристик. Пользователь, работающий с экспертной системой такого рода, по существу находится в программной среде, из которой извлекает те или иные процедуры и выполняет с их помощью очередную шаг обработки информации. Процесс обработки информации при этом имеет ярко выраженный диалоговый характер, и система играет роль



манипулятора, применяемого для выполнения трудоемких операций. Примерами экспертных систем, основанных на указанном принципе, могут служить различные системы компьютерной алгебры [2, 4]. В этих системах реализованы мощные пакеты процедур обработки полиномиальных данных, дифференцирования и символического интегрирования, символического решения некоторых классов дифференциальных уравнений и т.п. Вместе с тем, при развитии методов компьютерного решения алгебраических задач обнаружилось, что для некоторых их классов не удается предложить эффективной процедуры, основанной на упаковке в одну "блок-схемную" типа конструкцию компактного подмножества теоретических знаний. Такими оказались, например, задачи на упрощение алгебраических выражений, решение систем уравнений и неравенств без существенного ограничения характера для элементарной алгебры их разнообразия, доказательство неравенств и т.п. Каждый из этих классов задач требует для своего решения весьма обширного многообразия знаний, не укладывающихся в простые "блок-схемные" конструкции. Более того, даже в тех случаях, когда теоретические исследования приводят к созданию одного общего алгоритма для обширного класса задач, например, в формальном интегрировании трудоемкость вычислений по этому алгоритму оказывается, как правило, существенно большей, чем при эвристическом использовании многообразия "классических" частных методов. В результате возникающих здесь трудностей даже такие мощные системы компьютерной алгебры, как "DERIVE", "MATHEMATICA", "AXIOM" оказываются не в состоянии решить подавляющего большинства задач по элементарной алгебре и математическому анализу, содержащихся в стандартных сборниках задач.

Второй подход к созданию процедур автоматического решения задач, тесно примыкающий к исследованиям по математической логике [6, 8, 9, 10], основан на применении баз знаний декларативного или продукционного характера, представленных на формальном логическом языке. Такая база знаний используется для сведения текущей решаемой задачи к одной или нескольким вспомогательным задачам либо для извлечения следствий из некоторых утверждений, встречающихся в задаче. Характерной особенностью возникающих здесь процедур является рассмотрение так называемых И-ИЛИ-деревьев задач, а также древовидных процессов логического вывода в рамках одной задачи. Вследствие экспоненциального роста сложности указанных древовидных конструкций практическое применение процедур такого рода наталкивается на существенные трудности, известные под названием проблемы перебора. Вопросам ограничения перебора, возникающего при решении задач, и, в особенности перебора при автоматическом доказательстве теорем, посвящено большое количество работ по математической логике и дискретной математике. В этих работах развивается техника эквивалентных преобразований логических выводов и приведения их к определенному стандартному виду, что позволяет при поиске необходимого вывода определенным образом ограничивать пространство перебора. Достигнутые здесь результаты, однако, имеют скорее теоретическое, чем практическое значение, не устраняя ощутимого уже для сравнительно несложных задач экспоненциального нарастания трудоемкости. На основе

продукционно-логического подхода создан ряд программ автоматического решения задач, в том числе по элементарной алгебре [7]. В этих программах для выбора альтернатив в процессе решения используются общие, реализованные главным образом вне базы продукций механизмы, как правило — достаточно примитивные оценочные функции. Отсутствие заложенной в продукциях информации, отражающей адекватным образом тот весьма обширный опыт, который накапливается учащимся при освоении задач соответствующих разделов и необходим для эффективного их решения, приводит к сохранению переборного в целом характера функционирования указанных программ и накладывает принципиальные ограничения на их эффективность уже на уровне задач средней вычислительной сложности.

