

БЛОК АНАЛИЗА ЗНАНИЙ  
В ИНСТРУМЕНТАЛЬНОЙ ЭКСПЕРТНОЙ СИСТЕМЕ ЭКСНА

М.В.Бушуев, В.Н.Ёлкина, Н.Г.Загоруйко, Е.Н.Шемякина

В в е д е н и е

Наиболее полные общие сведения об инструментальной экспертной системе ЭКСНА и подробное описание Блока Анализа Данных этой системы приведены в [1-3]. Данная публикация посвящена другому важному разделу системы ЭКСНА - Блоку Анализа Знаний.

Проблемно-ориентированные экспертные системы - это системы, уже заполненные знаниями и данными и готовые для работы в своей проблемной области. Инструментальная экспертная система, как следует из ее названия, это программный комплекс, с помощью которого специалист самостоятельно может в достаточно короткое время создать свою экспертную проблемно-ориентированную систему, не будучи при этом программистом. Инструментальная система предоставляет специалисту все необходимые для этого средства, взяв на себя всю программистскую нагрузку.

Системы должны быть достаточно "дружественными" - простыми в освоении и работе с ними, иметь удобные средства общения с пользователем. В этой связи хотелось бы привести лозунг, провозглашенный японцами для машин пятого поколения: "Вычислительная машина должна быть в обращении не сложнее, чем стиральная. Только тогда она станет таким же предметом домашнего обихода, как телефон".

Отметим также, что до сих пор в этой области нет единой терминологии. Так, наряду с термином "инструментальные системы" можно встретить, например, термины "оболочки" или "интенциональные базы знаний", а для проблемно-ориентированных систем - "экстенциональные базы знаний" и др.

Тем, кто хочет более подробно познакомиться с этой проблемой, мы можем порекомендовать, например, работы [4-8].

Полный состав системы ЭКСНА рассчитан на специалиста-исследователя, создающего с помощью инструментальных средств свою проблемно-ориентированную экспертную систему, наполняя ее конкретными знаниями.

Описание Блока Анализа Знаний системы ЭКСНА мы будем приводить в таком порядке: структура Базы Знаний и возможности Блока Анализа Знаний; логический вывод; наполнение Базы Знаний; проверка непротиворечивости знаний; проверка знаний на данных; получение новых знаний из данных.

#### Структура Базы Знаний. Возможности Блока Анализа Знаний.

База Знаний представляет собой совокупность правил (продукций), которыми могут быть описаны знания экспертов о любой предметной области.

Существуют различные способы представления знаний в памяти вычислительной машины: логические модели, семантические сети, фреймы, продукционные системы [1,4].

В нашей системе знания представляются в виде текстовых предложений (продукций) вида: "ЕСЛИ А, ТО В", где А и В - некоторые последовательности элементарных высказываний, которые могут содержать в себе "фактор неопределенности", например<sup>\*</sup>):

---

\*)

Здесь и в дальнейшем мы будем иллюстрировать работу системы на примере проблемно-ориентированной системы ИНТЕГРОН, созданной нами совместно со специалистами фирмы "Агропрогресс" для выбора стратегии по защите капусты от вредителей в период вегетации.

## ЕСЛИ

Период рыхлый кочан	
Заражение капустной молью	5.0-100.0 шт/раст
Заселение растений капустной молью	5.0-100.0%
Возраст гусениц капустной моли	3.0-5.0

## ТО

Биологическая защита НЕ ПРИМЕНЯЕТСЯ!

Блок Анализа Знаний (БЗ) в системе ЭКСНА состоит из следующих разделов:

- 1 - наполнение БЗ, ввод новых продуктов в БЗ (загрузка базы);
- 2 - информация о содержимом БЗ, распечатка содержимого БЗ-словаря и правил в виде ЕСЛИ..., ТО... ;
- 3 - сохранение и восстановление БЗ;
- 4 - обновление БЗ - полная чистка места, отведенного под БЗ;
- 5 - поиск противоречий в БЗ;
- 6 - автоматическое получение знаний из данных;
- 7 - проверка знаний на данных;
- 8 - логический вывод в БЗ.

Далее будут подробно рассмотрены все разделы Блока Анализа Знаний.

### Начало работы с системой ЭКСНА.

Для работы с диалоговой системой необходимо запустить на выполнение файл: DIALOG.BAT. На экране появится текст приветствия и сообщение:

Введите Ваше имя

Нужно набрать имя, предварительно введенное с систему программой LOOKUZER. В поставляемой версии системы введено имя " " (символ пробела), для входа в систему нужно нажать клавишу **↵** (RETURN).

Если имя введено неправильно, то система проинформирует о том, что такого имени в ней нет, и сеанс работы закончен. При благополучном исходе система выдает на экран Главное меню (см. п. "Как начать работу с Блоком Анализа Знаний").

пазначение функциональных клавиш.

Клавиша	Выполняемая функция
cursor UP cursor DN	Перемещение по строкам "меню" вверх и вниз с целью выбора нужного режима
↵ (Return)	1. Переход в отмеченный режим в блоках меню 2. Фиксирование введенного значения параметра
F1	Получение пояснения к отмеченной строке
F10	Выход из системы
F2	Возврат в блок "начало"
ESC	Отказ от работы в текущем режиме
F5	Запуск программы с параметрами на выполнение
f	Запись информации в текстовый файл в процессе работы

Что делать в случае сбоя системы. В случае сбоя системы следует запустить на выполнение файл SBOJ.BAT, который скопирует резервную базу по диалогу в рабочую. Затем для восстановления базы Знаний использовать режим восстановления и сохранения БЗ (если Вы имеете резервную копию БЗ). Полезно также проверить, все ли программы, необходимые для работы системы, имеются в директории INTEGRON. Для этого обратитесь к программе TESTALL. Программа запросит имя файла (система поставляется с файлом INTEGRON.DIR), в котором содержится перечень обязательных программ, проверит их наличие в директории и выдаст соответствующую информацию.


Как завершить работу с системой. Для окончания работы, выхода из системы ЭКСНА и возврата в MS DOS следует нажать клавишу F10. При окончании работы на экран выводится сообщение:

```
===== Сеанс работы закончен =====  
===== До свидания =====
```

Нажатие клавиши ↵ возвратит Вас в MS DOS.

Как начать работу с Блоком Анализа Знаний. В системе ЭКСНА первое меню, которое пользователь видит, войдя в нее, выглядит следующим образом:

```
***** Начало *****  
  
Блок Анализа Знаний  
Блок Анализа Данных  
Запуск произвольной команды операционной системы  
Временный выход в Norton Commander
```

Чтобы войти в Блок Анализа Знаний, следует в Главном меню выбрать пункт "Блок Анализа Знаний", т.е. подвести курсор к этой строке меню и активизировать ее, нажав клавишу . После этого на экране появится меню Блока Анализа Знаний.

```
**** Блок Анализа Знаний ****  
  
Наполнение и восстановление Базы Знаний  
Информация о содержимом БЗ  
Логический вывод в Базе Знаний (БЗ)  
Обновление БЗ  
Работа с редактором  
Автоматическое получение знаний из данных  
Проверка знаний на данных  
Поиск противоречий в БЗ
```

#### Логический вывод в Базе Знаний

Если Вы выбрали в меню Блока Анализа Знаний пункт "Логический вывод в Базе Знаний (БЗ)", Вы увидите на экране следующее меню Логического вывода:

```
БАЗА ЗНАНИЙ - ЛОГИЧЕСКИЙ ВЫВОД  
  
Ввод Исходных данных о Вашей ситуации  
Указание "цели" Логического вывода  
(F3) Просмотр и модификация Исходных данных  
(F2) Вывод заключения Системы  
(F4) Конец работы
```

Чтобы начать работать с Блоком Логического вывода, пользователь должен ввести данные о своей ситуации, т.е. ответить на вопросы системы о его конкретных условиях. Из Ваших ответов система создает временную Базу Исходных Данных, которая будет использована для логического вывода.

Для этого следует активизировать пункт меню "Ввод Исходных данных о Вашей ситуации". На экране появится сообщение:

Ответьте на вопросы, касающиеся Вашей ситуации:

В системе ИНТЕГРОН последовательно задаются вопросы:

- Какой период развития растения?
- Какой вредитель?
- Степень заселенности растений.
- Какой возраст вредителя?
- Какова температура воздуха?
- Сколько дней до уборки урожая?
- Какой препарат будете использовать?

. . . . .

Отвечать на все вопросы не обязательно. Клавишами управления курсивом выберите вопрос, на который Вы хотите ответить, и нажмите **↵** (RETURN).

Для ответов на вопросы система предлагает выбор из ряда возможных вариантов в специальном окне. Например, для ответа на первый вопрос (Какой период развития растения?) выдаются такие варианты:

Выберите нужный показатель.

Период 3-5 дней после высадки в грунт  
Период 7-9 листьев  
Период розетка листьев  
Период рыхлый кочан  
Период уплотненный кочан

Для того, чтобы включить выбранное значение показателя в Исходные данные, подведите курсор к нужной строке и нажмите  $\leftarrow$ . Если среди них нет нужного Вам, нажмите  $\langle \text{Esc} \rangle$ .

Если показатель может принимать различные значения, то напротив имени появятся минимальное и максимальное значения и единицы измерения выбранного показателя. Например: Количество дней до уборки 1 - 120.

Система попросит Вас ввести интервал значений показателя. Можно ввести одно число или интервал значений (два числа, разделенные пробелом, запятой или тире). После нажатия  $\leftarrow$  значение показателя включается в список Исходных данных.

