МОСКОВСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ
ПРИБОРОСТРОЕНИЯ И ИНФОРМАТИКИ

Кафедра
“Персональные компьютеры и сети”

Ульянов М.В., Шептунов М.В.

МАТЕМАТИЧЕСКАЯ ЛОГИКА И ТЕОРИЯ АЛГОРИТМОВ

Часть I

МАТЕМАТИЧЕСКАЯ ЛОГИКА

Учебное пособие

Москва
2003

УДК 519.713

Ульянов М.В., Шептунов М.В. Математическая логика и теория алгоритмов, часть 1: Математическая логика. – М.: МГАПИ, 2003. – 47 с.

ISBN 5-8068-02 68 - X

Рекомендовано Ученым Советом МГАПИ в качестве учебного пособия для специальности 2201.

Рецензенты: к.т.н., проф. Зеленко Г.В.
к.т.н., проф. Рощин А.В.

Предлагаемое издание рекомендуется в качестве учебного пособия для подготовки студентов различных специальностей, изучающих математическую логику и теорию алгоритмов.

Для специальности 2201 «Вычислительные машины, комплексы, системы и сети» это издание может быть использовано в качестве учебного пособия по разделу «Математическая логика» дисциплины «Математическая логика и теория алгоритмов».

В первой части учебного пособия рассмотрены основы таких разделов математической логики как: исчисление высказываний, исчисление предикатов, модальная логика, немонотонные рассуждения и методы поиска в глубину и ширину, а так же элементы нечёткой логики.

$Л\frac{240\ 402\ 0000}{ЛР020418-97}$

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Логика – это наука о формализации мышления, задачей которой является изучение формальных законов построения и вывода суждений и доказательств. Логика использует формальный язык для описания и анализа суждений, формального доказательства утверждений. Логика исследует формальные схемы рассуждений, верные в силу одной их формы, независимо от содержания, что есть результат применения операции абстрагирования к рассуждениям естественного языка.

Современная парадигма научного исследования состоит в том, что формальное изучение любой проблемы начинается с замены реальных объектов их абстрактными представлениями, выбираемыми таким образом, чтобы в этих идеализациях были отражены именно те свойства исходных объектов, которые мы хотим изучать [4].

В настоящее время основанные на законах логики и теории алгоритмов специальные логические языки программирования, системы искусственного интеллекта и базы знаний получают всё большее распространение. В связи с этим в государственный стандарт введёна специальная дисциплина “Математическая логика и теория алгоритмов”. Описание государственного стандарта регламентирует ряд понятий и методов математической логики, которые и отражены в настоящем учебном пособии.

В учебном пособии рассмотрены основы таких разделов математической логики как: логика высказываний в классическом и конструктивном определении, исчисление предикатов, включая правила вывода и метод резолюций, модальная логика, немонотонные рассуждения и методы поиска в глубину и ширину, а так же основные понятия и определения нечёткой логики

1. СИСТЕМЫ СЧИСЛЕНИЯ

1.1 Понятие системы счисления

Любой современный человек неизбежно сталкивается с цифрами и числами. Многие ли задаются вопросом о том, в чём разница между этими двумя терминами? А при попытках дальнейших рассуждений могут возникнуть и вопросы относительно понятия “система счисления”.

Цифрами в математических дисциплинах принято называть символы, участвующие в записи числа. Под числом, следовательно, должна подразумеваться его величина, а не просто символьная запись.

Понятие числа в процессе развития цивилизации претерпевало значительные изменения. Одно из определений было дано И. Ньютоном: "Под числом мы понимаем не столько множество единиц, сколько отвлечённое отношение какой-нибудь величины к другой величине того же рода, принятой нами за единицу".

Что же тогда представляет собой система счисления?

Системы счисления, по-видимому, следует обобщённо рассматривать как разновидность информационных систем, созданные человеком. Целью создания любой системы счисления является выработка наиболее удобного способа записи чисел, в частности для простого и быстрого решения логических задач. Для “удобства” использования система счисления должна обладать следующими свойствами:

- простота способа записи на физическом носителе;
- удобство выполнения арифметических операций;
- наглядность обучения основам работы с числами.

Исторически системы счисления развивались от знаковых или символьных систем (римская) к позиционным (десятичная). Наиболее совершенными в свете упомянутых свойств являются позиционные системы счисления.

