

## И/ИЛИ-ДЕРЕВО КАК СРЕДСТВО АБСТРАКТНОГО ПРЕДСТАВЛЕНИЯ БАЗЫ ЗНАНИЙ

Григорьев А.В.

Кафедра ПМИИ, ДонГТУ  
grigorie@r5.dgtu.donetsk.ua

Каспаров А.А.

artemka08@ukr.net

### Abstract

*Grigoriev A.V., Kasparov A.A. AND/OR-tree as way of abstract presentation of knowledgebase. This article is about method of use And/OR-tree as the instrument of presentation of knowledgebase, when deciding a problem of designing an object.*

### Введение

При создании интеллектуальных систем (ИС) целесообразно использовать существующие технологии СУБД для хранения и доступа к данным, составляющим базу данных ИС. Таким путем можно обеспечить выполнение всех основных операций с данными в ИС. Данный подход позволяет возложить все задачи по обеспечению эффективной работы с данными на выбранную СУБД.

С другой стороны, для построения базы знаний (БЗ) интеллектуальной системы и работы с ней необходима некоторая абстрактная интерпретация набора фактов, представленных данными (отличная от модели данных, используемой СУБД). Данная интерпретация необходима для обеспечения работы алгоритмов, решающих типичные задачи интеллектуальной системы [1]. Среди таких интерпретаций можно назвать продукции, семантические сети, фреймы и т.п.

Метод И/ИЛИ-дерева может рассматриваться как один из наиболее простых методов абстрактной интерпретации набора фактов, представленных данными, т.е. методом представления знаний в ИС. В силу своей простоты И/ИЛИ-дерево имеет относительно узкую сферу применения в ИС. Собственно И/ИЛИ-дерево является классическим средством представления пространства решений в САПР, предназначенных для решения типичных задач проектирования [2,3].

Однако, возможно показать, что, вводя ряд новых положений, касающихся методов использования И/ИЛИ-деревьев как аппарата представления БЗ, можно получить мощный аппарат представления БЗ в ИС. В данном случае такая ИС будет предназначена для решения типичных задач проектирования, т.е. выбора из ряда возможных решений некоторого подмножества решений, удовлетворяющего заданным ограничениям.

Среди таких новых положений можно назвать:

- 1) задание БЗ как системы продукции над И/ИЛИ-деревом и организация логического вывода в БЗ;
- 2) разработка средств для автоматического пополнения такой БЗ в процессе обучения и т.п.

Целью данной статьи является иллюстрация основных положений использования И/ИЛИ-дерева в БЗ на примере конкретной ИС - программной реализации специализированной инструментальной оболочки, предназначенной для

автоматизации формирования интеллектуальных САПР, носящей название мета-эвристической оболочки. Таким образом, мы постараемся изложить конкретный подход к решению задачи, апробированный при этом на практике и показавший свою работоспособность.

В частности, предполагается изложить следующие вопросы: 1) общая интерпретация И/ИЛИ-дерева с точки зрения формальных систем представления знаний; 2) определение подкласса И/ИЛИ-деревьев, с которым предполагается работать; 3) описание метода формирования системы продукций над И/ИЛИ-деревом и процесса организации вывода.

### **1. И/ИЛИ-дерево как аппарат представления БЗ**

Опишем принятый в среде мета-эвристической оболочки метод представления знаний [4]. Основой данного метода является использование И/ИЛИ-дерева как аппарата представления БЗ.

Основой представления модели знаний является открытая формальная система, носящая наименование семиотической модели и представленной восьмеркой [5]

$$G = \{A, S, C, П, a, s, c, п\},$$

где  $A$  – аксиомы,  $S$  – множество синтаксически верных выражений,  $C$  – множество семантически верных выражений,  $П$  – множество правил вывода,  $a, s, c, п$  – соответственно правила изменения для  $A, S, C, П$ .