Сопоставляя достоинства и недостатки изложенных двух подходов, можно отметить, что первый из них использующий базы процедур библиотечного типа, реализует целенаправленные в некотором смысле градиентные процессы решения задач, но для каждой процедуры — из своего узкого класса задач и на основе узкой выборки знаний, второй же, наоборот, реализует плохо ориентированные, переборного типа процессы, но на основе весьма обширного многообразия привлекаемых для этого знаний. Вместе с тем, при решении задач человеком типичной является ситуация, когда преобразование имеют градиентный или квазиградиентный характер, с локальным анализом небольшого числа вариантов перед выбором очередного шага и с крайне редкими переключениями нелокального характера, но используемые для решения задачи знания чрезвычайно разнообразны, а выборка их из полной базы знаний определяется лишь в процессе решения и допускает значительные вариации от задачи к задаче. Такая ситуация явно указывает на необходимость разработки нового подхода к созданию процедур автоматического решения задач, обеспечивающего целенаправленное, квазиградиентное решение задачи при использовании полномасштабной базы знаний. В основе этого подхода лежит база алгоритмических процедур, осуществляющих усмотрение в задаче определенных ситуаций и выполнение в этих ситуациях преобразований градиентного (в весьма широком смысле) или квазиградиентного характера. Каждая такая алгоритмическая процедура, которую естественно назвать приемом решения задач, осуществляет следующие действия:

- 1) Усмотрение в текущей ситуации возможности применить некоторое ее допустимое (в смысле логической корректности) преобразование.
- 2) Проверка условий, определяющих эвристическую целесообразность ("квазиградиентность") применения данного преобразования.
- 3) Реализация преобразования.

В основе применяемого приема преобразования лежит некоторая теорема, относящаяся к рассматриваемой предметной области, однако представление этой теоремы в части 3) приема составляет, как правило, лишь незначительную долю всей его конструкции, преимущественно алгоритмической и требующей для своего представления достаточно развитых алгоритмических языков. В широком смысле прием можно рассматривать как процедуру, осуществляющую настолько сложное преобразование ситуации, что для ее записи приходится использовать не логические, а алго-



ритмические средства самого общего характера. Если в производственных системах управление срабатыванием производств осуществлялось главным образом за счет вынесенной во внешние блоки оценочной, то в системах, основанных на базе приема, вся управляющая логика заключена в самих приемах, что позволяет осуществлять принятие решений на основе гораздо более детального разбора конкретной ситуации. Решающее правило, реализованное в части 2 приема, представляет собой по существу небольшую базу знаний, хранящую описание ряда специфических ситуаций, в каждой из которых принятие решения происходит практически однозначно и почти не требует привлечения каких-либо статических соображений, характерных для теории распознавания образов. Накопление таких баз знаний в приемах осуществляется в процессе обучения системы и является главным источником той квазиградиентности, о которой говорилось выше. При этом сама база приемов оказывается не столько базой теоретических знаний из предметной области, сколько базой стандартных ситуаций, складывающихся при решении задач в этой области. Необходимость в использовании базы знаний теоретического характера, опирающейся на логическое представление информации, возникает лишь в тех случаях, когда имеющихся приемов ("автоматики" решателя) не хватает для решения. В этих случаях инициируются механизмы порождения новых приемов, ориентированных на наблюдаемую в задаче ситуацию, и таким образом происходит пополнение базы приемов. Анализ многообразия приемов, накопленных при обучении, в рамках излагаемого подхода, системы решению задач по элементарной алгебре и дифференциальному исчислению, позволяет высказать предположение о том, что в основе механизма порождения новых приемов лежит один весьма общий принцип. Этот принцип заключается в рассмотрении двух различных приемов, первый из которых осуществляет преобразование исходной ситуации  $A$  "против градиента" в некоторую ситуацию  $B$ , а второй выполняет "спуск по градиенту" ситуации  $B$  в ситуацию  $C$ . Если затем осуществить "спуск по градиенту" ситуаций  $A, C$  к тупиковым ситуациям  $A', C'$ , используя все имеющиеся в распоряжении "градиентные средства" базы приемов и при этом окажется  $A' \neq C'$ , то преобразование  $A$  в  $C'$  (либо обратное —  $C'$  в  $A'$ ), применяемое с учетом рассматриваемой целевой установки, порождает принципиально новый прием, не сводящийся по своей действительности к совокупности имевшихся ранее приемов. Этот принцип, который естественно назвать "принципом перевала", несмотря на свою простоту, позволяет проследить источники возникновения большинства приемов упомянутой системы решения математических задач и создает предпосылки для последующей автоматизации процесса обучения.