Если Вы ввели значения, выходящие за рамки предусмотренных в системе минимальных и максимальных значений, то система вы - даст предупреждение:

Значения выходят за границы! Введите другие

Если Вы закончили отвечать на вопросы, нажмите  $\langle \text{Esc} \rangle$  или  $\langle \text{F10} \rangle$ , чтобы вернуться в меню Блока Логического вывода.

Используя пункт "Просмотр и модификация Исходных данных", Вы можете получить информацию о том, что уже записано в Исходных данных, удалить "лишние" показатели, изменить значения, удалить все Исходные данные, чтобы описать новую ситуацию. Войти в данный режим можно через основное меню, а при работе в других режимах - нажатием клавиши F3.

Активизировав этот пункт меню, Вы увидите информацию о введенных Вами условиях и "подсказку" о том, какие действия Вы можете выполнить:

Del - удалить, F6 - добавить, F7 - чистить,  
F9 - заменить, F2 - вывод, ESC - выход,

где  $\langle \text{Del} \rangle$  - удалить: из Исходных данных удаляется отмеченный показатель;  $\langle \text{F6} \rangle$  - добавить: при добавление в Исходные данные

путем ввода с клавиатуры необходимо точно знать ИМЯ добавляемого показателя; <F7> - чистить: из Исходных данных удаляются ВСЕ показатели (подготовка к описанию новой ситуации); <F9> - заменить: замена численного значения отмеченного показателя; <F2> - активизировать Логический вывод.

Указание "цели" Логического вывода. Выбор в меню Логического вывода строки "Указание "цели" Логического вывода" используется для ограничения объема информации, выводимой на экран. Если Вы выбрали данный режим, то машина предложит Вам указать одно или несколько ключевых слов, и в процессе Логического вывода на экран будет выдаваться только та информация, которая относится к заданным Вами "целям". Если "цели" не указаны, то на экран выдаются все сообщения и рекомендации, соответствующие введенным Вами Исходным данным. Кроме предлагаемых, Вы можете указать и свои "цели". Общее количество "целей" не должно превышать 10.

Для того, чтобы посмотреть, какие "цели" заданы на текущий момент, нажмите клавишу F3. Для отмены текущих "целей" и возможного задания новых используйте клавишу Del, расположенную в правом нижнем углу клавиатуры.

Вывод заключения Системы (получение рекомендаций Экспертной Системы). Мы уже говорили, что можно отвечать не на все вопросы, перейти к фазе "Вывод заключения Системы", и система попытается сделать вывод на основании введенной Вами информации.

Механизм Логического вывода использует прямую стратегию вывода, которая моделирует способ рассуждения по аналогии. Если из факта А следует факт В и факт С близок в некотором смысле (аналогичен) факту А, то из знания С выводится В с некоторой долей уверенности. Мера расстояния между знаниями позволяет реализовать эту стратегию вывода. Подробнее о мере близости между знаниями см. [6].



Алгоритм Логического вывода использует введенные Вами Исходные данные и все правила, хранящиеся в Базе Знаний. Вне основного меню можно использовать режим "Вывод заключения Системы", нажав клавишу F2.

Например, для исходных данных:

Период рыхлый кочан	
Заражение капустной белянкой	50.0-60.0 шт/раст
Заселение растений капустной белянкой	50.0-60.0%
Возраст гусениц капустной белянки	4.0 дней
Температура воздуха	25.0-30.0 град
Количество дней до уборки	50.0 дней

в результате Логического вывода система выдаст сообщение:

**СИСТЕМА СОВЕТУЕТ ИЛИ ПРЕДУПРЕЖДАЕТ ...**

Химзащита от капустной белянки  
Опрыскивание в период вегетации  
Использовать вентиляторные опрыскиватели и штанговые опрыскиватели  
Норма расхода рабочей жидкости - 300-500 л/га  
Биологическая защита НЕ ПРИМЕНЯЕТСЯ!  
Максимальное число обработок - 2.00 раз  
Токсично для полезных насекомых  
или препарат Децис 2.5% к э  
.....  
или препарат Цимбуш 10% к э  
ЭПВ капустной совки 1-2 шт/раст  
ЭПВ капустной белянки 5-10 шт/раст  
ЭПВ лугового мотылька 8-10 шт/кв м  
ЭПВ капустной моли 5-10 шт/раст

Если размер результирующего сообщения больше экрана, то можно просмотреть результаты Логического вывода, используя клавиши управления курсором.

В системе предусмотрена возможность пояснения полученного результата. Чтобы получить пояснения, нажмите клавишу ←, затем F8. Система последовательно покажет Вам все правила, на основании которых был сделан вывод.

Запись информации в текстовый файл в процессе работы. В будущем Вам может понадобиться информация, которая есть на экра-

не в данный момент. Для записи информации текущего окна в текстовый файл в процессе работы следует нажать клавишу f. После нажатия этой клавиши система создает специальный файл с именем PrFile.pxx, где xx - порядковый номер этого файла. Файл записывается в директорию INTEGRON, и пользователю сообщается его имя.

Например, если Вы нажали клавишу f, находясь в основном меню, система запомнит все пункты основного меню; если в момент нажатия клавиши f Вы вводите Ваши исходные данные, то будут записаны в файл все вопросы, даже те, которых Вы не видите на экране. Точно так же могут быть записаны в файл рекомендации системы (если Вы находитесь в окне "Система советует или предупреждает"), описание исходной ситуации (в окне "Модификация исходных данных") и др.

Конец работы подсистемы Логического вывода. Для того, чтобы закончить работу программы, не выходя в основное меню подсистемы Логического вывода, нажмите клавишу F4. Все правила Базы Знаний сохраняются. Временная база Исходных Данных, созданная системой из Ваших ответов - описание Вашей ситуации, - удаляется.

Подводя итоги, напомним Вам примерный протокол Вашей работы в подсистеме Логического вывода:

1. Ответить на вопросы в режиме "Ввод Исходных данных о Вашей ситуации".
2. Нажать клавишу F3, чтобы посмотреть, какие Исходные данные Вы ввели.
3. Нажав F2, получить Рекомендации Экспертной Системы. Для просмотра правил Базы Знаний, которые использовались при выводе, нажмите ← и затем F8.
4. Добавьте или измените Исходные данные (используйте режимы: Ввод Исходных данных о Вашей ситуации, Указание "цели"

Логического вывода, Просмотр и модификация Исходных данных), затем получите новые рекомендации.

5. Если в результате Логического вывода выдается много информации, используйте режим "Указание "цели" Логического вывода" для ограничения выдаваемой информации.

6. Конец работы.

Если Вас не удовлетворяют результаты работы, обратитесь к разработчикам Экспертной Системы.

### Работа по созданию Базы Знаний

Выбрав в основном меню пункт НАПОЛНЕНИЕ И ВОССТАНОВЛЕНИЕ БАЗЫ ЗНАНИЙ, Вы увидите следующее меню:

Загрузка СЛОВАРЯ Базы Знаний (БЗ)
Загрузка знаний продукционного типа (вида ЕСЛИ..., ТО)
Загрузка закономерностей в виде вершин дерева
Компиляция БЗ во внутреннее представление
Создание РЕЗЕРВНОЙ копии Базы Знаний
Восстановление Базы Знаний из РЕЗЕРВНОЙ копии

Для наполнения БЗ необходимо последовательно выполнить четыре первых пункта этого меню. Этап "Загрузка СЛОВАРЯ Базы Знаний" может отсутствовать, но это значительно замедлит ввод знаний на втором этапе.

Вся вводимая информация готовится заранее с помощью любого текстового редактора. Количество правил в Базе Знаний ограничивается только объемом дисковой памяти и скоростью Логического вывода. (В Базе Знаний, содержащей около 500 правил, время Логического вывода не превышает 1 мин для машин класса IBM PC/AT).

#### Описание структуры входных файлов.

База Знаний состоит из продукции вида: ЕСЛИ A1 & ... & AN, ТО B. Повторим пример продукции, который мы уже приводили:

## ЕСЛИ

Период рыхлый кочан	
Заражение капустной молью	5.0-100.0 шт/раст
Заселение растений капустной молью	5.0-100.0 %
Возраст гусениц капустной моли	3.0-5.0

## ТО Биологическая защита НЕ ПРИМЕНЯЕТСЯ!

Входящие в продукцию элементы  $A_1, A_2, \dots, A_N, \dots, B \dots$  будем называть фактами или показателями. Показатели могут быть как булевыми (т.е. принимающими только два значения - 0 или 1, соответствующие ответам "нет" и "да"), так и количественными. Каждый показатель, сс авляющий продукцию, имеет следующие атрибуты:

- $ИмяI$  - имя показателя: строка символов длиной не более 40 (символы после 40 игнорируются при вводе). Имя может содержать буквы, цифры, пробелы и другие символы, кроме разделителей и скобок. Разделителями служат символы "<", ">", ":", ",", ";", "!",

- $ЕдI$  - для количественных показателей - единицы измерения, для булевых - символ пробела. Параметр  $ЕдI$  задается строкой символов длины не более 8, не содержащей разделителей и скобок;

- $MinI, MaxI$  - минимальное и максимальное возможные значения показателя;

- $cI, dI$  - границы интервала фактических значений показателя ( $MinI, MaxI, cI, dI$  - целые или вещественные числа);

- $pI$  - достоверность показателя (степень уверенности эксперта),  $pI$  - число из интервала  $[0,1]$ , по умолчанию  $pI = 1$ .

ПРИМЕРЫ ИМЕН: Препарат Амбуш 25% к э; Биологическая защита НЕ ПРИМЕНЯЕТСЯ!; Заражение луговым мотыльком.