1.2 Позиционные системы счисления

Определение 1.1 *Позиционными* называются системы счисления, в которых вклад каждой цифры в величину числа зависит от её положения (позиции) в изображающей число последовательности цифр.

Определение 1.2 *Алфавитом системы счисления* называется совокупность различных цифр, используемых в позиционной системе счисления для записи чисел.

При рассмотрении позиционных систем счисления существенным является понятие базиса системы счисления.

Определение 1.3 *Базис* позиционной системы счисления - это последовательность чисел, каждое из которых задаёт “вес” каждого разряда.

В общем виде базис позиционной системы счисления можно записать в виде последовательных членов геометрической прогрессии:

$$\ldots,\ P^{-2},\ P^{-1},\ 1,\ P,\ P^{2},\ P^{3},\ \ldots,\ P^{3},\ \ldots,\ P^{n},\ \ldots\ .$$

Определение 1.4 Знаменатель P геометрической прогрессии, члены которой образуют базис позиционной системы счисления, называется *основанием* системы.

Относительно десятичной системы можно сказать следующее:

- основанием системы является число 10;
- базисом десятичной системы счисления являются члены геометрической прогрессии со знаменателем 10;
- алфавитом десятичной системы являются цифры: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

Существуют и реально используются другие P-ичные системы счисления, например, двоичная, троичная, восьмеричная, шестнадцатеричная.

Наряду с этим существуют другие системы счисления, построенные не на прямом позиционном принципе – базис системы счисления не является геометрической прогрессией.

1.3 Нетрадиционная фибоначчиева система счисления

Одной из нетрадиционных систем счисления является фибоначчиева система счисления.

Определение 1.5 *Базисом фибоначчиевой системы* называется последовательность чисел Фибоначчи**.**

Напомним, что числа Фибоначчи образуют последовательность, где каждый последующий член равен сумме двух предыдущих. Они возникают в самых разных математических задачах – комбинаторных, числовых, геометрических.

Первая десятка чисел Фибоначчи такова:

1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, …

Определение 1.6 *Алфавит фибоначчиевой системы* **-** это цифры 0 и 1.

Характерно, что в записи числа в фибоначчиевой системе не могут стоять две единицы подряд, т.к. это равносильно единице следующего разряда.

Можно показать, что при условии соблюдения этого правила представление чисел в данной системе счисления будет однозначно при кодировании слева, т.е. если каждый раз единицей будет отмечаться разряд (число Фибоначчи), с весом наиболее близким и меньшим текущего остатка числа.

Покажем, как записываются числа в такой системе счисления:

$37_{10}=34+3=100000100_{Ф}$;

$25_{10}=21+3+1=1000101_{Ф}$.

Системы, аналогичные фибоначчиевой, применяются при кодировании чисел и анализе алгоритмов.

Поскольку числа Фибоначчи асимптотически растут экспоненциально с основанием 1.618, то количество разрядов в записи числа в фибоначчиевой системе счисления будет больше, чем в двоичной.

1.4 Примеры перевода чисел из одной системы счисления в другую

1.4.1. Переводы чисел из десятичной системы

Опишем формально *алгоритм перевода целого числа а из десятичной системы счисления в Р-ичную:*

1) делим исходное число *a* на *P* нацело в десятичной системе счисления и записываем в качестве нового значения десятичного числа *a* целую часть результата от деления;

2) остаток от деления заменяем на соответствующую цифру в *P*-ичной системе счисления, и приписываем её слева к полученным ранее цифрам в *P*-ичной записи числа *a* (первая полученная цифра соответствует младшему разряду и её просто записывают);

3) выполняем пункты (1) и (2) до тех пор, пока *a* не станет равным нулю.

При переводе чисел может оказаться полезной следующая таблица представления чисел в наиболее часто используемых системах счисления:

Десятичные	Шестнадцатеричные	Двоичные
0	0	0 0 0 0
1	1	0 0 0 1
2	2	0 0 1 0
3	3	0 0 1 1
4	4	0 1 0 0
5	5	0 1 0 1
6	6	0 1 1 0
7	7	0 1 1 1
8	8	1 0 0 0
9	9	1 0 0 1
10	A	1 0 1 0
11	B	1 0 1 1
12	C	1 1 0 0
13	D	1 1 0 1
14	E	1 1 1 0
15	F	1 1 1 1

Таблица 1.1 Представление чисел в различных системах счисления

Пример 1.1 Перевести число 183_{10} из десятичной системы в двоичную систему счисления.

