**Л 8. Вывод на знаниях**

База правил, рабочая память и механизм вывода. Компонента вывода и управляющая компонента. Операции цикла вывода.

Демоны, присоединенные процедуры и механизм наследования в ИС с фреймовой МПЗ. Правила вывода в логических моделях.

1. **База правил, рабочая память и механизм вывода**

Благодаря простоте, модифицируемости и гибкости в применении знаний для решения задач продукционная модель, как оказалось, может быть важным механизмом для конструирования экспертных систем и других приложений ИИ.

База знаний в этом случае состоит из набора правил, а сама экспертная система должна содержать три основные компоненты: **базу правил, рабочую память и механизм вывода**.

**База правил** (БП) — формализованные с помощью правил продукций знания о конкретной предметной области.

**Рабочая память** (РП) - область памяти, в которой хранится множество фактов, описывающих текущую ситуацию, и все пары атрибут-значение, которые установлены к определенному моменту. РП - это динамическая часть базы знаний, содержимое которой зависит от окружения решаемой задачи. В простейших ЭС факты РП не изменяются в процессе решения задачи, но существуют системы, в которых допускается изменение фактов в РП.

**Механизм вывода** выполняет две основные функции: просмотр существующих в рабочей памяти фактов и правил из БП, а также добавление в РП новых фактов; определение стратегии поиска и применения правил.

1. **Компонента вывода и управляющая компонента**

Для выполнения указанных функций механизм вывода включает **компоненту вывода и управляющую компоненту**.

Действие компоненты вывода основано на применении правила вывода, называемого Modus Ponendo Ponens (правило отделения), которое звучит в продукционных системах так: «если верно А и верно, что из А следует В, то верно В». То есть если в РП присутствует факт А, и в БП существует правило вида «ЕСЛИ А, ТО В», то факт В признается истинным и заносится в рабочую память. Такой вывод легко реализуется на компьютере.

Невысокие интеллектуальные возможности продукционных систем обусловлены тем, что ЭС способны вывести лишь небольшое количество заключений, используя заданное множество правил. При этом компонента вывода ЭС должна иметь способность функционировать в условиях недостатка информации.

1. **Операции цикла вывода**

Управляющая компонента определяет, порядок применения правил, а также устанавливает факты, которые могут быть изменены в случае продолжения работы. Механизм вывода работает циклически, при этом в одном цикле может срабатывать только одно правило. Схема цикла показана на рис. 8.1. В цикле выполняются следующие операции:

1) **сопоставление** - образец правила сопоставляется с имеющимися в РП фактами;

2) **выбор** - если в конкретной ситуации может быть применено сразу несколько правил, то из них выбирается одно, наиболее подходящее по заданному критерию, т. е. происходит разрешение конфликтного выбора;

3) **срабатывание** - если образец правила при сопоставлении совпал с какими-либо фактами из РП, то правило срабатывает, оно отмечается в БП;

4) **действие** - рабочая память подвергается изменению путем добавления в нее заключения сработавшего правила.

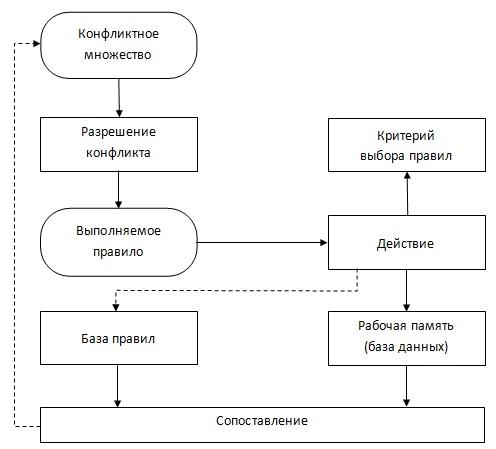


Рис. 8.1. Цикл работы механизма вывода

Условием окончания цикла является либо исчерпание всех правил из БП, либо выполнение некоторого условия, которому удовлетворяет содержимое рабочей памяти (например, появление в ней какого-то образца), либо комбинация этих способов.

**Особенностью ЭС является то, что они не имеют процедур, способных построить в пространстве состояний сразу весь путь решения задачи. Траектория поиска решения полностью определяется данными, получаемыми от пользователя в процессе вывода.**

Стратегия управления выводом зависит от выбранного метода поиска, описанного выше, от него будет зависеть порядок применения и срабатывание правил. Процедура вывода сводится к определению направления поиска и способа его осуществления. Процедуры, реализующий поиск, как правило, недоступны инженерам по знаниям. При выборе стратегии необходимо решить, какую точку в пространстве состоянии принять в качестве исходной (в прямом или обратном направлении), какова стратегия перебора (в ширину, в глубину, по подзадачам или иначе).