Здесь два первых показателя - булевы, последний - количественный. "Заражение луговым мотыльком" характеризуется количеством вредителей на одном квадратном метре площади посадки. Известно, что необходимо проводить обработку, если это количество больше 8. Поэтому можно установить:

Для количественного показателя:

единицы измерения:  $EдI = \text{шт/кв.м}$ ;

минимальное и максимальное возможные значения:

$[MinI, MaxI] = [0, 100]$ ;

фактические значения:  $[cI, dI] = [0, 7]$  или  $[cI, dI] = [8, 100]$ .

Для булевых показателей полагаем (автоматически):

единицы измерения:  $EдI =$  ; (символ пробела);

минимальное и максимальное возможные значения:

$[MinI, MaxI] = [0, 1]$ ;

фактические значения:  $[cI, dI] = [1, 1]$  (для "да")

$[cI, dI] = [0, 0]$  (для "нет")

Загрузка "Словаря" Базы Знаний. Программа загрузки "Словаря" Базы Знаний запрашивает один параметр: имя загружаемого файла (в контрольном примере - файл SLOV.DAT).

Вам необходимо задать имя ASCII-файла, в котором содержится "Словарь" Базы Знаний, т.е. записаны имена показателей, их минимальные и максимальные значения и единицы измерения.

Структура входного файла для программы ввода "Словаря" Базы Знаний.

Входной файл, который необходимо подготовить для программы ввода "Словаря" Базы Знаний (см. файл SLOV.DAT), имеет следующую структуру:

```
VOU:
Имя1, Min1, Max1, Ед1*
Имя2, Min2, Max2, Ед2*
.
.
ИмяI, MinI, MaxI, ЕдI*
.
.
ИмяN, MinN, MaxN, ЕдN*
:
КОНЕЦ:
```

Первая и две последние строки играют роль меток для программы, поэтому должны в точности совпадать с указанными. Все остальные обозначения описаны выше. Не допускается замена символов "пробел" знаками табуляции!

Пример входного файла, подготовленного для программы ввода "Словаря" Базы Знаний:

ВОУ			
Опрыскивание в период вегетации	,0.000,1.000,		*
Амбуш токсичен для рыб	,0.000,1.000,		*
Биологическая защита НЕ ПРИМЕНЯЕТСЯ!	,0.000,1.000,		*
Биозащита против лугового мотылька	,0.000,1.000,		*
Возраст гусениц лугового мотылька	,1.000,5.000,		*
Использовать штанговые опрыскиватели	,0.000,1.000,		*
Количество дней до уборки	,0.000,120.000,	дней	*
Корсар токсичен для рыб	,0.000,1.000,		*
Норма расхода Амбуша 25% к э	,0.000,5.000,	кг/га	*
. . . . .			
Обработка не нужна	,0.000,1.000,		*
Токсично для полезных насекомых	,0.000,1.000,		*
Препараты безопасны для окружающей среды	,0.000,1.000,		*
:			
КОНЕЦ:			

Ввод знаний продукционного типа. Программа загрузки знаний продукционного типа требует задания двух параметров: 1) имя файла со знаниями (в контрольном примере - RULES.DAT) и 2) имя файла для записи знаний в виде вершин дерева (в контрольном примере - RULES.LIN).

Первый файл - входной для программы загрузки, в нем содержатся вводимые "знания" (продукции); этот ASCII-файл должен быть создан заранее с помощью любого текстового редактора.

Второй файл - файл знаний в виде вершин дерева - создается в процессе работы программы и будет использоваться на следующем этапе загрузки БЗ.

Правила, загружаемые в БЗ, могут быть подготовлены в нескольких файлах. В этом случае второй и третий этапы загрузки повторяются столько раз, сколько требуется файлов со знаниями.

Правила создания входного файла, содержащего знания продукционного типа. Входной файл программы ввода знаний продукционного типа (файл RULES.DAT) состоит из продукции вида: ЕСЛИ A1 & ... & AN, ТО B, записанных по определенным правилам:

- слова ЕСЛИ и ТО обязательно пишутся в отдельных строках большими буквами; допускается замена ЕСЛИ и ТО на IF и THEN соответственно, написание вместо указанных любых других символов приведет к ошибкам ввода информации;

- отдельная продукция обязательно должна оканчиваться символом "точка с запятой" (;);

- в конце строк, составляющих посылку правила, ставится символ "запятая" (,) или символ "логическое И" (&);

- вместо интервала  $[cI, dI]$  может быть написано:  $> cI$  ("больше"  $cI$ ) или  $< dI$  ("меньше"  $dI$ ). Программа переведет эти данные в стандартную для нее форму - интервалы  $[cI, MaxI]$  или  $[MinI, dI]$  соответственно.

Если во время ввода продукции программа не находит имени показателя в "Словаре" БЗ, то она попросит ввести минимальное и максимальное возможные значения показателя, а также единицы измерения:

```
"Введите МИНИМАЛЬНОЕ значение ИмяI"  
"Введите МАКСИМАЛЬНОЕ значение ИмяI"  
"Введите ЕДИНИЦЫ ИЗМЕРЕНИЯ ИмяI"
```

Для булевых признаков вводить 0, 1, ПРОБЕЛ соответственно. Показатель будет введен в "Словарь" БЗ.

Таким образом, входной файл программы ввода знаний продукционного типа, подготовленный по требуемым правилам, будет иметь следующий вид:

```
ЕСЛИ  
Имя1 [cI,dI],  
Имя2 &  
. . . . . &  
. . . . . ,  
ИмяI > cI,  
  
ТО  
ИмяL [cL,dL] С ВЕРОЯТНОСТЬЮ pL;
```

ЕСЛИ  
ИмяJ &  
. . . . .  
ТО  
ИмяK;  
. . . . .

Пример входного файла для ввода знаний продукционного ти-

па.

ЕСЛИ  
Период рыхлый кочан &  
Заражение капустной молью [5.0, 100.0] &  
Заселение растений капустной молью [5.0, 100.0] &  
Возраст гусениц капустной моли [3.0, 5.0],  
ТО  
Химзащита от капустной моли;  
ЕСЛИ  
Химзащита от капустной моли,  
ТО  
Опрыскивание в период вегетации, с вероятностью 0,9;  
ЕСЛИ  
Период рыхлый кочан,  
ТО  
В данный период капустной мухи нет;  
ЕСЛИ  
Период рыхлый кочан &  
Заражение капустной молью [5.0, 100.0] &  
Заселение растений капустной молью > 5 &  
Возраст гусениц капустной моли [3.0, 5.0],  
ТО  
Биологическая защита НЕ ПРИМЕНЯЕТСЯ!;

Ввод в Базу Знаний закономерностей в виде вершин дерева.

Программа запрашивает имя файла. Необходимо ввести имя того файла, который был создан в результате работы этапа ввода правил ЕСЛИ..., ТО (файл знаний в виде вершин дерева - RULES.LIN).

Компиляция Базы Знаний во внутреннее представление.

После того, как все знания введены, необходима компиляция БЗ во внутримашинное представление. Без этого этапа Логический вывод невозможен, система работать не будет.



Время работы программы компиляции БЗ квадратично зависит от количества введенных правил. Ориентировочное время работы программы для БЗ из 500 правил составляет 10 минут для машин типа IBM PC AT.

#### Создание резервной копии Базы Знаний. Восстановление Базы Знаний из резервной копии.

Вся информация Базы Знаний хранится в файлах с расширением .ARE. При работе с Базой Знаний в результате каких-либо сбоев в ЭВМ или неправильных действий пользователя информация может быть испорчена. Чтобы не проводить повторно процесс наполнения Базы Знаний, рекомендуется иметь Резервную копию БЗ. Если в последующей работе произойдут сбои, то копированием резервных областей в рабочую Базу Знаний восстанавливается быстро и просто. При этом вся информация, имеющаяся в данный момент в Базе Знаний, заменяется на информацию из указанных Вами файлов.

Создание Резервной копии Базы Знаний - это копирование файлов с расширением .ARE в набор файлов с теми же именами и другим расширением. Новое расширение вводит пользователь. Можно иметь несколько Резервных копий (например, на различных этапах создания Базы Знаний), но при этом Вы должны помнить, какие файлы чему соответствуют. При сбое БЗ создавать резервную копию нельзя!

Кроме указанного выше, данным режимом можно воспользоваться для того, чтобы работать с двумя или более независимыми БЗ. Для этого необходимо иметь Резервные копии каждой БЗ и восстанавливать ту, которая требуется в данный момент.

Информация о содержимом Базы Знаний. Распечатка содержимого Базы Знаний. Этот вспомогательный раздел предоставляет возможность просмотра содержимого Базы Знаний и ее словаря. Для этого предварительно все правила Базы Знаний в виде ЕСЛИ ... ТО ... записываются в файл KNOW.TXT, а имена всех показателей, входящих в правила Базы Знаний, а также их возможные минималь-

ные и максимальные значения и единицы измерения - в файл GLOS.TXT. Для булевых признаков минимальное значение 0, максимальное - 1, а вместо единицы измерения - пробелы.

Как запись в файлы KNOW.TXT и GLOS.TXT, так и последующий просмотр этих файлов, обеспечивается выбором соответствующего пункта меню:

Информация о содержимом Базы Знаний

Вывод содержимого Базы Знаний (БЗ) в файл KNOW.TXT  
Просмотр файла KNOW.TXT  
Вывод словаря БЗ в файл GLOS.TXT  
Просмотр файла GLOS.TXT.

Структура создаваемых файлов полностью соответствует той, которая требуется для ввода знаний в БЗ, т.е. эти файлы могут в дальнейшем использоваться в качестве входных при наполнении правилами Базы Знаний или при внесении некоторых изменений в БЗ.

