УДК 004.891

СПОСОБЫ ВЫБОРА ПРАВИЛ ПРИ ОБРАТНОМ ВЫВОДЕ В СТАТИЧЕСКИХ ЭКСПЕРТНЫХ СИСТЕМАХ

А.М. Юрин, М.П. Денисов

Аннотация

В статье рассмотрена задача оптимизации процесса поиска решений статической экспертной системы. Для ее решения было проведено исследование этапа обратного вывода по выбору правила для исполнения. Приведен метод сбора статистики для анализа данного этапа. Представлены результаты сбора статистики в виде сравнительного анализа 4 способов выбора правил для баз знаний в 3 категориях решаемых задач. Выявлены наиболее эффективные способы в этих категориях. На их основе предложены вариант комбинированного способа и алгоритм обратного вывода, минимизирующий число правил, которые используются при решении рассмотренных категорий задач.

Ключевые слова: экспертные системы, базы знаний, подсистема поиска решений, обратный вывод, продукционное представление знаний.

Введение

Постоянное повышение уровня автоматизации в различных сферах человеческой деятельности требует развития методов решения задач, традиционно считающихся интеллектуальными. Такие задачи решаются с использование методов искусственного интеллекта (ИИ) [1–4], активно развивающегося последние годы. Одним из подходов решения подобных задач с применением ИИ является получение знаний у экспертов предметной области, их накопление, систематизация и дальнейшее использование. Такой подход выделяется в направление инженерии знаний [2, 3], тесно связанный с созданием экспертных систем (ЭС) [5, 6].

Отличительными чертами ЭС являются: наличие языка представления знаний [1, 2], схожего с естественным языком, доступным специалисту предметной области; наличие подсистемы поиска решений [1, 7], осуществляющей решение поставленной задачи с использованием накопленных знаний; возможность объяснения [3] действий системы и формирование последовательного протокола решения. Интеллектуальность процесса решения во многом зависит от базы знаний и подсистемы поиска решений. Для продукционной модели представления знаний [1, 2], являющейся наиболее простой для понимания экспертов, используются следующие стратегии поиска решений: прямой вывод, обратный вывод, а также их комбинации.

Прямой вывод работает по принципу цикла распознавание-действие [3] и включает в себя: сопоставление правил с рабочей памятью [3, 8] и формирование конфликтного набора [3, 8], разрешение конфликта [8] и активацию правила. При этом возникают вопросы быстродействия при формировании конфликтного набора (одним из решений вопроса является использование алгоритма RETE [9] и его разновидностей) и вопросы выбора правила при разрешении конфликта. Наилучшие методы разрешения конфликтов основаны на знаниях определенной задачи, однако существуют исследования по разработке стратегий разрешения конфликтов [8] для широкого круга предметных областей, которые используются во многих современных инструментальных средствах создания экспертных систем (ИЭС). Кроме того,

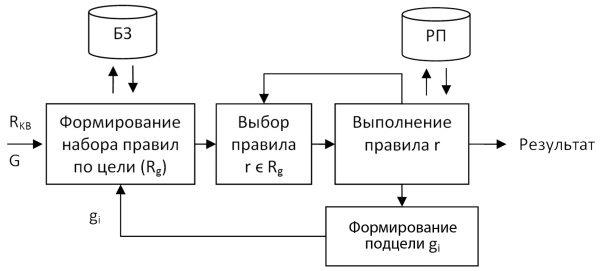

Рис. 1. Схема обратного вывода

активно развивается теория использования метаправил [5], позволяющая в алгоритмах поиска решений при выборе правила использовать метазнания о предметной области.

Для обратного вывода [3, 5] требуется задание искомой цели в явном виде. Процесс поиска решений в случае обратного вывода содержит: поиск правил, предположительно определяющих цель или подцель; выбор правила, которое будет рассмотрено на текущей итерации; проверка условий, исполнение выбранного правила и при необходимости поиск подцелей. Проблема повышения быстродействия поиска правил, определяющих подцель, решается путем построения индексов по базе знаний. Выбор правила во многом схож с проблемой разрешения конфликта, однако в большинстве систем с обратным выводом используются только следующие способы: выбор первого правила, определяющего подцель, и выбор по приоритетам.