Процедуры решения задач представляются при описываемом подходе в виде "однородного" многообразия независимым образом формулируемых приемов, используемого как одно целое при решении задач любых указанных выше типов. На каждом шаге в процессе решения задачи возникает некоторое множество  $M$  допустимых вариантов применения приемов, из которого необходимо осуществлять выбор одного из таких вариантов. Этот выбор может происходить, например, путем сопоставления каждому элементу множества  $M$  числовой оценки и выделения произвольного элемента

с минимальной оценкой. Одним из основных этапов процесса обучения решателя является такая регулировка связанной с оценками управляющей логики приемов, чтобы на каждом шаге делался наиболее естественный с точки зрения человека выбор.

Согласно обычной схеме организации многообразия приемов (или производств) в базу знаний, они объединяются в древовидную процедуру  $D$ , узлы которой определяют выделение тех или иных элементов задачи и проверку связанных с ними условий, а концевые вершины соответствуют различным приемам. При перемещении от корня процедуры  $D$  к концевой вершине происходит выделение в задаче ряда объектов, определяющих конкретный вариант применения приема, и проверка условий его применимости. Поиск приема осуществляется здесь в процессе перебора всевозможных допустимых (в смысле истинности проверяемых условий) путей, ведущих от корня к концевым вершинам, и его можно рассматривать как сканирование многообразия приемов. Недостатком такой схемы является то, что для различных приемов приходится осуществлять независимые просмотры задач. Чтобы сделать один просмотр задачи общим для целой группы приемов, естественно применить другую схему, при которой поиск приема основан на сканировании задачи. Оказалось, что для подавляющего большинства приемов локализация в задаче ситуаций, определяющих варианты их применения, начинается с обнаружения вхождения в текст задачи некоторого фиксированного для данного приема понятия (логического символа). Это сделало целесообразным объединение всех приемов соответствующих одному и тому же логическому символу  $\varphi$ , в одну древовидную процедуру  $D_\varphi$  указанного выше типа, а приемов, применение которых не связано с выделением в задаче вхождения какого-либо конкретного логического символа — в процедуры "общих" приемов решения задач соответствующего типа. При разработке системы решения математических задач были выделены 4 таких процедуры — для задач на доказательство, описание, преобразование и исследование; обозначаем их далее  $D_1 - D_4$ . В результате вся база знаний оказывается организованной по принципу энциклопедии, где за каждым понятием (логическим символом  $\varphi$  либо типом задачи) закреплена соответствующая ему информация — процедура  $D_\varphi$  либо одна из процедур  $D_1 - D_4$ . Для поиска приема предпринимается последовательный просмотр всех вхождений логических символов в текст задачи (сканирование задачи) и обращение для каждого такого вхождения символа  $\varphi$  к процедуре  $D_\varphi$ . Перед началом очередного цикла сканирования реализуется соответствующая типу задачи процедура  $D_i$ ;  $i \in \{1, 2, 3, 4\}$ . Предлагаемая общая схема ориентирована главным образом на разработку "энциклопедически" построенных процедур, охватывающих сразу многие предметные области, так как при переходе к новой предметной области происходит формирование группы приемов, связанных с новыми понятиями, лишь в незначительной степени затрагивающее группы приемов, соответствующих ранее рассмотренным понятиям, и практически не увеличивающее время их поиска. Созданная на основе этой схемы процедура автоматического решения задач элементарной алгебре (упрощение выражений, решение уравнений, равенств, систем уравнений и неравенств, доказательство тождеств и не-



равенств, причем в задачах разрешается использовать полный перечень элементарных функций, логические связки и кванторы) и дифференциальному исчислению (вычисление пределов и производных, поиск глобальных и локальных экстремумов, нахождение интервалов монотонности, оценка числа корней и доказательство неравенств с помощью производной) демонстрирует высокий процент решаемых задач (порядка 90% задач из указанных разделов для распространенных задачников по элементарной алгебре и математическому анализу) и достаточно хорошую имитацию поведения человека при решении задач. Разработана технология, позволяющая сравнительно несложным образом продолжать обучение этой процедуры как для постепенного повышения уровня сложности решаемых задач из "старых" предметных областей так и для расширения возможности системы на новые предметные области.