1. **Демоны, присоединенные процедуры и механизм наследования в ИС с фреймовой МПЗ**

В интеллектуальных системах с **фреймовым представлением знаний** используются **три способа** управления логическим выводом: **демоны, присоединенные процедуры и механизм наследования**, который является основным механизмом вывода.

Управляющие функции механизма наследования заключаются в автоматическом поиске и определении значений слотов фреймов нижележащих уровней по значениям слотов фреймов верхних уровней, а также в запуске присоединенных процедур и демонов. Присоединенные процедуры и демоны могут реализовать любой механизм вывода, однако, эта реализация имеет конкретный характер и требует значительных затрат труда проектировщиков и программистов.

Итак, в интеллектуальных системах с фреймовым представлением знаний невозможно четко отделить процедурные знания от декларативных, поскольку демоны и присоединенные процедуры одновременно являются и знаниями, и средствами управления логическим выводом.

Пример:

Фрейм «Научная конференция»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя слота | Значение слота | IF-NEDDED | IF-ADDED | IF-REMOVED |
| Дата | 1.06.10:10 |  |  |  |
| Место проведения | Аудитория 533 |  | ЗАКАЗ |  |
| Тема доклада | Генетические алгоритмы при решении оптимизац. задач |  |  |  |
| Докладчик | Терехов В. И. | КТО? |  |  |

При попытке вставить значение в слот Место проведения автоматически запускается демон ЗАКАЗ, который является процедурой. Демон КТО? автоматически запускается при обращении к слоту Докладчик. Если значение этого слота не определено, то происходит генерация запроса «Кто выступает?», получение ответа и его запись в качестве значения слота.

Реализация вывода с помощью присоединенных процедур требует наличия механизма обмена между фреймами, таким механизмом обычно бывает механизм сообщений.

В **семантических сетях**, так же как при фреймовом представлении знаний, декларативные и процедурные знания не разделены, следовательно, база знаний не отделена от механизма вывода. Процедура вывода обычно представляет совокупность процедур обработки сети.

1. **Правила вывода в логических моделях**

Для рассмотрения вывода в логических моделях необходимо ввести следующие определения.

Неравнозначностью (исключающим «или», сложением по модулю 2 двух высказываний А и В называется высказывание, истинное, когда истинностные значения А и В не совпадают, и ложное в противном случае. Обозначение: АВ, читается «или А, или В», «или» понимается в разделительном смысле. Составное высказывание «Сегодня понедельник или вторник» состоит из двух простых: А = «Сегодня понедельник»; В = «Сегодня вторник». Эти высказывания соединены связкой «или» в разделительном смысле. Таким образом, данное высказывание представимо формулой А⨁В.

Если описание системы представлено совокупностью сложных высказываний—логических формул, истинных для данной системы, то с помощью логических преобразований могут быть получены новые представления, характеризующие данную систему (истинные для данной системы). Таким образом, появляется возможность вывода новых знаний из имеющихся с помощью допустимых в логике преобразований. Итак, операция, называемая выводом, — это получение заключения из группы заданных представлений; вывод является формулой, полученной из некоторой группы описанных формул. Как правило, эти формулы истинны при каких бы то ни было допустимых интерпретациях.

Действие компоненты вывода в логических моделях основано на применении следующих правил вывода:

1. Modus Ponens «если верно А и верно, что из А следует В, то верно В»:

2. Modus Tollens «если верно, что из А следует В и В ложно, то ложно А»:

3. Modus Ponendo Tollens «если верно или А или В (в разделительном смысле) и верно одно из них, то другое ложно»:

4. Modus Tollendo Ponens

а) «если верно А или В (в разделительном смысле) и ложно одно их них, то верно другое»:

б) «если верно А или В (в неразделительном смысле) и ложно одно из них, то верно другое»:

5. Правило транзитивности «если из А следует В, а из В следует С, то из А следует С»:

6. Закон противоречия «если из А следует В и B, то ложно А»:

7. Правило контрапозиции (или метод резолюций) «если из А следует В, то из того, что ложно В, следует, что ложно А»:

Пример:

Студент сдал экзамен на «хорошо» или «отлично». Он не получил отличной оценки. Следовательно, он сдал экзамен на «хорошо».

Введем обозначения: А = «студент сдал экзамен на хорошо», В = «студент сдал экзамен на отлично», тогда приведенное рассуждение описывается схемой

которая, как показано выше, является схемой правильного рассуждения.