Воспользуемся вышеприведённым алгоритмом:

1) 183/2=91, ост. 1
2) 91/2=45, ост. 1
3) 45/2=22, ост. 1
4) 22/2=11, ост. 0
5) 11/2=5, ост. 1
6) 5/2=2, ост. 1
7) 2/2=1, ост. 0
8) 1/2=0, ост. 1

В итоге, получим: 183_{10}=10110111_2.

Пример 1.2 Перевести число 123_{10} из десятичной системы в шестнадцатеричную систему счисления.

123/16=7, ост. 11_{10}=B_{16}

7/16=0, ост. 7_{10}=7_{16}.

Таким образом :

123_{10}=$7B_{16}$.

1.4.2 Переводы чисел в десятичную систему счисления

Обратный *алгоритм перевода целых чисел из Р-ичной системы счисления в десятичную* следующий:

1) каждую цифру числа a в P-ичной системе счисления переводят в число в десятичной системе;

2) полученные числа нумеруют справа налево, по многочлену $a = a_n P^n + a_{n-1} P^{n-1} + ... + a_1 P + a_0$, начиная с нуля;

3) десятичное число, соответствующее каждой P-ичной цифре, умножают на P^k, где k – номер этого числа (см. п. 2), и результаты складывают, причём все эти арифметические действия проводят в десятичной системе.

Пример 1.3 Перевести число $B0F9_{16}$ из шестнадцатеричной в десятичную систему счисления.

$B0F9_{16}=[11_{10}][0][15_{10}][9]=11_{10}\cdot 16_{10}^3+15_{10}\cdot 16_{10}+9=45305_{10}$.

Здесь цифра B была заменена на десятичное число 11, а цифра F на 15 (см. табл. 1).

2. ЛОГИКА ВЫСКАЗЫВАНИЙ

2.1. Краткий экскурс в историю логики высказываний

Логика известна человечеству с древних времён и представляет собой искусство правильно рассуждать. Термин “логика” происходит от греч. ”logos”, что означает: слово, понятие, рассуждение.

Вообще говоря, способность к рассуждениям – это именно искусство. Имея в распоряжении какие-либо утверждения, истинность которых проверена, к примеру, экспериментально, логик может через умозрительные построения прийти к другому утверждению. Оно, в свою очередь, при искусстве правильно рассуждать, также оказывается истинным (но, вполне возможно, не во всех случаях).

Существует логика формальная, логика диалектическая, логика исследования и др. Данное учебное пособие посвящено формальной, а именно математической логике.

Первые учения о формах и способах рассуждений возникли в странах Древнего Востока: (Китай, Индия), но в основе современной логики лежат учения созданные в 4 веке до нашей эры древнегреческими мыслителями (главным образом Аристотелем). Аристотелю принадлежит исторически первое отделение логической формы речи от её содержания.

Он, в частности, открыл атрибутивную форму высказывания как утверждения или отрицания «чего-то о чем-то» и определил простое суждение (т.е. высказывание) как атрибутивное отношение двух терминов. Он также описал основные виды атрибутивных суждений и правильных способов их обращения.

Аристотель рассмотрел конкретные виды рассуждений, которые назвал силлогизмами. Более конкретно, он рассмотрел так называемые категорические утверждения следующих четырех видов:

- все A обладают свойством B (все A суть B);
- некоторые A обладают свойством B (некоторые A суть B);
- все A не обладают свойством B (все A суть не B);
- некоторые A не обладают свойством B (некоторые A суть не B).

Пример 2.1 Все люди смертны. Сократ – человек. Следовательно, Сократ смертен.

Пример 2.2 Все дикари раскрашивают свои лица. Некоторые современные женщины тоже раскрашивают свои лица. Следовательно, некоторые современные женщины – дикари.

Первое рассуждение правильно, оно подходит под один из образцов силлогизмов Аристотеля. Второе рассуждение неправильно, хотя все, входящие в него рассуждения истинны.

В начале 17 века Г. Галилей вводит в научный обиход понятие о гипотетико-дедуктивном методе: он восстанавливает права абстракции, обосновывает потребность в абстракциях, которые «восполняли» бы данные опытных наблюдений. Он указывает на необходимость введения этих абстракций в систему логических функций в качестве гипотез, с последующим сравнением результатов дедукции с результатами наблюдений.