В настоящей статье представлено исследование влияния различных способов выбора правил на ход решения и вид алгоритма поиска решений при обратном выводе. Исследование проводилось на базе системы ExPRO 4 [10] путем сбора статистики применения различных способов выбора правила и ее анализа.

1. Алгоритм обратного вывода в продукционной экспертной системе

Алгоритмы поиска решений с обратной стратегией вывода, реализованный в инструментальной экспертной системе (ИЭС) ExPRO и модифицированный для возможности реализации исследуемых способов выбора правил, представлен на рис. 1.

На вход алгоритма поиска решений поступает цель. Затем запускается итерационный алгоритм, состоящий из следующих этапов: формирование набора правил (R_g), определяющих заданную цель; выбор правила $r \in R_g$ для выполнения; выполнение r, где возможен рекурсивный вызов поиска подцелей.

Выполнение правила r происходит в 2 этапа: проверка условий правила, выполнение тела правила. Запуск поиска подцелей может быть на обоих этапах в случае, если система находит неопределенную переменную. Если условная часть правила оказалась ложной, то выбирается альтернативное правило из $R_g \backslash r$.

Формирование набора правил R_g осуществляется по синтаксическому принципу наличия переменной-цели в левой части выражения: <левый операнд><оператор присваивания><выражение>, где <левый операнд> ::= <имя объекта> | <имя переменной> | <имя свойства объекта>[1].

[1] Пример правила: ЕСЛИ глубинаОтверстия > 10 * диаметрОтверстия, материалПоверхности = «Углеродистая и легированная сталь» ТО материалСверла=[«Т5К12»,«ВК8»,«Т15К6»]. Данное правило определяет цель «материалСверла» и требует нахождения подцелей «глубинаОтверстия» и «диаметрОтверстия» при проверке условия.

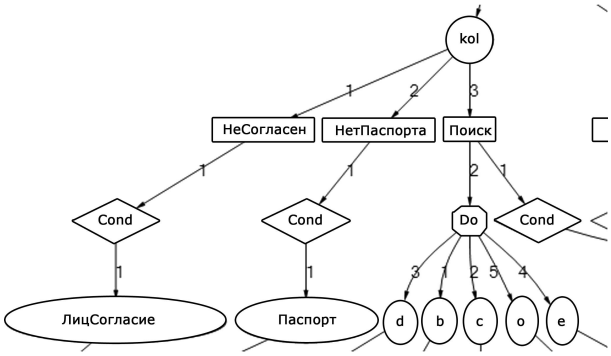

Рис. 2. Фрагмент дерева поиска решений

Табл. 1

Условия правил из фрагмента дерева поиска решений

Имя правила	Условия правила
НеСогласен	ЛицСогласие="Нет"
НетПаспорта	Паспорт="Нет"
Поиск	F="Поиск"

Определение способа выбора $r \in R_g$ является одной из актуальных проблем алгоритма обратного вывода. От него зависит эффективность и существование решения. Так, в случае примера на рис. 2 определение подцели kol можно достигнуть путем выполнения одного из трех правил, условия которых приведены в табл. 1. Для выполнения каждого из трех правил необходимо нахождение своих подцелей. Таким образом, от выбора правила, которое будет рассмотрено на текущей итерации, зависят общая последовательность и количество выполненных правил.

2. Статистика использования способов выбора правил

Для определения способа выбора правила из R_g при обратном выводе проведено исследование в рамках системы ExPRO. Собрана статистика в процессе поиска решений для различных способов выбора правил в рамках баз знаний, созданных студентами КФУ, КНИТУ-КАИ и КГАСУ.