Например, Вы создали какую-то Базу Знаний, но при этом неправильно записали названия некоторых показателей или границы их изменения или обнаружилось еще какие-либо неточности. В режиме получения информации о содержимом БЗ создаем файл со "Словарем" и файл правил продукционного типа ЕСЛИ ... ТО. Используя текстовый редактор, вносим изменения в файлы. Обновляем БЗ. Повторяем процесс наполнения БЗ, и в качестве входных файлов используем те, которые откорректировали. Если первоначально загрузка производилась из нескольких файлов, то это получается гораздо быстрее.

Обновление Базы Знаний - полная чистка. В этом режиме удаляется вся информация из Базы Знаний. Все текстовые файлы с подготовленными данными сохраняются.

"Превращение" Базы Знаний в Экспертную Систему. Итак, Ваша База Знаний готова. Но пока что это лишь хранилище ценной

информации, без возможности автоматического обобщения знаний. Чтобы уже накопленные знания могли быть использованы в качестве основы для получения конкретных рекомендаций, оценки ситуации и т.п., необходима программа Логического вывода.

Так как сама по себе программа Логического вывода есть лишь некоторый универсальный инструмент, она предварительно должна быть настроена на решение проблем Вашей предметной области. Интерфейсом (связующим звеном) между Базой Знаний и программой Логического вывода служит так называемый "файл ключевых слов".

#### Правила создания файла ключевых слов.

Выше уже говорилось, что для получения рекомендации системы необходимо описать исходную ситуацию. При вводе данных об исходной ситуации имена показателей должны в точности совпадать с теми, которые имеются в правилах БЗ.

Чтобы сделать процедуру описания исходной ситуации удобной для пользователя, в программе Логического вывода используется способ ответов на вопросы по выбору из нескольких предлагаемых вариантов. Поэтому для работы программы Логического вывода необходимо создать текстовый файл с именем CLU.TXT, в котором записываются вопросы к пользователю и соответствующие им ключевые слова - по ним программа будет искать варианты ответов среди имеющихся в БЗ показателей.

Создатель Экспертной системы записывает в файл пары "вопрос - ключевой слово". Например, вопрос - "Какой вредитель?", ключевое слово - "Заражение". Пользователю Экспертной системы машина будет задавать этот "вопрос", а используя "ключевое слово", подскажет пользователю варианты возможных ответов:

Заражение капустной мухой  
Заражение луговым мотыльком

Пример файла CLU.TXT

```
7
Какой период развития растения?
Период
Какой вредитель?
Заражение
Степень заселенности растений?
Заселение
Какой возраст вредителя?
Возраст
Какова температура воздуха?
Температура
Сколько дней до уборки урожая?
Количество
Какой препарат будете использовать?
Препарат
5
защита
Обработка
препарат
МДУ
Токсично
```

В первой строке должно стоять целое число - количество ключевых слов, в приводимом примере это число 7. Далее - текст вопроса, который пользователь увидит при работе программы, и под ним соответствующее ключевое слово. Вопрос и ключевое слово должны быть написаны с первой позиции. Длина вопроса не более 55 символов. Для ключевых слов значащими являются не более семи первых символов (или до пробела, если слово короче 7 символов). Для наглядности можно писать все слово. В файле должно быть столько пар "текст вопроса - ключевое слово", сколько указано в первой строке. О второй части файла CLU.TXT будет сказано несколько позже.

Во время работы программа будет выбирать все показатели, в именах которых встречается данное ключевое слово. Проверяется полное совпадение части имени показателя с ключевым словом. Например, если ключевое слово написано с большой буквы, то выберутся только те показатели, в которых данное слово записано с

большой буквы. Могут быть побочные эффекты . Например, если в качестве ключевого слова записано "час", то кроме тех показателей, в которых встречается слово "час", будут выбраны еще и показатели со словами "участие", "частота" и т.д.

При создании Базы Знаний и файла CLU.TXT необходимо следить, чтобы выделенными ключевыми словами "охватывались" все показатели, которые входят только в посылки правил Базы Знаний, но не встречаются среди заключений. Если показатель входит только в заключения каких-либо правил, то желательно, чтобы он не был связан ни с каким вопросом (через ключевое слово).

Вторая часть файла CLU.TXT относится к заданию целей Логического вывода. Вначале записывается целое число - количество ключевых слов (в примере это число 5), а затем сами ключевые слова (с первой позиции по одному в строке). Эти ключевые слова должны быть связаны с той частью фактов (показателей), которые являются заключениями правил. Если конечный пользователь в процессе работы с экспертной системой укажет одно или несколько ключевых слов, то в процессе Логического вывода на экран будут выдаваться только те рекомендации, в которых данные ключевые слова являются частью "имени".

#### Поиск противоречий в Базе Знаний

Напомним, что База Знаний состоит из продукций вида: ЕСЛИ  $A_1 \& \dots \& A_n$ , ТО В, где  $A_i$  и В - некоторые элементарные высказывания. В процессе наполнения Базы Знаний вполне возможно возникновение ситуации, когда могут появиться противоречивые высказывания, т.е. такие цепочки Логического вывода, когда из одних и тех же условий (совокупности левых частей) могут быть получены заключения, несовместимые друг с другом. Такая ситуация может возникнуть в случае, если одно из утверждений ошибочно или если неполны списки условий. В любом случае требуется до-

полнительное рассмотрение вопроса, дополнительный анализ материала.

При наполнении Базы Знаний важной функцией, без которой практически невозможно создание партнерской системы, является возможность автоматического обнаружения противоречий между знаниями, уже имеющимися в Базе Знаний, и новыми знаниями, поступающими от экспертов или от программ извлечения знаний из данных. Более того, результаты автоматического обнаружения противоречий между знаниями имеют и самостоятельную ценность для развития науки, уточнения положений рассматриваемой прикладной области знаний, являются серьезным стимулом для более глубокого изучения возникшей проблемы.

Автоматическое обнаружение противоречий между знаниями в системе ЭКОНА выполняется программой CONTR. Данная программа представляет собой инструмент, помогающий создателю прикладной экспертной системы наполнять Базу Знаний. В программе используется принцип многооконной системы, переходы между окнами осуществляются либо через основное меню, либо нажатием специальных клавиш (список клавиш указывается в информационной строке в нижней части экрана). Для отказа от работы в каком-либо режиме нажимается клавиша ESC. Для каждого окна имеется свой HELP, который вызывается нажатием клавиши F1.

При обращении к программе CONTR Вы увидите ее основное меню, которое проинформирует Вас о том, какие функции выполняет эта программа и какие она предоставляет дополнительные возможности:

Поиск противоречий в Базе Знаний	
	Поиск "претендентов" на противоречия
(F2)	"Расшифровка"
(F3)	Просмотр файла результатов Установка параметров
	Поиск противоречий для одного факта
(F4)	Конец работы

Примерная схема алгоритма поиска противоречий. На первом этапе в БЗ отыскиваются пары "фактов", у которых Имя совпадает, а интервал значения "сильно отличается" (такие пары в программе называются "претендентами на противоречие"). Пороговое значение расстояния между "фактами", при котором они заносятся в "претенденты на противоречие", может быть задано пользователем в режиме Установки параметров (параметр "константа близости" - вещественное число от 0 до 1).

Возможны два варианта выполнения поиска цепочек Логического вывода. Выбор варианта устанавливается заданием параметра (переключателя) "Учет близких", который может принимать логическое значение TRUE или FALSE.

Если установлено значение параметра TRUE, то при поиске цепочек Логического вывода учитываются не только те факты, которые непосредственно указаны в правилах, входящих в цепочку, но и те, которые "близки" по значению (из других правил БЗ). При этом появляются новые цепочки, приводящие к сравниваемым фактам. Факты считаются "близкими", если расстояние между ними меньше, чем разность между 1 и "Константой близости".

На экран выводится Имя факта из пары претендентов и два интервала фактических значений. Для булевых признаков интервал значений не указывается, но если фактическое значение FALSE, то на экран выводится Имя с отрицанием НЕ. Пользователю предлагается просмотреть найденные программой пары "претендентов на противоречие" и отметить (клавишей Ins) те из них, которые, по его мнению, могут действительно быть противоречивыми или просто требуют пояснения.

Не рекомендуется отмечать одновременно слишком много пар из-за возможного большого увеличения объема информации. Лучше отметить одну или несколько пар, просмотреть для них результаты, затем отметить другие и т.д.

Для отмеченных пар в режиме "Расшифровка" программа находит все возможные цепочки вывода, из которых могут быть получены указанные фактические значения. Затем для каждой цепочки сравниваются между собой два множества исходных посылок: те, из которых выводится первый "факт" из пары "претендентов", с теми, из которых выводится второй (противоречащий первому) "факт".

Если эти множества совпадают, то полагаем, что противоречивость равна 1, т.е. это действительное противоречие, так как из одних и тех же исходных значений можно получить два различных заключения. Противоречивость может быть равна 1 и в том случае, если эти множества формально различаются, но при анализе несовпадающих фактов оказывается, что они имеют одно имя и существенно перекрывающийся интервал значений (см. факты 4 и 7 в приведенном ниже примере).

Пример противоречивых высказываний.

\*\*\*\*\* ПРОТИВОРЕЧИЕ \*\*\*\*\*

1	2	3	5	6	7	⇒	10
1	2	3	4	5	6	⇒	11

ЕСЛИ

1	Период рыхлый кочан	
2	Заражение капустной молью	5.0-100.0 шт/кв м
3	Заселение растений капустной молью	5.0-100.0%
5	Защита от капустной моли	
6	Возраст гусениц капустной моли	1.0-2.0
7	Температура воздуха	0.0-12.0 град

ТО

10 НЕ Применяется биологическая защита.