Очевидным успехом движения за математическую логику явилось его признание на Втором Философском Конгрессе в Женеве в 1904г. Главным идейным противником применения математических методов к системе логических понятий был *психологизм* в логике. Психологизм в логике воспринимал математизацию логики как своего рода возрождение схоластики, которое менее всего было способно поставить логические исследования на научный фундамент. Борьба за математизацию логики и привела к мощному развитию этой науки.

Начиная с 1930-х годов, закладываются основы т. н. «машинного мышления» – теории алгоритмов. Её выдающиеся деятели: К. Гедель, С. Клини, А.

Тьюринг, А. Черч, Э. Пост, А. Марков, А. Колмогоров и др. И хотя была выяснена ограниченность такого мышления, что проявилось в установлении алгоритмической неразрешимости ряда логических проблем (знаменитая теорема Геделя о неполноте символических логик и обоснование алгоритмически неразрешимых задач), все же существенно вырос спрос на применение логики в вычислительной математике и технической кибернетике.

На сегодняшний день в многообразии логических теорий выражаются требования, предъявляемые логике современной наукой и практикой. Важнейшим из них является требование в содействии точной постановке и формулировке научно-технических задач и разысканию возможных путей их разрешения. Предлагая строгие методы анализа определенных аспектов рассуждений, логические теории одновременно содействуют и объективному анализу положения вещей в той области знания, которая находит отражение в соответствующих мыслительных процессах.

2.2. Логика и исчисление высказываний

Элементами логических рассуждений являются утверждения, которые либо истинны, либо ложны, но не то и другое вместе. Такие утверждения называются высказываниями (простыми). Простые высказывания обозначаются пропозициональными переменными, принимающими истинностные значения «И» и «Л». Из простых высказываний с помощью логических связок могут быть построены составные высказывания. Обычно рассматривают следующие логические связки:

- *отрицание* (читается “НЕ”, обозначается “$\neg$”),
- *конъюнкция* (читается “И”, обозначается “&”),
- *дизъюнкция* (читается “ИЛИ”, обозначается “V”),
- *импликация* (читается “ЕСЛИ… ТО, обозначается “$\to$”).

Примечательно, что тождества алгебры множеств могут быть переведены на язык математической логики и обратно – таблица 2:

Логическая терминология	**Язык теории множеств**
“*A* или *B*”	$A+B$
“*A* и *B*”	AB
“Не *A*”	A'
“Ни *A*, ни *B*”	$(A+B)'$, или, что то же, $A'B'$
“Не сразу *A* и *B*”	$(AB)'$, или, что то же, $A'+B'$
“Если *A*, то *B*”,	$A\subset B$
“Какое-то *A* есть *B*”	$AB\neq\varnothing$
“Никакое *A* не есть *B*”	$AB=\varnothing$
“Какое-то *A* не есть *B*”	$AB'\neq\varnothing$
“Нет никакого *A*”	$A=\varnothing$

Таблица 2.1 Соответствие языка теории множеств и математической логики

Кроме того, в терминах теории множеств силлогизм “если всякое *A* есть *B* и всякое *B* есть *C*, то всякое *A* есть *C*”, принимает простой вид:

если $A\subset B$ и $B\subset C$, то $A\subset C$.

Аналогично “закон противоречия”, утверждающий, что “объект не может одновременно обладать и не обладать некоторым свойством, записывается в виде: $AA'=\varnothing$, и “принцип исключённого третьего”, говорящий, что “объект должен или обладать, или не обладать некоторым свойством”, записывается как : $A+A'=U$.

Правильно построенные составные высказывания называются *формулами* (пропозициональными). Формулы имеют следующий синтаксис:

- <формула > :: = И | Л |
- <пропозиц. переменная > | (¬ <формула>) |
- (<формула>&<формула>)|
- (<формула>V<формула>)|
- (<формула>→ <формула>|

Истинностное значение формулы определяется через истинностные значения ("И" или "Л") её составляющих, в соответствии с таблицей 3:

A	B	$\neg A$	$A \,\&\, B$	$A \vee B$	$A \rightarrow B$
Л	Л	И	Л	Л	И
Л	И	И	Л	И	И
И	Л	Л	Л	И	Л
И	И	Л	И	И	И

Таблица 2.2 Истинностные значения логических формул

В логике высказываний вводится понятие вывода для формул: если некоторые формулы принять априори истинными (“аксиомы”), то на их основе можно по следующим правилам (“правилам вывода”) ввести дальнейшие формулы как истинные (“доказать”). Эта связь между формулами выражается специальным знаком $\vdash$, называемым *символом вывода*. Говорят, что *формула A выводима из множества* Н *формул*, и записывают:

$$\text{Н} \vdash A.$$

Если A выводима из пустого множества, то это выражается через $\vdash A$.