Анализ проводился для следующих способов выбора правил:

1) ручная приоритезация правил (способ упорядочивания);

2) выбор случайного правила (способ случайного выбора);

3) выбор правила с наиболее сложным условием (способ специальных случаев);

4) выбор наименее и наиболее используемых правил (способ различий).

Для оценки решения выделены следующие формальные параметры:

1) процент решенных задач;

2) среднее выборочное нормированного числа правил, задействованных в решении: $rc_n = (rc - rc_{\min})/(rc_{\max} - rc_{\min})$, где rc – количество правил, рассмотренных в процессе текущего решения, $rc_{\min}$, $rc_{\max}$ – минимальное и максимальное числа правил, рассмотренных в решениях текущей задачи. Характеризует среднее отклонение отобранной группы решений от наилучшего решения данной задачи;

3) несмещенная выборочная дисперсия нормированного числа правил, задействованных в решении. Характеризует нестабильность работы способа в отобранной группе решений.

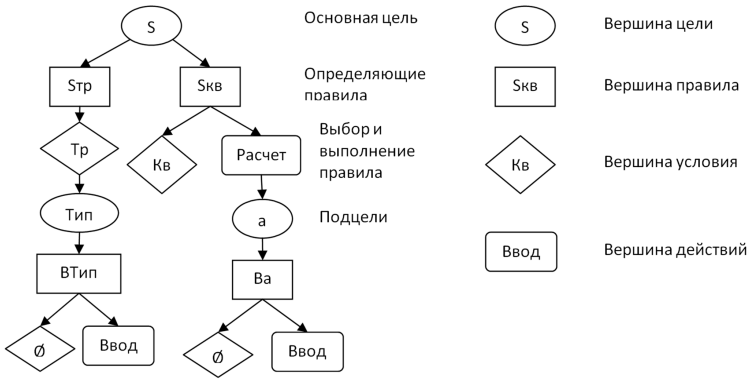

Рис. 3. Пример дерева поиска решений

Сбор статистики проходит по следующему методу:

1) в режиме решения задачи модуль сбора статистики протоколирует ход решения путем построения дерева поиска решений;

2) после окончания решения по дереву поиска решений строится свертка – ветвь дерева всех решений;

3) в случае существования дерева всех решений для текущих исходных и вводимых пользователем данных полученная ветвь добавляется к дереву; в случае отсутствия дерева оно создается;

4) изменения дерева всех решений и дерево поиска решений, ассоциированное с одной из его веток, сохраняются;

5) в случае успешной проверки дерева всех решений на полноту в таблицу статистики добавляется новая строка.

Дерево поиска решений (рис. 3) отражает процесс одного решения на одном наборе исходных/вводимых пользователем данных.

Данное дерево полностью отражает процесс поиска решений и обладает следующими свойствами:

1) каждое правило является дочерней вершиной вершины подцели, которую оно может определить;

2) каждое правило, которое было выбрано для рассмотрения, имеет дочернюю вершину условий;

3) каждое правило, которое было выбрано для рассмотрения и условие которого оказалось истинно, имеет дочернюю вершину действий;

4) вершины условий и действий могут иметь дочерние вершины подцелей, если при проверке условий или выполнении действий был необходим их поиск.

Дерево поиска решений в вершинах целей содержит также состояние рабочей памяти на момент постановки цели, что позволяет в дальнейшем проводить полный анализ процесса.

Для осуществления функций модуля сбора статистики, описанных выше, данное дерево может быть свернуто в 1 ветвь дерева всех решений (рис. 4), которое отражает все решения одной задачи одной базы знаний на одном наборе исходных/вводимых пользователем данных. Свертка дерева осуществляется путем обхода дерева в глубину и учета только вершин правил, имеющих дочернюю вершину условий.