ЕСЛИ

1	Период рыхлый кочан	
2	Заражение капустной молью	5.0-100.0 шт/кв м
3	Заселение растений капустной молью	5.0-100.0%
4	Температура воздуха	0.0-40.0 град
5	Защита от капустной моли	
6	Возраст гусениц капустной моли	1.0-2.0



ТО

11 Применяется биологическая защита

ПРОТИВОРЕЧИВОСТЬ = 1.000

\* \* \* \* \*

В приведенном выше результате найдено действительное противоречие, так как интервалы температуры полностью перекрываются, остальные условия совпадают, а выводы противоположны.

Если сравниваемые множества отличаются, то "процент противоречивости" Р вычисляется по формуле:  $P = C / (S1 + S2 - C)$ , где S1, S2 и C – количество посылок, имеющих в первой, второй и одновременно в обеих цепочках соответственно.

Пример такого варианта.

\*\*\*\*\* ПРОТИВОРЕЧИЕ \*\*\*\*\*

1	2	3	4	5	9	⇒	10
1	2	3	4	5	6	⇒	11

ЕСЛИ

1	Период рыхлый кочан	
2	Заражение капустной молью	5.0-100.0 шт/кв м
3	Заселение растений капустной молью	5.0-100.0%
4	Температура воздуха	0.0-40.0 град
5	Защита от капустной моли	
9	Возраст гусениц капустной моли	3.00-5.00

ТО

10 НЕ Применяется биологическая защита

ЕСЛИ

1	Период рыхлый кочан	
2	Заражение капустной молью	5.0-100.0 шт/кв м
3	Заселение растений капустной молью	5.0-100.0%
4	Температура воздуха	0.0-40.0 град
5	Защита от капустной моли	
6	Возраст гусениц капустной моли	1.0-2.0

ТО

11 Применяется биологическая защита

ПРОТИВОРЕЧИВОСТЬ = 0.667

\* \* \* \* \*

В данном случае, несмотря на частичное совпадение условий, это непротиворечивый результат, так как вывод о применимости биологической защиты определяется в зависимости от возраста гусениц капустной моли, который различен у этих двух "претендентов".

В программе также имеется возможность поиска противоречий только для одного факта. Программа запрашивает у пользователя имя интересующего его факта и затем проверяет, имеются ли в БЗ факты с указанным именем и "сильно различающимися" значениями. Если имеются, то для них повторяется описанный выше алгоритм. Введенное Имя должно полностью совпадать с имеющимся в БЗ, в противном случае на экран выдается сообщение "В Базе Знаний не найден!". В данном режиме проверка происходит не по всей БЗ, поэтому поиск выполняется гораздо быстрее.

Если Вы затрудняетесь назвать имя конкретного факта, то возможен предварительный выбор имени из Базы Знаний по ключевым словам. Ключевое слово может содержать от 1 до 7 символов (символы после 7 игнорируются при поиске). Программа находит в Словаре БЗ все имена фактов, в которых есть указанное ключевое слово, и предлагает пользователю выбрать интересующее его.

Ключевые слова могут быть записаны заранее в файл CONTR.CLU или задаваться во время работы программы (используйте режим "Новое ключевое слово"). Для того чтобы записать в файл введенные Вами ключевые слова, нажмите клавишу F9.

Результаты работы программы (сведения о логических цепочках, для которых "противоречивость" превышает установленное пороговое значение) выводятся на экран и записываются в файл с указанным именем. Имя файла для результатов - строка длиной не более 12 символов (соответствует именам файлов MS DOS, задается в режиме установки параметров).

В режиме "Просмотр файла результатов" программа открывает тот файл, который указан в параметре "Имя файла для результата - тов". Используя режим "Просмотр файла результатов", можно посмотреть любой текстовый файл (даже не связанный с данной программой), если предварительно указать его имя в описываемом параметре. В каждый момент времени программа работает с тем текстовым файлом, имя которого указано в данном параметре, т.е. в него будут записываться результаты работы и из него же они будут считываться в режиме "Просмотр результатов". За один сеанс работы с программой можно многократно изменять имя файла, тем самым разная информация будет занесена в разные файлы.

В режиме "Установка параметров" можно изменить значение некоторых переменных, которые используются в программе. Если Вы хотите, чтобы текущее состояние параметров было запомнено, нажмите клавишу F9. Все значения записываются в файл CONTR.SET и при последующей загрузке программы считываются из него. Если Вы не записали измененные значения, то они сохраняются только на время одного сеанса работы с программой. При входе в пункт основного меню "Установка параметров" программа предлагает Вам новое меню:

```

"Установка параметров"
Имя HELP-файла
"Константа похожести"
Пороговое значение "противоречивости"
Учет "близких"
Имя файла для результатов

```

Назначение всех параметров, кроме параметра "Имя HELP-файла", и способы их задания уже были объяснены. Имя HELP-файла - это имя группы файлов, которые будут выводиться программой на экран при нажатии клавиши F1. Имя файла должно состоять из девяти символов. Четыре последних фиксированы, это <.HLP>. Четыре первых могут быть изменены по желанию пользователя. Пя-

тый символ - цифра - соответствует номеру рабочего "окна" программы, для которой создан этот файл.

Система поставляется с уже готовыми HELP-файлами CONT\*.HLP, но если специалист, разрабатывающий свою прикладную экспертную систему желает создать свои пояснения, он может подготовить свои HELP-файлы по аналогии с уже существующими и указать их имена параметром "Имя HELP-файла".

#### Диагностика ошибок, элементы системного контроля

Во время работы система отслеживает некоторые неправильные действия пользователя и информирует его об этом. Например, сообщение "Нет Исходных данных! Введите хотя бы один показатель." появится в случае, если пользователь попытается сделать Логический вывод (нажмет клавишу F2), не введя данных о текущей ситуации.

Сообщение "Ошибка в записи числа. Введите снова -" вы увидите, если при вводе числовых значений появится нецифровой символ или поставлена запятая вместо десятичной точки ...

При мере необходимости появляются также сообщения:

"Введите новое значение - "

"Не содержит информации. Нажмите Esc."

"Введите имя факта (F8 - выбор по ключевым словам)"

"В Базе Знаний не найден!"

"Используйте другое ключевое слово."

Их смысл не требует пояснений, кроме дополнительной осознанности предупреждения "В Базе Знаний не найден!". Если оно появляется при оперировании с показателем, который уже точно был там раньше, это может означать, что информация Базы Знаний испорчена и рекомендуется ее обновить.

## Проверка знаний на данных

Если Вы имеете текстовый файл с записанными в нем правилами типа ЕСЛИ..., ТО... и таблицу данных, где каждому показателю, участвующему в описании правила, соответствуют числовые значения, то с помощью алгоритма SELECT можно проверить достоверность этих правил. В результате работы алгоритма для каждого правила будут указаны объекты, которые подтверждают данное правило, и отдельно объекты, которые удовлетворяют условиям правила, но не удовлетворяют заключению (противоречат правилу).

Если вместо значения показателя, участвующего в описании правила, встречается пробел, то мы считаем, что это значение не противоречит условию, и тогда такие объекты определяем как условно удовлетворяющие правилу.

Для работы алгоритма необходимо подготовить следующие файлы:

1. Файл с данными WORK.DAT в виде таблицы. В этой таблице каждому из показателей, участвующему в описании правила, соответствует столбец (или строка) числовых значений. Те данные, которые относятся к заключениям проверяемых правил, будем называть целевым вектором. В таблице обязательно должен присутствовать целевой вектор. Если целевого вектора в таблице нет, программа проинформирует об этом, и анализ проводиться не будет.

2. Файл с описанием таблицы данных WORK.PET.

3. Файл со словесным описанием объектов и показателей WORK.INF.

Файл WORK.INF. формируется следующим образом: вначале задаются имена всех объектов, затем имена всех показателей. Имя каждого объекта пишется с новой строки и занимает не более 70 символов, имя каждого показателя также пишется с новой строки, но занимает не более 40 символов.

Правила создания файлов WORK.DAT и WORK.PET будут приведены ниже.

Программа SELECT - проверки знаний на данных - требует для работы задания двух параметров: имя файла с правилами и режим печати. Правила для обработки могут быть получены из Базы Знаний либо взяты из файла, сформированного следующим образом:

```
ЕСЛИ
показатель 4 < 16.4
показатель 6 > 0.8
показатель 1 [ 5.4, 10]
ТО
заключение
```

Необходимо учесть, что:

1. Слова ЕСЛИ и ТО пишутся большими буквами с первой позиции.
2. Все описания условий и целевого признака в правиле должны начинаться с первой позиции и совпадать полностью с описаниями во входном файле WORK.INF. При несовпадении будет выдано сообщение о том, что такого признака или условия в списке показателей нет, и проверка будет выполнена без их учета.
3. Описание показателей в условиях занимает не более 40 позиций. В случае, когда значение показателя булевого типа соответствует отрицанию, частица НЕ пишется большими буквами.
4. Символы > , < , [ , ] (больше, меньше, квадратные скобки) записываются в 41-й позиции, затем через пробел записывается число.

Пример записи такого файла.

```
(1) . . . . . (41) . . . . .
ЕСЛИ
Обеспеченность тракторами < 11.4 ТЫС.ГА,
Внесение органических удобрений [1.8000,1,900] Т/ГА,
НЕ химическая обработка
ТО
Урожайность зерновых [17.5, 30.0];
```

Режимы печати.Выбирая режим печати, Вы регулируете количество информации, выдаваемой на печать. Для программы SELECT возможны следующие варианты:

1 - выдаются сообщения о достоверности правила, порядковые номера и имена объектов, подтверждающих данное правило, и объектов, которые удовлетворяют условиям правила, но противоречат заключению;

2 - дополнительно к режиму 1 для отобранных объектов выдаются значения только тех показателей, которые участвуют в описании правила;

3 - дополнительно к режиму 1 для отобранных объектов выдаются значения всех показателей.