Определение 2.1. Конкретный набор истинностных значений, приписанных переменным $x_1,.., x_n$, называется *интерпретацией формулы A*.

Формула может быть истинной (иметь значение И) при одной интерпретации и ложной (иметь значение Л) при другой. Значение формулы A в интерпретации И будем обозначать I(A).

Определение 2.2. Формула, истинная при некоторой интерпретации, называется *выполнимой*.

Определение 2.3. Формула, истинная при всех возможных интерпретациях, называется *общезначимой* (или *тавтологией*).

Определение 2.4. Формула, ложная при всех возможных интерпретациях, называется *невыполнимой* (или *противоречием*).

2.3 Классическое определение исчисления высказываний

Исчисление высказываний – это формальная теория L, в которой определены следующие компоненты:

1. Алфавит:

- и $\rightarrow$ есть связки
- (,) есть служебные символы;
- $a, в, \ldots, a_1, в_1, \ldots$ есть пропозициональные переменные.

2. Формулы:

- переменные суть формулы;
- если A, B есть формулы, то $(\neg A)$ и $(A\rightarrow B)$ есть формулы.

3. Аксиомы:

- A_1: $(A\rightarrow(B\rightarrow A))$;
- A_2: $((A\rightarrow(B\rightarrow C))\rightarrow((A\rightarrow B)\rightarrow(A\rightarrow C)))$;
- A_3: $((\neg B\rightarrow \neg A)\rightarrow((\neg B\rightarrow A)\rightarrow B))$.

4. Правило: $\dfrac{A, A\rightarrow B}{B}$ Modus ponens (правило отделения).

Здесь A и B – любые формулы. Таким образом, множество аксиом теории L бесконечно, хотя задано 3-мя *схемами* аксиом. Множество правил вывода также бесконечно, хотя оно задано только одной схемой.

При записи формул лишние скобки опускаются, если это не вызывает недоразумений. Другие связки вводятся определениями (но не аксиомами):

- $A\&B:=\neg(A\rightarrow\neg B)$,
- $A\vee B:=\neg A\rightarrow B$.

Любая формула, содержащая эти связки, рассматривается как синтаксическое сокращение собственной формулы теории L.

2.4. Конструктивное определение исчисления высказываний.

Алфавит и множество формул те же, что и в подразделе 2.3,аксиомы представляют собой следующие три формулы:

- A_1: $(a\to(в\to a))$;
- A_1: $((a\to(в\to c))\to((a\to в)\to(a\to c)))$:
- A_1: $((\neg в\to\neg\, a)\to((\neg в\to a)\to в))$.

Правила вывода таковы:

Правило подстановки: если формула B является частным случаем формулы A, то B непосредственно выводима из A.

Правило Modus ponens (что означает «правило отделения»): если набор формул A, B, C является частным случаем набора формул a, $a\to в$, $в$, то формула C является непосредственно выводимой из формул A и B.

<u>Примечание</u>: здесь a, $a\to в$, $в$ – это три конкретные формулы, построенные с помощью переменных a, $в$ и связки $\to$.

Для вывода формул из заданного множества аксиом применяются следующие правила заключения или правила вывода (справедливые для любых формул A, B, C).

- $\vdash A \vee \neg A$ (Tertium non datur);
- $\{B, \neg B \vee B\} \vdash C$ (Modus ponens);
- Если $B = C$ есть применение правила семантической эквивалентности, булевской алгебры или булевского терма, то также справедливо $\{B\} \vdash C$ (применение закона равенства).

<u>Отступление</u>:

а) Принцип Termium non datur означает т. н. *принцип исключенного третьего*. Чтобы обосновать этот принцип, в качестве терма допускают только такие термы, для которых на основании их внешнего вида обеспечивается, что они при интерпретации обладают значениями «Истина» и «Ложь», но не значением $\perp$ («*дно*», что символизирует отсутствующий результат незавершающегося вычисления).