Дерево всех решений позволяет анализировать конкретное решение в сравнении со всеми остальными решениями данной задачи. Для этого необходима проверка наличия всех возможных решений в данном дереве (полноты дерева всех решений):

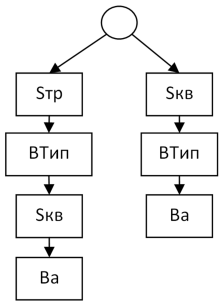

Рис. 4. Пример дерева всех решений

Предметные области	Примеры баз знаний по решаемым задачам	Количество БЗ	Тестов
Диагностика	Выбор диеты при заболеваниях желудка	5	15
	Диагностика заболевания зубов		
	Диагностика неисправностей автомобилей ВАЗ		
Подбор	Выбор авто	13	24
	Подбор квартиры		
	Биржа труда: подбор рабочего места		
Справочные	Справочная система риэлтора	4	7
	Справочный фармацевтический терминал		
	Итог:	22	46

Рис. 5. Анализируемые базы знаний по решаемым задачам

1) лист дерева всех решений является полным;

2) вершина дерева всех решений считается полной, когда все её дочерние вершины полны и их количество соответствует размеру набора R_g, содержащего правила каждой из них на момент их выбора;

3) дерево всех решений полное, когда все его вершины полны.

После построения полного дерева всех решений может быть сформировано табличное представление статистики со следующими полями:

1) номер записи;

2) лучшее решение (число задействованных правил);

3) худшее решение (число задействованных правил);

4) способ выбора 1 (число задействованных правил);

5) ...

6) способ выбора n (число задействованных правил);

7) категория базы знаний.

Базы знаний по предметным областям, для которых проводилось исследование, приведены на рис. 5

Фрагмент собранной статистики приведен на рис. 6.

На рис. 7 приведены данные статистики по процентам решений задач с использованием различных способов выбора правил. Ни один из способов не позволил решить все анализируемые задачи. Выбор случайного правила дал худший результат.

мин	макс	первое	случайное	длинное	частота	категория
145	261	257	154	257	257	выбор
3	23	13	5	13	13	выбор
4	17	12	4	12	12	выбор
14	19	15	19	15	15	выбор
3	10	3	9	3	4	диагностика
5	30	7	28	7	7	диагностика
...	...	...	...	...	...	...

Рис. 6. Фрагмент статистики числа правил решения для различных способов

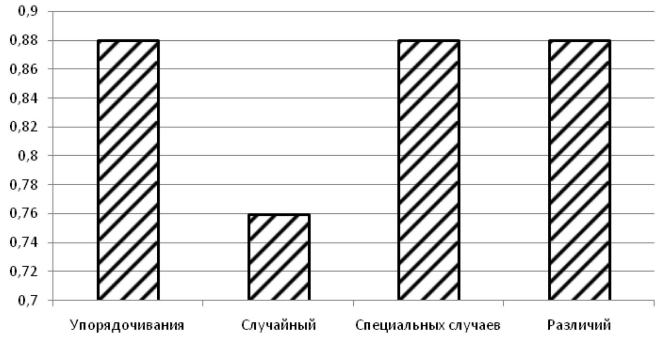

Рис. 7. Решено задач с использованием различных способов

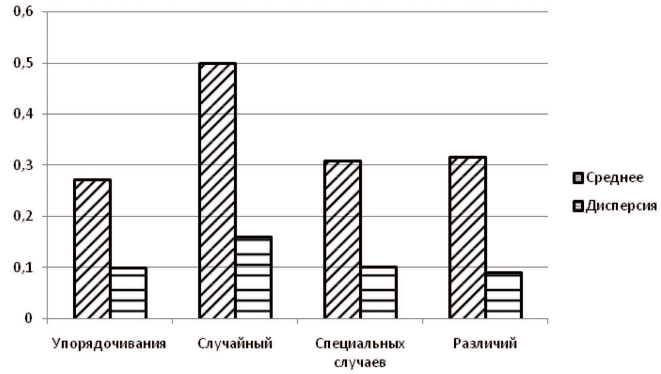

Рис. 8. Выборочное среднее и дисперсия нормированного числа задействованных в решении правил

На рис. 8 видно, что способ упорядочивания дает лучший результат для общего круга задач по критерию числа правил, задействованных в решении. Способ различий дает худший результат после способа случайного выбора, однако имеет наибольшую стабильность решения, что позволяет с большей уверенностью прогнозировать эффективность решения. Выбор случайного правила в общем случае дает наихудший результат и наибольшую нестабильность.