Правила создания файла с данными WORK.DAT.

Числа вводятся последовательно через пробел. Есть два способа ввода: "по объектам", т.е. сначала пишутся все показатели 1-го объекта, затем все показатели 2-го и т.д.; "по признакам" - вначале пишется 1-й показатель для всех объектов, затем 2-й и т.д. Способ ввода данных необходимо отразить в файле описания параметров таблицы.

Правила создания файла параметров таблицы (с расширением .pet).

Любая таблица данных должна быть описана в файле ее параметров, в который заносится информация о числе объектов, признаках, наличии или отсутствии пробелов и т.д. Ниже мы приведем образец "стандартного" файла WORK.PET, в котором для наглядности пронумеруем все строки и дадим дополнительные подробные пояснения по правилам заполнения каждой из строк с номерами 13-18. В "Правилах создания файлов" использована эта же нумерация.

Обратите внимание на то, что единого для всех задач файла WORK.PET не существует. В зависимости от варианта Вашей за-

дачи Вы должны будете заполнять те или иные строки этого файла. Строки 1-12 обязательны и присутствуют всегда. Присутствие информации в строках 13-18 не оказывает никакого влияния на работу программ, к которым эти параметры не относятся.

Стандартный образец файла WORK.PET.

- 1\* |
- 2\* | В строках 1-4 вы можете поместить любые комментарии
- 3\* | к вашей таблице и варианту решения
- 4\* |
- 5\* | Количество объектов
- 6\* | Количество признаков
- 7\* | Вид записи (по признакам - 0, по объектам - 1)
- 8\* | Обозначение пробела: вещественное число (0 - нет пробелов)
- 9\* | Тип задания целевого признака (-1 - вектор рас -
- | пределения указан в п.18, 0 - дополнительно проверяется п.12,
- | > 0 - для некоторых программ номер конкретного признака в таб-
- | ллице данных)
- 10\* | Количество образов
- 11\* | Индекс типов признаков: 0 - разнотипные признаки
- | (тип указан в файле параметров), > 0 - признаки одного типа
- 12\* | Индекс упорядочения по образам (1 - упорядочены, 0 -
- | нет)
- 13\* | "Вектор типов признаков" (заголовок к п. 14)
- 14\* | Значения вектора типов признаков: 1 - булевы, 2 - в
- | шкале наименований, 3 - в шкале порядка, 4 - в сильной шкале
- 15\* | "Вектор количества объектов в образах" (заголовок к
- | п. 16)
- 16\* | Значения вектора количества объектов в образах
- 17\* | "Отдельный целевой признак" (заголовок к п. 18)
- 18\* | Вектор распределения объектов по образам

В пп. 14,16,18 нет ограничений на длину векторов, информация каждого из остальных пунктов должна занимать не более одной строки. При отсутствии информации в каких либо из пп.13-18 резервировать для них место в файле .PET нет необходимости.

Автоматическое получение знаний на данных

Для автоматического обнаружения знаний, содержащихся в таблицах данных, по обучающей таблице и заданному разделению



объектов этой таблицы на классы строится решающее правило в виде Логического дерева. Содержание Логического дерева легко преобразуется в список правил ЕСЛИ ..., ТО ... Эти правила фактически и есть те знания, которыми может наполняться База Знаний экспертной системы.

Построение решающего правила может быть выполнено как на всех признаках, так и только на указанных пользователем. Для того, чтобы не изменять таблицу исходных данных, предусмотрена возможность указания нужных признаков путем задания специального вектора. В этом векторе на месте, соответствующем порядковому номеру признака, должна быть поставлена 1, если признак не будет учитываться при построении решающего правила, и 0 - в противном случае.

Допускаются пропуски некоторых значений признаков, признаки могут быть разнотипными.

Чтобы информацию о Логических правилах, полученных в результате работы алгоритма, представить в выходном файле ACQ.RS в виде правил типа ЕСЛИ..., ТО..., необходимо сформировать файл WORK.INF со словесным описанием признаков и классов. Возможны 2 варианта ввода такой информации:

1. Вначале вводятся имена всех признаков (имя признака занимает одну строку и не более 40 позиций), затем - имена всех классов (также каждое с новой строки, не более 72 позиций).

2. Вначале вводятся все имена объектов (каждое с новой строки, не более 72 позиций), затем имена признаков и классов (как в варианте 1).

Предусмотрено два способа задания параметров: каждый параметр вводится отдельно (режим 5), либо задается определенный набор значений всех параметров (режимы 1-4). Наборы значений параметров, предусмотренные для режимов 1-4, приведены в таблице.

Параметры	Режимы			
	1	2	3	4
Минимальное число неправильно распознанных объектов в вершине	0	0	1	0
Минимальная пороговая величина приращения значения критерия в вершине	0	0.01	0	0.05
Величина параметра смещения в критерии правдоподобия	1.	0.8	0.5	1
Минимальное число объектов в конечной вершине	1	1	5	3
Порог валидности (для всех признаков одинаков)	0	0	0.01	0.01

Выбранный режим (т.е. заданная совокупность значений параметров) определяет различные способы построения Логического дерева и, в конечном итоге, вид полученных правил. Изменяя режимы, Вы получите несколько вариантов правил, из которых сможете выбрать наиболее подходящий для Вас вариант решения.

В зависимости от режима работы программы при входе в соответствующий раздел меню системы Вам будут заданы вопросы об условиях решения. Общий список вопросов приведен ниже.

1	способы формирования файла со словесным описанием (1,2)
2	ограничение на количество получаемых правил при построении логического дерева
3	учитывать ли запреты на признаки (0 - нет, 1 - да)
4	имя файла с вектором, определяющим запреты на признаки
5	по какому критерию вести разбиение (1,2)
6	способы получения правил (1-5)
7	1-е условие прекращения ветвления в вершине
8	2-е условие прекращения ветвления в вершине
9	значение параметра смещения для вычисления критерия правдоподобия
10	3-е условие прекращения ветвления в вершине
11	одинаков ли порог валидности для всех признаков (0 - да, 1 - нет)
12	порог валидности
13	имя файла с вектором значений порогов валидности

Система задает только те вопросы, которые нужны для дан - ного конкретного варианта. Все варианты возможных меню могут быть получены на основе приведенной ниже таблицы:

Если параметр	принимает значение	то нет вопроса
3	0	4
5	1	9
6	1-4	7-13
11	0	13
11	1	12

Например, если при ответе на 6-й вопрос меню "способы получения правил" Вы выбрали один из вариантов (1-4), то система уже не будет задавать оставшиеся вопросы (7-13) - нужные программе значения этих параметров будут проставлены "по умолчанию". В этом случае Вы увидите только короткое меню:

1	способы формирования файла со словесным описанием (1,2)
2	ограничение на количество получаемых правил при построении логического дерева
3	учитывать ли запреты на признаки (0 - нет, 1 - да)
4	имя файла с вектором, определяющим запреты на признаки
5	по какому критерию вести разбиение (1,2)
6	способ получения правил (1-5)

Поясним еще назначение параметра 12 (порог валидности). Этот параметр определяет допустимое минимальное отличие между значениями одного и того же показателя при включении его в решающее правило для разделения разных классов. Другими словами, значения показателей в логических высказываниях, разделяющих классы, должны "достаточно существенно" отличаться друг от друга.

Если порог одинаков для всех признаков, то достаточно задать его только один раз (задавать нулем в случае несущественности критерия). В противном случае пороговые значения долж-

ны быть указаны для каждого признака в произвольном файле (с именем, отличным от WORK.\*) рядом последовательных чисел, разделенных пробелами. Если для каких-либо признаков валидность учитывать не нужно, то на соответствующем месте следует поставить "0".

Условия останова. В процессе построения решающего правила прекращение ветвления дерева в каждой вершине может происходить, если в данной вершине не превосходит указанного минимума одна из следующих трех величин:

- число неправильно распознанных объектов;
- пороговая величина приращения значения критерия;
- число объектов.

Критерием для окончания процедуры построения дерева также является предельно допустимое число вершин. В данной программе число вершин не должно превышать 35.

#### Конструктор диалога

Данный раздел представляет интерес для пользователей, решивших написать свой сценарий диалога. Для этого необходимо ознакомиться с правилами создания дерева диалога и с работой программ, осуществляющих его загрузку в систему, т.е. подготавливающих программно реализованный алгоритм, готовый к использованию. Загрузка диалога осуществляется из файла, в котором предварительно создается дерево.

#### Правила описания дерева диалога в файле

Дерево диалога - это последовательность описаний входящих в него блоков и переходов (ссылок) между ними. Блок - часть дерева диалога, связанная с конкретным пунктом меню, создаваемая для независимой части системы, выполняющей решение отдельной законченной самостоятельной задачи. Каждый блок имеет уникальный номер.

Перед описанием блоков приводится текст ПРИВЕТСТВИЯ пользователю, который появляется на экране при начале работы с диалогом. Текст приветствия в файле оформляется так: с первой позиции первой строки пишется слово "ПРИВЕТСТВИЕ"; в последующих строках длиной не более 72 символов - сам текст. Сразу же обращаем Ваше внимание на то, что если Вы хотите использовать в Вашем тексте символ "\*", то его нельзя ставить с первой позиции - звездочка в первой позиции воспринимается программой как разделитель между описаниями различных блоков.