Базой знаний в системе является:

- И/ИЛИ-дерево, то есть упорядоченное множество вершин по «И» и по «ИЛИ», при этом все вершины имеют свой адрес в И/ИЛИ-дереве;
- Множество семантических двунаправленных зависимостей между ИЛИ-вершинами, описывающими их совместимость между собой, то есть множество правил вывода или продукций.

Представление данных в виде И/ИЛИ-дерева необходимо для выполнения теоретико-множественных операций над БЗ системы в процессе обучения и вывода. Поиск решений в системе ведется по И/ИЛИ-дереву, которое состоит из единиц информации, представленных некоторой структурой.

Под И/ИЛИ-деревом понимается некоторый связный граф, не содержащий циклов и имеющий иерархическую многоуровневую структуру. Вершинам И/ИЛИ-дерева может быть дана следующая трактовка, выполненная с точки зрения терминов теории формальных грамматик [6,7]. Любая вершина рассматривается либо как терм либо как синтерм. Терм определяется как элементарный терминальный символ множества и включает оригинальный фрагмент описания некоторого объекта(ов) на некотором языке. Термы соединяются между собой только посредством операции "И" (&). Синтерм, т.е. нетерминальный символ, задается как имя множества, которое может в дальнейшем раскладываться. Элементами разложения могут быть как термы, так и синтермы, соединенные посредством операции "И" (&) или "ИЛИ" (V). Синтермы всегда записываются в соответствии с нотацией Бекуса-Наура только в угловых скобках "<>". Очевидно, что термы будут являться только листьями дерева, которые, не могут иметь "сыновей" в данном дереве.

Для избранного подхода к представлению И/ИЛИ-деревьев справедливы следующие утверждения:

- Каждая вершина или узел дерева представляет собой описание фрагмента объекта;

- Каждая полная цепочка от вершины дерева до некоторого листа есть объект, состоящий из фрагментов, которые представлены узлами входящими в данную цепочку;
- Синтермы, имеющие одного «отца», являются альтернативами по отношению к друг другу;
- Отношения между ИЛИ связывает те термы, комбинация которых принадлежат некоторому непустому множеству семантически верных (проверенных) объектов, имеющих место в И/ИЛИ-дереве;
- Аксиомы или верифицированные объекты есть основа построения И/ИЛИ-дерева;
- И/ИЛИ-дерево есть средство:
  - компактной записи множества известных объектов;
  - порождения гипотез о возможных новых объектах;
- И/ИЛИ-дерево – это множество синтаксически правильных выражений;
- Продукции определены над И/ИЛИ-деревом и задают правила вывода, которые в совокупности позволяют вычленивать из И/ИЛИ-дерева семантически верное подмножество, то есть те же самые аксиомы-объекты.

## 2 Построение И/ИЛИ-дерева

Обозначим множество элементов И/ИЛИ дерева латинской литерой  $V$ . Тогда  $|V|$  - количество синтермов и термов в дереве. Разобьем  $V$  на два непересекающихся подмножества:  $V_{\vee}$ - подмножество «ИЛИ» синтермов и  $V_{\&}$ - подмножество «И» синтермов и термов. Справедливы следующие соотношения:

- $V_{\vee} \cup V_{\&} = V$ , то есть оба подмножества однозначно определяют И/ИЛИ дерево;
- $V_{\vee} \cap V_{\&} = \emptyset$ , то есть элементы каждого отдельного синтерма должны соединяться только по "И", или только по "ИЛИ".

Следует отметить, что множество термов, которые связываются между собой только по И, входит в подмножество  $V_{\&}$  ( $V_I \subseteq V_{\&}$ ).

Уровень И/ИЛИ дерева  $V_i$  - подмножество множества  $V$  такое, что синтермы данного множества являются посылками, а синтермы множества  $V_{i+1}$  выводами одной и той же группы продукций. То есть:

$$V_i \subset V, \sum V_i = V, \text{ где } 0 < i < k, k - \text{число уровней; } V_i \Rightarrow V_{i+1}.$$

Любой уровень И/ИЛИ дерева можно однозначно идентифицировать его номером, нумерацию удобно производить сверху вниз сопоставляя  $k=1$  корню дерева.