Относительно статистики выбора правил для различных категорий задач (рис. 9–11) видно:

1) для задач диагностики наиболее применимые способы: способ упорядочивания и способ специальных случаев;

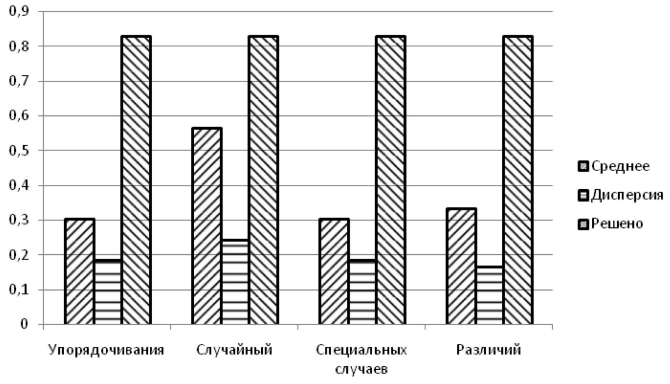

Рис. 9. Решение диагностических задач с использованием различных способов выбора правил

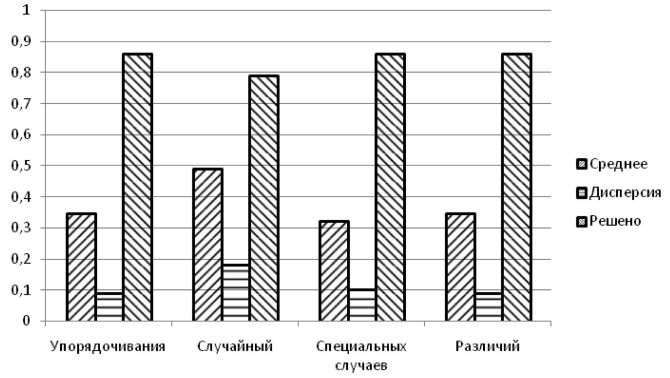

Рис. 10. Решение задач выбора с использованием различных способов выбора правил

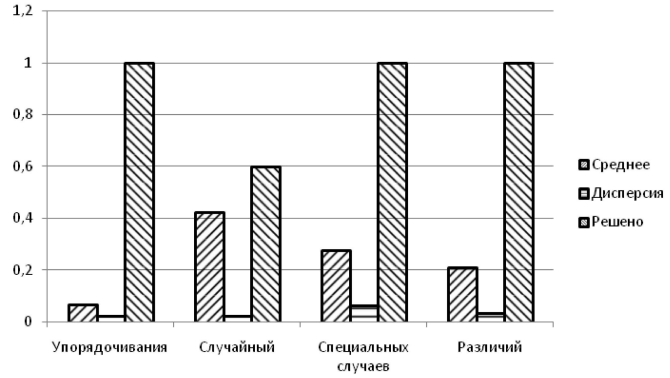

Рис. 11. Решение справочных задач с использованием различных способов выбора правил

2) для задач подбора наилучший результат дает использование способа специальных случаев (способы упорядочивания и различий дают одинаковый результат);

3) для справочных задач наилучший результат дает способ упорядочивания.