Отметим ряд возможных направлений дальнейшего развития системы и создание систем другого профиля, опирающихся на указанную схему организации процесса решения задач. Накопленный при обучении системы опыт извлечения приемов решения задач, содержащихся неявным образом в сборниках задач, создает реальные предпосылки для освоения основных разделов элементарной математики (в том числе элементарной геометрии), математического анализа и других областей математики, а также различных разделов естественных и технических наук, опирающихся на математический аппарат. Такую развитую систему решения задач можно было бы использовать как в составе интеллектуальных обучающих систем так и для автоматизации разработки "символьных" математических моделей и основанных на них пакетов программ в различных инженерных проектах, автоматизации конструирования и т.п. Другим направлением работ по применению описанной схемы сканирования задачи является создание систем, воспринимающих тексты естественного языка и изображения и транслирующих содержащуюся в них информацию во внутреннее логическое представление. Специфика функционирования таких систем состоит в предварительной обработке информации, осуществляемой процедурами периферического характера, результатом которой становится промежуточное сетевое представление воспринимаемых объектов. В случае текстов здесь предпринимается синтаксический разбор отдельных слов языка, состоящий в выделении основных компонент (корень, приставка и т.п.) и перекодировке этих компонент посредством логических символов внутреннего представления. При обработке изображений выделяются простейшие типовые геометрические элементы (углы, пересечение линий, закругление и т.п.), также обозначаемые посредством логических символов. Формируемое сетевое представление имеет, вообще говоря, лишь частичный характер и содержит ссылки на различные точки исходной полной структуры данных (текста либо изображения), из окрестности которых при необходимости извлекается дополнительная информация. Обработка полученного таким образом сетевого представления объекта осуществляется в процессе сканирования его аналогичного описанному выше сканированию задач. При сканировании просматриваются всевозможные вхождения логических символов, связанных с элементами сети, и для каждого такого вхождения символа  $\varphi$  пред-

принимается обращение к программе символа  $\varphi$ , объединяющей некоторое многообразие анализа контекста данного символа. Эти приемы, в зависимости от ситуации, формируют новые фрагменты логического представления объекта, корректируют ранее сформулированную часть логического представления, запрашивают необходимую для пополнения сетевого представления новую информацию об объекте и изменяют это представление, иницируют процессы решения задач, необходимые для осуществления семантической коррекции представления объекта, и т.п. Специальный алгоритмический язык, разработанный при создании системы автоматического решения задач, позволяет реализовать совместное функционирование базы приемов решения задач и вспомогательных баз приемов указанного типа, ориентированных на сканирование сетевых представлений объектов, что создает предпосылки для включения процедур восприятия текстов естественного языка и изображений в интерфейс системы решения задач.

В заключение отметим еще одно важное направление исследований, связанное с разработкой интеллектуальных систем, ориентированных на управление сложными процессами. Здесь также выделяется ядро системы, осуществляющее решение потока возникающих в процессе управления задач, и ряд блоков периферической обработки информации, в первую очередь следующие два блока:

- 1) Архив данных управляющей системы накапливающий информацию о предистории управляемого объекта и его текущем состоянии. Этот архив представляет собой систему информационного поиска, обрабатывающую поступающие на нее запросы и осуществляющую динамическое слежение по определенной извне целевой установке.
- 2) Блок управляющей автоматике, осуществляющий сканирование концентрированным образом отображающего текущую ситуацию сетевого представления и иницирующий как внешние реакции системы, так и внутренние процессы формирования планов управления. Этот блок, по существу является "базой рефлексов", заблаговременно накапливаемых решателем задач на основе анализа возможных вариантов развития событий и срабатывающих в зависимости от их фактической реализации.

Решатель задач, входящий в состав системы управления, получает исходную информацию в виде смешанного логико-сетевого представления, описывающего существенные компоненты текущей ситуации и формулирующего целеуправление. Дальнейшая обработка этого представления осуществляется в процессе сканирования его и срабатывания закрепленных за соответствующими логическими символами приемов, выполняющих пополнение описания ситуации на основе логического вывода, прогнозирование событий, формирование элементов плана управления и оптимизацию управляющих параметров, обращение к решению вспомогательных задач логического либо вычислительного характера и иницирующих рассмотрение альтернатив. Для накопления приемов решения задач в системе управления необходимо применение имитационных моделей управляемой системы, играющих примерно ту же роль, что и сборники задач при обучении системы решению задач теоретического характера.