Можно выделить следующие свойства уровней:

- $\forall |V_m| < |V|$  т.е. мощность (количество элементов) любого уровня меньше мощности (количества элементов) всего дерева;
- Наибольшая возможная удельная мощность уровня  $|V_m| = |V| - 1$  при  $m=k=2$ .

Определим порядок нумерации синтермов в И/ИЛИ дереве таким образом, чтобы он был наиболее удобен для доступа к его элементам.

Номер отдельного синтерма  $S$  будем представлять в виде множества  $I$ , мощность которого  $|I|=5$ ;  $I = \{i_1, i_2, i_3, i_4, i_5\}$ , определим составляющие множества:

- $i_1$ - уровень,  $0 < i_1 < k$ , где  $k$  - количество уровней данного дерева.
- $i_2 = \{0, 1, 2\}$   $i_2 = 0$  если  $S_i \in V_{\&}$ ,  $i_2 = 1$  если  $S_i \in V_{\vee}$ ,  $i_2 = 2$  если  $S_i \in V_I$ .

- $i_3 = n$ , где  $n \in \mathbb{N}$  - номер синтерма в дереве при нумерации сверху-вниз и слева-направо.
- $i_4 = J$ , если данный синтерм является посылкой для некоторой продукции, выводом которой есть синтерм с  $i_3 = J$ , т.е.  $i_4$  - номер синтерма-вывода при нумерации сверху-вниз и слева-направо для данного синтерма-посылки. Для синтерма - корня дерева он равен нулю, т.е. не определен.
- $i_5 = l$ , где  $l \in \mathbb{N}$  - порядковый номер следования синтерма/терма (слева-направо), относительно синтермов/термов имеющих равные  $i_4$ .

Приведем пример И/ИЛИ дерева и нумерации его вершин на рис. 1.

Синтермы и термы дерева на данном рисунке задаются своими номерами:

$A_1 - S_{1,1,1,0,0}$ ,  $A_2 - S_{2,1,2,1,1}$ , ...  $A_3 - S_{2,1,5,1,2}$ , ...  $A_4 - S_{2,0,9,1,3}$ ,

$A_5 - S_{3,2,10,9,1}$ ,  $A_6 - S_{3,2,11,2,9,2}$ .

### 3. Процесс организации вывода

При организации процесса вывода удобно рассматривать БЗ как «кучу», т.е. совокупность всех синтермов, которые соотносятся с некоторыми аксиомами - прототипами и определены правила вывода в виде продукций. Куча это, в первую очередь, список всех синтермов по ИЛИ, то есть альтернатив в их исходном, полном виде, взятых из исходного И/ИЛИ-дерева ("полная" куча), т.е. множество  $V_v$ .

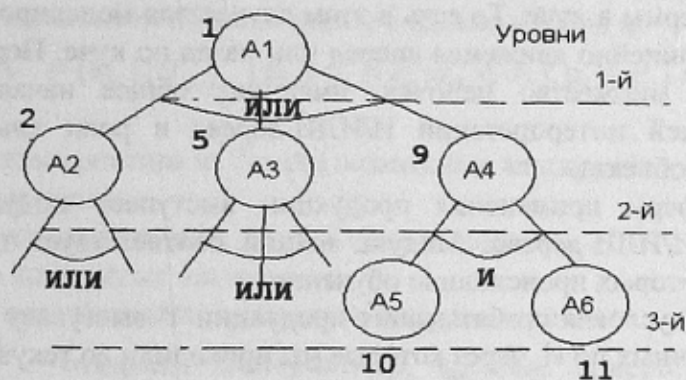


Рисунок 1 - Пример И/ИЛИ дерева

При этом:

- Все альтернативы имеют адреса, по которым можно однозначно и с максимальным быстродействием определить их положение в И/ИЛИ-дереве, адрес задается множеством  $I$ .
- По абсолютным адресам альтернатив в И/ИЛИ-дереве формируется их относительный порядок между собой, по  $i_5$ . Данный порядок предусматривает сортировку альтернатив таким образом, чтобы альтернативы, обладающие большей сложностью, имели больший порядковый номер  $i_5$ , то есть были расположены правее в И/ИЛИ-дереве.
- И-синтермы не входят в кучу, т.е.  $V_v \cap V_{\&} = \emptyset$ ;
- Куча формируется при начале вывода как копия всех существующих синтермов множества  $V_v$ .