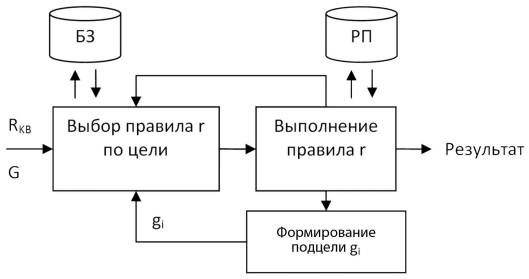

Рис. 12. Модифицированная схема обратного вывода

Задачи подбора имеют большее пространство состояний в сравнении с остальными рассмотренными категориями. Предположительно этим обусловливается сложность ручной расстановки приоритетов для использования способа упорядочивания. Справочные задачи, напротив, являются более формализованными и определяющими порядок исполнения, в связи с этим большинство конфликтных ситуаций разрешается на этапе проектирования базы знаний.

3. Алгоритм обратного вывода с комбинированным способом выбора правил

В общем случае для рассмотренных задач из исследованных способов наилучшим является способ упорядочивания. Его использование также позволяет отказаться от явного формирования набора R_g, что значительно экономит использование памяти и увеличивает производительность алгоритма. Однако для задач подбора наилучший результат получается при использовании способа специальных случаев.

В связи с этим предлагается для рассмотренных задач использовать способ упорядочивания с приоритетами и автоматический пересчет приоритетов с учетом сложности условий правил. На этапе проектирования базы знаний предоставляется возможность ручной расстановки приоритетов. Поиску решений предшествует подготовительный этап, включающий сортировку правил R_{KB} в следующей последовательности:

1) правила базы знаний R_{KB} сортируются в порядке убывания приоритетов. При этом правила с одинаковым приоритетом не меняют порядок относительно друг друга;

2) выбираются последовательности $p_s = \{p_i, \cdots, p_k\} \subset \{p_1, \cdots, p_i, \cdots p_k, \cdots p_n\}$, $i < k$, где $\{p_1, \cdots, p_n\}$ – невозрастающая последовательность приоритетов правил R_{RB}. $p_i = p_k, p_{i-1} > p_i, p_{k+1} < p_k$;

3) для каждой сформированной последовательности происходит пересчет приоритетов: для всех $p_j \in p_s$, $p_j = p_j + CondCount(r_j)$, где $CondCount$ – функция подсчета числа условий в правиле, r_j – правило, чей приоритет p_j рассматривается. При этом запоминается $p_m = \max\limits_{i<j<k} CondCount(r_j)$;

4) после пересчета каждой последовательности приоритетов происходит пересчет остальных приоритетов: для всех $j < i$, $p_j = p_j + p_m$;

5) после пересчета всех приоритетов вновь осуществляется сортировка R_{KB} в порядке убывания.

При такой сортировке правил R_{KB} нет необходимости выбора всего набора правил R_g, так как всегда выбирается его первое правило, поэтому схема поиска решений преобразуется в вид, представленный на рис. 12.

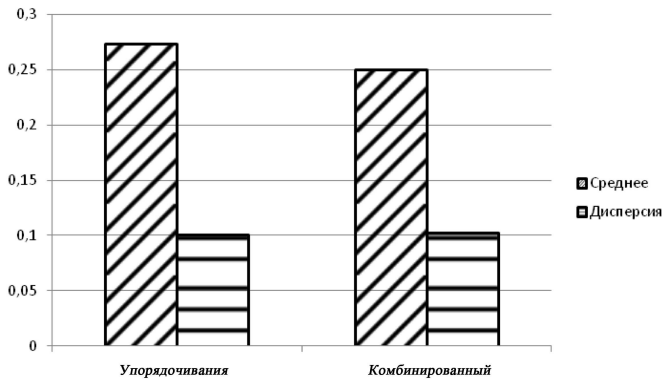

Рис. 13. Показатели решений при использовании комбинированного способа

На рис. 13 представлена гистограмма, демонстрирующая улучшение показателей решений при использовании предложенного способа (среднее выборочное нормированного числа правил, задействованных в решениях, уменьшается на 7.5%).