#### Пример описания приветствия.

##### ПРИВЕТСТВИЕ

```
* * * * *
*          ЗДРАВСТВУЙТЕ!   СИСТЕМА  ЭКСНА ГОТОВА К РАБОТЕ!   *
*                                ВЕРСИЯ  01.92                   *
* * * * *
*****
```

Описание приветствия и начало описания блоков разделяется строкой звездочек - обязательно с 1-й позиции новой строки, не более 72 символов в строке.

Итак, после приветствия идут описания блоков, каждое из которых отделяется строчкой звездочек (от 1 до 72). Существует два типа блоков: блок описания переходов (блоки меню и вопросов на схеме дерева диалога) и блок вызова обрабатывающего модуля (или просто обрабатывающий модуль, сокращенно ОМ). Номер блока для обрабатывающего модуля всегда задается больше тысячи. Для работы диалога обязательно присутствие в описании дерева блока с номером 1. Именно с этого блока начинается работа в системе.

#### Блоки описания переходов

Есть два вида блоков переходов - блоки меню и блоки вопросов. Рассмотрим сначала описание первых, а затем вторых.

Блоки меню. В первой строке, начиная с первой позиции, помещается имя блока длиной не более 72 символов. Во второй строке записывается номер блока и код защиты блока (целые числа, разделенные одним или более пробелов). Номер для блока перехода задается числом, меньшим 1000.

Код защиты (число от 0 до 5) позволяет сделать блок доступным только пользователям, имеющим статус не больше этого кода (статус, как и имя пользователя, загружается специальной программой LOOKUZER). Например, если пользователь Иванов имеет статус 2, то ему не будут доступны блоки с кодами защиты 1, 0. Система организована так, что если пользователь не может перейти на защищенный от него блок из меню, то соответствующая строка меню не высвечивается на экране. Таким образом, защищенный блок маскируется.

Далее, начиная со следующей строки, идет текст меню. Каждая строка меню соответствует одному переходу и имеет длину не более 72 символов. После текста меню в первой позиции новой строки ставится запятая, обозначающая конец данного меню.

Затем идет текст комментариев к описываемому блоку. Эти комментарии будут высвечиваться в процессе работы системы при нажатии клавиши F1. Комментарии могут состоять из нескольких фрагментов, каждый из которых имеет вид:

Первая строка: символ  $\langle * \rangle$  в первой позиции, за которым следует одно или несколько целых чисел, разделенных знаком пробела или запятой. Число соответствует порядковому номеру строки меню. Например, если записано  $* 1, 2$ , то это означает, что комментарии относятся к строкам 1 и 2 (если Вы хотите написать одинаковый комментарий ко всем строкам меню, то наберите после символа  $*$  номера всех строк через запятую).

Затем в последующих строках текст комментария: произвольное количество строк длиной не более 72 символов каждая. Если

длина строки будет больше, то информация, начиная с 73 позиции, пропадает.

После последнего фрагмента комментариев для обозначения конца описания комментариев ставится по одной запятой в двух следующих строках в первой позиции.

Пример написания комментариев.

※ 1

1 - обрабатывают двумерную таблицу программами из ППП ОТЭКС и АГРИН

※ 2

2 - смотрите инструкции.

ПРИМЕЧАНИЕ. Создавать комментарии не обязательно, можно обойтись и без них. В этом случае после текста меню должны идти запяты в первых позициях трех строк. Тогда при нажатии клавиши F1 в режиме диалога пользователь увидит на экране сообщение об отсутствии комментариев.

Далее следует одна или несколько строк, в которых через пробел записаны целые числа - номера блоков для переходов в соответствии с порядком строк в меню: номер блока перехода для первой строки меню, затем номер блока, соответствующий второй строке, и т.д.

Пример описания блока меню под номером 1.

\* \* \* \* \*

\* \* \* \* \* НАЧАЛО \* \* \* \* \*

1            5

подсистема прикладных программ  
работа с таблицами

,

※ 1

1 - обрабатывают двумерную таблицу программами из ППП ОТЭКС и АГРИН

※ 2

2 - смотрите инструкции

,

,

3            4

\* \* \* \* \*

Обратите внимание, что звездочки, используемые в названии блока, записаны не с первой позиции!

Если пользователь выберет подсистему прикладных программ, то он перейдет на блок номер 3, если работу с таблицами, - то на блок номер 4.

Пример описания блока без комментариев.

```
* * * * *
* * * * * НАЧАЛО * * * * *
  1   5
подсистема прикладных программ
работа с таблицами
,
,
,
  3   4
* * * * *
```

Блоки вопросов.

В первой строке записывается текст вопроса длиной не более 72 символов, начиная с первой позиции; во второй строке - номер блока и код защиты аналогично блоку описания переходов; далее - текст возможных ответов. Текст может состоять из нескольких строк, каждая из которых длиной не более 72 символов, или из двух строк, в первой из которых ставится ДА или Y, а во второй НЕТ или N. После описания ответов ставится запятая в первой позиции следующей строки.

Текст комментариев подготавливается так же, как и для блоков меню. Заканчивается блок вопросов строкой, содержащей номера блоков для перехода. Номера записываются через пробел. Первый номер соответствует первому ответу, второй - второму и т.д. В случае вопроса "да-нет" первым записывается номер блока, в который будет осуществляться переход при ответе ДА (Y), вторым - номер блока для перехода при ответе НЕТ (N).

Например. Пусть мы хотим подготовить описание блока под номером 2 и чтобы система, задав вопрос "Таблица сформирова-



на?", при ответе ДА перешла на блок под номером 1013, а при отрицательном - на блок под номером 1. Мы хотим также предусмотреть и возможность получения комментариев, если у пользователя возникнет заминка с ответом. В комментариях напишем: "при ответе ДА Вы будете работать с алгоритмом KRAB, при ответе НЕТ Вам будет предложено меню блока номер 1". Текст блока-вопроса нужно набрать так:

\* \* \* \* \*

Таблица сформирована?

2 5

ДА

НЕТ

,

# 1,2

При ответе ДА вы будете работать с алгоритмом KRAB,

При ответе НЕТ вам будет предложено меню блока номер 1.

,

,

1013 1

\* \* \* \* \*

Если комментарии не вводятся, то данный блок оформляется так:

\* \* \* \* \*

Таблица сформирована?

2 5

ДА

НЕТ

,

,

,

1013 1

\* \* \* \* \*

Правила описания блока обрабатывающего модуля. Первая строка - имя блока длиной не более 72 символов, начиная с первой позиции.

Вторая строка - номер блока и код защиты блока (аналогично блокам меню за исключением того, что номер блока обрабатывающего модуля задается всегда числом большим 1000).

Третья строка - количество входных параметров ОМ (целое число). Эта строка должна присутствовать обязательно! Если входных параметров нет, то ставится 0 (ноль). В примере в данной строке написано помимо числа еще и слово "параметров", которое служит лишь для удобства просмотра файла с деревом и поэтому может отсутствовать или быть замененным другим словом.

Фрагменты описания параметров. Каждый фрагмент имеет стандартную структуру:

- первая строка - номер параметра (номера параметров должны быть строго упорядочены по возрастанию);

- вторая строка - название параметра длиной не более 72 символов. Это название будет выводиться на экран пользователя при входе в ОМ в процессе диалога;

- третья строка - значение параметра по умолчанию (текст не более 72 символов). Это значение будет также выводиться на экран при работе с диалогом, информируя о том, какое значение параметра будет использовано программой по умолчанию, если вы не введете другое значение.

Если от значения описываемого параметра никакие другие параметры не зависят, то делается переход к описанию следующего, в противном случае описываются "нестандартные" значения этого параметра.

Каждый фрагмент такого нестандартного описания состоит из трех строк:

Первая строка начинается со слова ЕСЛИ, после которого может стоять любой текст до 72 позиции, например, "ЕСЛИ параметр" или просто ЕСЛИ (большими буквами).

Следующая строка - непосредственно значение параметра, при котором отсутствуют какие-либо другие параметры.

И, наконец, третья строка содержит слово ТО НЕТ и номера тех входных параметров, которые не требуются при данном значении. Количество таких фрагментов для каждого параметра с усло-

виями не ограничено. После описания всех нестандартных значений ставится символ ",", (запятая) в первой позиции новой строки. Номера параметров, которые нужно исключить, отделяются друг от друга пробелами.

Если пользователь при обращении к ОМ из диалога задаст значение параметра, не совпадающее ни с одним из описанных нестандартных, то это не отразится на задании оставшихся параметров, т.е. они будут задаваться в том порядке, в котором они описаны. Если будет введено одно из нестандартных значений, то параметры с номерами, указанными после слов ТО НЕТ, запрашиваться у пользователя не будут. Конец описания параметров оформляется запятой в первой позиции новой строки.

В примере 1, приведенном ниже, нестандартным параметром является параметр номер 5. Если пользователь задаст из диалога значение параметра равное 1, то у него не будут запрашиваться значения параметров под номерами 6 и 7. Если пользователь задаст из диалога значение параметра равное 2, то у него не будет запрашиваться значение параметра под номером 8.

Текст пояснений (комментариев) к параметрам: любое количество строк длиной не более 72 символов. Комментарии оформляются как и для блоков перехода. Единственное различие заключается в том, что номера комментариев соответствуют номерам описанных параметров. Конец описания комментариев оформляется запятой в первой позиции новой строки.

Строки команд операционной системы: каждая команда должна быть записана в отдельной строке с первой позиции. Число команд произвольно. Данный раздел описания ОМ предоставляет широкое поле деятельности (ограниченное лишь рамками той операционной системы, в которой Вы работаете).