В дальнейшем куча используется в процессе вывода, из ее элементов динамически строится ветвь вывода.

#### 4. Формирование системы продукций

Можно определить несколько алгоритмов формирования системы продукций по И/ИЛИ-дереву. Например, можно рассмотреть «парный» алгоритм, который предполагает выбор пары ИЛИ-синтермов А и В с последующим определением на них продукций. Можно так же определить две его вариации - «единичный» и «групповой» парный алгоритмы.

Ниже описан наиболее эффективный метод задания системы продукций, основанный на алгоритме «полных цепочек».

Данный алгоритм соответствует случаю, когда для того, что бы задать семантически верные объекты, мы заводим набор отношений как цепочку, связывающую отдельные альтернативы из множества возможных комбинаций альтернатив ИЛИ-синтермов, которые и описывают данный объект целиком. В этом случае просматриваются не пары синтермов по ИЛИ, а цепочка синтермов максимальной длины, т.е. рассматриваются одновременно все синтермы. Процесс состоит в движении по цепочке и удлинению ее имеющими смысл альтернативами из новых ИЛИ-синтермов. Данный подход представляет собой вариант формирования из исходного И/ИЛИ-дерева некоторого его поддерева. Но в этом поддерева любая комбинация альтернатив ИЛИ-синтермов задает семантически верный объект.

В этом случае могут применяться различные стратегии обхода И/ИЛИ-дерева. Например, слева-направо и сверху-вниз. В таком порядке уже заранее могут быть выстроены ИЛИ-синтермы в куче. То есть в этом случае для моделирования движения по дереву мы просто линейно движемся вперед или назад по куче. После обхода нами будет сформировано множество цепочек, имеющих общее начало. Необходимо заметить, что в нашей интерпретации И/ИЛИ-дерева в роли альтернатив будут выступать фрагменты объектов.

В качестве сферы применения продукции выступает модуль знаний, для которого построено И/ИЛИ-дерево. Модуль знаний соответствует типу или классу объектов, на основе которых происходило обучение.

В качестве предусловия срабатывания продукции Y выступает множество всех альтернатив объединенных по И, через которые мы проходили до текущего синтерма:

$$A^T = \{a^t_j\}, \quad (1)$$

где

$A^T$  – множество всех выбранных альтернатив;

T – количество пройденных шагов вывода;

$a_j$  – альтернатива с номером j, по сути фрагмент @j;

t – номер шага присоединения альтернативы.

В качестве посылки выступает альтернатива  $a_i^{T+1}$ , которая выбирается на данном этапе. В качестве ядра выступает логическое выражение:

$$a_i^{T+1} \rightarrow (A^T \cup \{a_i^{T+1}\}) \& (A^T \cup A^{\Pi}) \& (A - \{a_i^{T+1}\}) \& (A - A^Y) \& (P - P^Y). \quad (2)$$

Тут:

$a_i^{T+1}$  – альтернатива, выбираемая в данном синтерме;

$A^T$  – множество альтернатив выбранных до данного синтерма;

P – множество возможных семантически правильных объектов на данном шаге;

$P^Y$  – множество удаляемых на данном шаге объектов;

$A^{\Pi}$  – множество прибавляемых альтернатив по И;

$A$  – полное множество возможных альтернатив И/ИЛИ-дерева;

$A^y$  – множество удаляемых альтернатив.