Заключение

В результате анализа обратного вывода получено, что наихудшее решение было получено при выборе случайного правила, что показывает необходимость оптимизации этапа выбора правил. При сравнении способов выбора правил были выявлены 2 наиболее эффективных способа в различных категориях задач: способ упорядочивания и способ специальных случаев. Их последовательное применение позволило уменьшить среднее число правил, задействованных в решении, на 7.5% по сравнению с ранее использовавшимися в системе методами, что означает нахождение системой лучшего решения в большем числе случаев. С целью увеличения быстродействия алгоритма поиска решений и оптимизации расхода памяти был разработан этап подготовки базы знаний с автоматическим пересчетом приоритетов таким образом, чтобы способ упорядочивания давал результат, аналогичный комбинации 2 выявленных способов. Это позволило отказаться от явного формирования набора R_g при обратном выводе и выбирать первое правило, определяющее подцель.

Предложенная методика исследования позволяет выявить способы выбора правил для прямого и обратного вывода, снижающие число правил, задействованных в решении. Это дает возможность реализации лучших вариантов прямого и обратного вывода, что позволяет сравнивать стратегии вывода и создавать алгоритм, формирующий наилучший процесс поиска решения для исследуемых предметных областей.

Summary

A.M. Yurin, M.P. Denisov. Methods of Rules Selection with Backward Chaining in Static Expert Systems.

The article discusses the problem of optimizing the solution search process in a static expert system. The research on the stage of backward chaining for rule selection and execution is conducted. A method of statistics collection for analysis of this stage is given. The results are presented in the form of a comparative analysis of the four methods of selecting rules

for knowledge bases in three categories of tasks. The most efficient methods are identified in each category. Based on these methods, a version of the mixed method and an algorithm of backward chaining minimizing the number of rules used in solving the tasks under study are proposed.

Keywords: expert system, knowledge base, inference engine, backward chaining, production knowledge representation.

Литература

1. *Новиков Ф.А.* Искусственный интеллект: представление знаний и методы поиска решений. – СПб.: Изд-во Политехн. ун-та, 2010. – 240 с.
2. *Гаврилова Т.А., Хорошевский В.Ф.* Базы знаний интеллектуальных систем. – СПб.: Питер, 2001. – 384 с.
3. *Попов Э.В., Фоминых И.Б., Кисель Е.Б., Шапот М.Д.* Статические и динамические экспертные системы. – М.: Финансы и статистика, 1996. – 320 с.
4. *Люгер Дж.Ф.* Искусственный интеллект: стратегии и методы решения сложных проблем. – М.: Изд. дом «Вильямс», 2003. – 864 с.
5. *Муромцев Д.И.* Введение в технологию экспертных систем. – СПб.: СПбГУ ИТМО, 2005. – 93 с.
6. *Джексон П.* Введение в экспертные системы. – М.: Изд. дом «Вильямс», 2001. – 624 с.
7. *Рассел С., Норвиг П.* Искусственный интеллект. Современный подход. – М.: Вильямс, 2006. – 1408 с.
8. *McDermott J., Forgy C.L.* Production system conflict resolution strategies. – URL: http://repository.cmu.edu/cgi/viewcontent.cgi?article=3143&context=compsci, свободный.
9. *Schneier B.* The Rete Matching Algorithm. – URL: http://drdobbs.com/architecture-and-design/184405218, свободный.
10. *Юрин А.М., Денисов М.П.* Разработка продукционной базы знаний с использованием инструментальной экспертной системы ExPRO 4 // Учен. зап. Ин-та социальных и гуманит. знаний. – 2014. – № 1(12), Ч. 2. – С. 299–305.

Поступила в редакцию
19.06.14

Юрин Арнольд Менделевич – кандидат технических наук, доцент кафедры системного анализа и информационных технологий, Казанский (Приволжский) федеральный университет, г. Казань, Россия.

E-mail: *yurin@fromru.com*

Денисов Максим Павлович – аспирант кафедры системного анализа и информационных технологий, Казанский (Приволжский) федеральный университет, г. Казань, Россия.

E-mail: *denmaxpav@gmail.com*