Заканчивается данная часть описания ОМ запятой в первой позиции новой строки.

В последней строке описания блока указывается номер блока, в который будет осуществляться переход после выполнения ОМ.

Значения параметров для решения конкретной задачи могут быть введены из диалога. При входе в блок обрабатывающего модуля, у которого параметры задаются из диалога, система запросит значения требующихся ей параметров. После того как Вы ответите на все вопросы, т.е. введете все требующиеся значения параметров и запустите задание на выполнение, система запишет значения заданных параметров в файл WOKR.PAR (причем каждое значение будет записано в отдельной строке). Затем система запустит на выполнение программу KRAB, которая, в свою очередь, считает параметры из файла WORK.PAR и начнет работу.

ПРИМЕР 1. Описание блока 1013 KRAB (значения параметров вводятся из диалога).

\* \* \* \* \*

KRAB

1013

5

9

параметров

1

требуемое количество таксонов

4

. . . . .

4

режим печати (0 - 3), 0 - нет печати, 1,2 - итог, 3 - полная

1

5

константа, задающая обращение к программе: KRAB - 1, KRAB2 - 2

1

ЕСЛИ параметр принимает значение

1

ТО НЕТ 6 7

ЕСЛИ параметр принимает значение

2

ТО НЕТ 8

,

6

количество таксонов для п/п FOREL2

8

7

количество итераций для п/п FOREL2

3

8

режим печати программы TAKSON (0,1), 0 - короткая, 1 - полная

0

. . .

,

≠ 2

Имеется в виду равномерность распределения количества объектов по таксонам.

≠ 3

Если  $KR2=0$ , то производится полный перебор ребер КНП для нахождения границ между таксонами. Если  $KR2 \neq 0$ , то производится предварительный отбор ребер-претендентов на проведение границ между таксонами. При этом из рассмотрения исключаются ребра, которые короче самого короткого из примыкающих ребер, т.е.  $\alpha = (L_{min}/L(i)) > 1$ , где  $L$  - длина ребра

,

KRAB

3

\* \* \* \* \*

В случае, когда описание параметров OM отсутствует, происходит выполнение команд (или команды) операционной системы, предусмотренных пользователем в описании данного блока. Тогда описание блока 1013 KRAB должно будет выглядеть следующим образом.

ПРИМЕР 2. Блок KRAB без описания параметров.

\* \* \* \* \*

KRAB

1013

5

0

параметров

KRAB

,

3

\* \* \* \* \*

Выполняя указания этого блока, система просто запустит на исполнение файл KRAB.EXE.

**ПРИМЕР 3. Использование команд операционной системы.**

```
* * * * *
      * * * Norton Commander * * *
3331   5
      0
c:\norton\ncnew
c:
cd \dialog
'
  1
* * * * *
```

**Оформление конца дерева и рекомендации**

Оформление конца дерева. В последней строке файла (после описания всех блоков) записывается слово КОНЕЦ большими русскими буквами с первой позиции. Слово КОНЕЦ должно стоять непосредственно под строкой звездочек, относящихся к концу описания последнего блока:

```
'
  33
* * * * *
КОНЕЦ
```

Рекомендации. Все номера блоков должны быть различны, причем номера могут задаваться произвольно, т.е. нет определенных ограничений на порядок их задания. Единственное замечание - номера ОМ должны быть заданы в соответствии с пунктом 2 рекомендаций, а номера блоков переходов задаются меньше 1000. Все блоки обрабатывающих модулей, являющиеся конечными вершинами дерева, имеют номер больше или равный 1000.

Не существует никаких ограничений на порядок записи описаний блоков, т.е., например, после блока 1 может быть описан блок 5, потом блок 2, блок 1022, затем блок 1020 и т.д.

Не рекомендуется в текстовых строках в качестве первого символа использовать символы, приведенные далее в кавычках: ",", ":", ":", ":", ":", так как это может привести к неправильному вводу.

Напоминаем о том, что описания всех блоков должны быть отделены друг от друга строками звёздочек.

Построенное описанным выше способом дерево диалога должно быть записано в файл \*\*\*\*\*.DIA, где \*\*\*\*\* - его имя длиной в 6 символов.

#### Загрузка дерева диалога из файла

Сценарий (дерево) диалога пользователя с системой загружается из файла \*\*\*\*\*.DIA в Базу Данных диалога - в файлы \*.ARE. Загрузка дерева диалога из файла в систему производится программой DIIN, имена файлов \*.ARE присваиваются автоматически. Предварительно необходимо скопировать пустую базу в рабочую командой COPY \*.NUL \*.ARE.

Для вызова программы DIIN на экране терминала необходимо набрать: DIIN и нажать клавишу  $\leftarrow$ . Далее программа запросит имя файла с описанием дерева (без расширения!) и, если имя указано правильно, начнет ввод информации. Файл \*\*\*\*\*.DIA, в котором описано дерево диалога, должен быть создан в соответствии с инструкцией (имя файла должно иметь длину 6 символов).

#### Схема работы программы DIIN.

1. Считывается из файла \*\*\*\*\*.DIA имя блока и номер блока.
2. Делается попытка найти блок с таким номером в системе.
3. Если блок не найден, то вся информация об этом блоке переписывается из файла в систему.
4. Если блок найден, то на экран выдается предупреждающее сообщение о существовании блока и запрашивается согласие пользователя на удаление старой информации и ввод новой. Если ответ Д или У, то происходит замена. Если любой другой ответ, то существующий блок сохраняется, а в файле пропускаются все

и статус нового пользователя. В ответ Вы должны набрать необ - ходимую информацию и нажать клавишу  $\leftarrow$ . Имя и статус пользо - вателя становятся известными системе;

F3 - изменить статус пользователя;

F8 - удалить имя пользователя из системы;

F4 - выход из программы.

### Требования к программам для работы в Блоке Анализа Данных системы ЭКОНА

Для включения в Блок Анализа Данных системы ЭКОНА требу - ется программа в виде .EXE-файла. Желательно, чтобы включаемая программа работала в едином ключе с остальными программами си - стемы. Для этого необходимо выполнение достаточно простых тре - бований, которые будут пояснены далее.

Все программы должны быть построены так, чтобы они рабо - тали с "текущими" файлами - файлами со стандартными именами:

WORK.DAT - таблица исходных данных;

WORK.PET - описание таблицы;

WORK.INF - наименования объектов и признаков;

WORK.PAR - значения входных параметров;

TABL.DAT - результат решения.

Подробно правила создания файлов WORK.DAT, WORK.PET, WORK.INF приведены в описании Блока Анализа Данных.

Файл WORK.INF не обязателен. Для некоторых задач есть воз - можность, по желанию пользователя, получить "словесно докумен - тированный" результат, т.е. наименования объектов и признаков могут быть выданы в их истинном виде, а не заменены номерами. В этом случае в директории DIALOG файл WORK.INF должен быть за - дан.

Для работы с алгоритмами выбора информативных признаков и распознавания необходимо выполнение ряда дополнительных требо - ваний (смотрите справки к данным режимам в меню Блока Анализа Данных).



Файл WORK.PAR программно формируется каждый раз при запуске задачи на выполнение - по окончании задания значений параметров блока обрабатывающего модуля. Программа должна считывать значения входных параметров из файла WORK.PAR. Каждый параметр пишется в дереве диалога в отдельной строке. Длина строки для задания значений по каждому параметру из диалога ограничена 72 символами.

Считывание параметров программой должно вестись в соответствии с их описанием в дереве диалога. В случае отсутствия входных параметров в Вашем программном модуле при описании его в дереве диалога после первых двух строк пишется строка - 0 параметров (см. Правила описания Обрабатывающего Модуля в дереве диалога).

Для простоты выявления ошибок при загрузке в систему измененного дерева рекомендуется добавлять описания новых блоков в конец дерева.

#### Л и т е р а т у р а

1. ЁЛКИНА В.Н., ЗАГОРУЙКО Н.Г. Блок Анализа Данных в экспертной системе ЭКСНА //Экспертные системы и анализ данных. - Новосибирск, 1991. -Вып. 144: - Вычислительные системы. -С.54-175.
2. ЗАГОРУЙКО Н.Г., ЁЛКИНА В.Н., ЕМЕЛЬЯНОВ С.В., ЛБОВ Г.С. Пакет программ ОТЭКС для анализа данных. - М.: Финансы и статистика, 1986.
3. ЁЛКИНА В.Н., ЗАГОРУЙКО Н.Г., НОВОСЕЛОВ Ю.А. Математические методы агроинформатики. - Новосибирск: Наука СО, 1987.
4. УОТЕРМЕН Д. Руководство по экспертным системам /Пер. с англ. под ред. В.Л.Стефанюка. - М.: Мир, 1989.
5. Будущее искусственного интеллекта /Редакторы-составители Д.А.Поспелов, К.Е.Левитин. - М.: Наука, 1991.
6. ЗАГОРУЙКО Н.Г. Партнерские системы //Анализ данных и знаний в экспертных системах. - Новосибирск, 1990. Вып. 134: Вычислительные системы. - С. 3-18.

7. ПОПОВ Э.В. Экспертные системы: Решение неформализованных задач в диалоге с ЭВМ. -М.: Наука, Гл. ред. физ.-мат.лит., 1987.

8. ЗАГОРУЙКО Н.Г. Экспертные системы и распознавание образов //Анализ данных в экспертных системах. - Новосибирск, 1986. -Вып. 117: Вычислительные системы. -С. 3-10.

9. ЗАГОРУЙКО Н.Г., БУШУЕВ М.В. Меры расстояний в пространстве знаний //Там же. - С.24-35.

Поступила в ред.-изд.отд.

20 июля 1992 года