Данная продукция трактуется следующим образом. Если выбрана альтернатива  $a_i^{T+1}$ , то:

- присоединить данную альтернативу к множеству пройденных альтернатив;
- удалить из множества возможных объектов те, которые не содержат  $a_i^{T+1}$ ;
- присоединить к множеству пройденных альтернатив, те которые связаны с  $a_i^{T+1}$  по И в дереве И/ИЛИ вывода;
- удалить альтернативу  $a_i^{T+1}$  из списка всех синтермов, которые еще могут рассматриваться;
- удалить все альтернативы присутствующие только в отбрасываемых из-за сужения ветвах, берущих начало в рассматриваемом синтерме.

Здесь:

$$P^y = \{p_y \mid \forall_{p_y \in P} (a_i \notin p_y)\}, \quad (3)$$

где  $P^y$  – множество тех объектов из потенциально возможных на данном синтерме, которые не содержат альтернативу  $a_i$ .

Раскроем принцип образования  $A''$  – множество альтернатив, которые содержатся в каждом возможном объекте вместе с альтернативой  $a_i$ :

$$A'' = \{a_k \mid (\forall_{p_l \ni a_k} (p_l \ni a_i)) \& (a_k \in A), p_l \in P, l = \overline{1, L}\}, \quad (4)$$

где

$a_k$  – некоторая альтернатива из списка возможных альтернатив;

$p_l$  – объект с номером  $l$ ;

$a_i$  – выбранная альтернатива на данном шаге вывода;

$A$  – множество возможных альтернатив;

$P$  – множество всех возможных объектов модуля знаний.

Множество  $A^y$  содержит альтернативы, которые присутствуют только в объектах множества  $P^y$ :

$$A^y = \{a_m \mid \forall_{a_m \in A} \neg \exists_{p_l \in P \setminus P^y} (a_m \in p_l)\}, \quad (5)$$

где

$A^y$  – множество удаляемых альтернатив;

$a_m$  – удаляемая альтернатива;

$A$  – множество возможных альтернатив;

$p_l$  – некоторый объект с номером  $l$ ;

$P$  – множество возможных семантически правильных объектов на данном шаге;

$P^y$  – множество удаляемых на данном шаге объектов.

### Заключение

В статье введен ряд новых положений, касающихся методов использования И/ИЛИ-деревьев в качестве средства представления знаний в ИС, решающих типичные

задачи проектирования. Предлагаемая интерпретация И/ИЛИ-дерева дает полнофункциональный эффективный аппарат представления БЗ в ИС.

Абстрактное представление БЗ в виде И/ИЛИ-дерева при решении задачи проектирования объекта позволяет эффективно и удобно решать следующий ряд задач:

- построения БЗ на имеющейся базе данных ИС, т.е. наборе описаний имеющихся верифицированных прототипов;
- выполнения теоретико-множественных операций над БЗ системы в процессе обучения и вывода;
- построения продукционной системы ИС, которая будет использоваться после окончания процесса обучения.

### **Литература**

1. Ф. Хейс-Рот, Д. Уотерман, Д. Ленат. Построение экспертных систем. - Москва: Мир, 1987.
2. Норенков И.П. Введение в автоматизированное проектирование технических устройств и систем. М.: Высш. шк., 1986. - 304 с.
3. Петренко А.И., Семенов О.И. Основы построения систем автоматизированного проектирования. - К.: ВШ, 1984. - 296 с.
4. Григорьев А.В. Семиотическая модель базы знаний САПР. Научные труды Донецкого государственного технического университета Серия: Проблемы моделирования и автоматизации проектирования динамических систем, выпуск 10: - Донецк, ДонГТУ, 1999. - С. 30-37.
5. Поспелов Д.А. Ситуационное управление: теория и практика. М.: Наука, 1986. - 288 с.
6. Ахо А., Ульман Дж. Теория синтаксического анализа, перевода и компиляции. В двух томах. Т.1. Синтаксический анализ. Пер. Агафонова В.Н.. Под. ред. Курочкина В.М. Мир, М.: 1978, с. 614.
7. М. Грос., А. Латен. Теория формальных грамматик. Мир, М.: 1971. с. 295.