В заключение автор выражает глубокую благодарность академику АТН РФ профессору В.Б. Кудрявцеву за создание исключительно благоприятных условий для работы над системой решения задач и постоянную поддержку.

## СПИСОК ЛИТЕРАТУРЫ

1. Бенерджи Р. Теория решения задач. — М.: Мир, 1972.
2. Бухбергер Б., Калме Ж., Калтофен Э. и др. Компьютерная алгебра: символьные и алгебраические вычисления. — М.: Мир, 1986.
3. Васильев С.Н. К автоматизации вывода теорем с аналогами функций Ляпунова и морфизмов // Computers and Artificial Intelligence. — 1982. — V. 1, № 6. — P. 517-538.
4. Девенпорт Дж. и др. Компьютерная алгебра: системы и алгоритмы вычислений. — М.: Мир, 1991.
5. Ершов Ю.Л.  $\lambda$ -определимость в допустимых множествах // ДАН СССР. — 1985. — Т. 285, № 4. — С. 792-793.
6. Kowalski R.A. Logic for Problem Solving. — Elsevier-North. Holland, New York, 1979.
7. Лорьер Ж.-Л. Системы искусственного интеллекта. — М.: Мир, 1991.
8. Тейз А., Грибамон П., Луи Ж. и др. Логический подход к искусственному интеллекту: от классической логики к логическому программированию. — М.: Мир, 1990.
9. Тыугу Э.Х. Концептуальное программирование. — М.: Наука, 1984.
10. Чень Ч., Ли Р. Математическая логика и автоматическое доказательство теорем. — М.: Наука, 1983.
11. Автоматизированное проектирование систем управления. — М.: "Машиностроение", 1989.
12. Диалоговые системы схмотехнического проектирования. — М.: "Радио и связь", 1988.
13. Искусственный интеллект (справочник). — М.: "Радио и связь", 1990. — ТТ. 1-3.
14. Подколзин А.С. Об организации баз знаний, ориентированных на автоматическое решение задач // Дискретная математика. — 1991. — Т. 3, Вып. 3. — С. 13-30.
15. Подколзин А.С. Компьютерный решатель математических задач // ДАН РФ. — 1994. — Т. 335, № 4.

## Система МЕДИС — новая информационная технология в организации лечебного процесса

С.Р. Родин

В работе описывается медицинская интегрированная система МЕДИС, предназначенная для комплексной автоматизации отечественных больниц и госпиталей на основе локальных вычислительных сетей ПЭВМ с использованием новой информационной технологии. Система охватывает все службы медицинского учреждения, непосредственно связанные с организацией лечебно-восстановительного процесса. Система обеспечивает автоматизированное создание и ведение формализованной истории болезни пациента врачами лечебных и диагностических подразделений. В системе использован оригинальный подход к автоматизации проведения диагностических обследований на основе построения целого семейства специализированных сложноорганизованных адаптивных схем диалога, логика работы которых определяется специальными алгоритмами, управляемыми вводимыми данными. В перспективе система будет состоять из семейства совместимых подсистем, ориентированных на использование в лечебных учреждениях различного типа, что позволит использовать единую историю болезни пациента и на этой основе интегрировать лечебные учреждения различного типа в единую сеть.

### 1. СОСТОЯНИЕ ПРОБЛЕМЫ

В последнее время как в нашей стране, так и за рубежом современная медицинская техника активно включается в лечебно-диагностический процесс. Появление и внедрение в практику лечения таких диагностических методов, как компьютерная томография, радиоизотопные методы лечения, глубокое исследование коронарных процессов и т.п., привело к тому, что рабочее место специалиста-диагноста оснащается мощным современным оборудованием, использование которого немислимо без компьютерной обработки. Однако, изучение зарубежного опыта в создании систем подобного класса показывает, что непосредственное использование их в условиях нашей страны практически невозможно, поскольку подход к организации лечебного процесса в нашей стране существенно отличается от зарубежного. У нас главный участник лечебного процесса — лечащий врач, остался, во-существу, на том же уровне оснащенности, что и несколько десятилетий назад. В его арсенал входят знания, опыт, стетоскоп и ручка, которой, часто в ущерб качеству лечения, приходится заполнять огромное