б) Modus ponens'у (см. правило 4 подраздела 2.3) соответствует правило заключения $\{x \to y, x\} \vdash$ y.

Заданная система правил вывода существенным образом применяет законы равенства булевской алгебры. Классически, в логике высказываний применяют другую систему вывода, в которой задается не закон равенства для высказываний, а специальные правила вывода для отдельных булевских операторов.

Определение 2.5 Пусть H – множество формул; тогда формулу A называют *непосредственно выводимой из* H, в этом случае пишут:

H$\vdash A$, если A выводима из H с помощью одного из правил вывода, т. е. существуют формулы $A_1 \dots A_n \in$ H и справедливо, что $\{A_1, \dots, A_n\} \vdash A$,что соответствует одному из правил (1-3).

Тем самым понятие выводимости монотонно в следующем смысле:

если $H_1 \subseteq H_2$ и имеет место $H_1 \vdash A_1$, то справедливо $H_2 \vdash A$.

Часто используется также запись $(A_1 \dots A_n)/A$ вместо $\{A_1, \dots, A_n\} \vdash A$.

Определение 2.6 Если H – множество формул и задана последовательность непосредственных выводов вида:

H $\vdash A_1$,

H $\cup \{A_1\} \vdash A_2$,

…

H $\cup$ $\{t_1, \dots, t_n\} \vdash A_{n+1}$,

то $A_1, \dots, A_{n+1}$ называют *выводом* над H, а высказывание A_{n+1} *–выводимым* из H (для чего также пишут H$\vdash A_{n+1}$).

Из заданного множества высказываний H могут быть выведены дальнейшие высказывания с помощью вышеприведенных правил вывода. Выводимые высказывания могут снова использоваться для вывода последующих высказываний.

Лемма 2.1 (Взаимосвязь между импликацией и выводимостью).

Если имеет место $H \vdash A_1 \rightarrow A_2$,

то справедливо также $H \cup \{A_1\} \vdash A_2$.

Понятие вывода формализует концепцию математических доказательств. Из заданного множества Н высказываний, которые принимаются за истинные, т. е. аксиом, выводятся дальнейшие высказывания по точно установленным правилам вывода.

Определение 2.7 Множество Н аксиом вместе с множеством правил вывода называется *формальной системой* или *теорией*, а выводимые формулы называются *теоремами*. Теория является *противоречивой*, если каждая формула является выводимой.

Теорема 2.1 (о противоречивости каждой теории логики высказываний для любого множества аксиом, которые содержат false).

Если в теории логики высказываний false выводимо, то выводимо любое высказывание A, т. е. Справедливо $\{\text{false}\} \vdash A$ для любой формулы A.

Теорема 2.2 (о корректности выводов).

Каждое (из пустого множества аксиом) выводимое высказывание A

со свободными идентификаторами x_1, …., x_n типа bool (где $n \in N$) для любой конкретизации И либо Л для X_1, …., X_n даёт значение И.

Теорема 2.3 (о полноте логики высказываний без атомарных высказываний).

Если формула логики высказываний A, которая не содержит никаких элементарных атомарных высказываний, со свободными идентификаторами x_1,..., x_n типа bool для каждой подстановки вместо x_1,…, x_n значений истинности определяет значение И («истина»), то формула A выводима.

Идея доказательства. Можно показать, что формула A только тогда определяет значение И для всех конкретизаций B, когда A сводима к true.

2.5 Другие аксиоматизации исчисления высказываний

Формальная теория L не является единственно возможной аксиоматизацией исчисления высказываний. Её основное достоинство – лаконичность при сохранении определённой наглядности.

Известны и многие другие аксиоматизации исчисления высказываний, предложенные различными авторами.

1. Гильберт и Аккерман, 1938
   - Связки: V, $\neg$, ($A \to B := \neg A \vee B$).
   - Аксиомы: $A \vee A \to A$,
   - $A \to A \vee B$,
   - $A \vee B \to B \vee A$,
   - $(B \to C) \to (A \vee B \to A \vee C)$.
   - Правило: Modus ponens.

2. Россер, 1953.
   - Связки: &, $\neg$, ($A \to B := \neg (A \& \neg B)$).
   - Аксиомы: $A \to A \& A$,
   - $A \& B \to A$,
   - $(A \to B) \to (\neg (B \& C) \to \neg (C \& A))$.
   - Правило: Modus ponens.

3. Клини, 1952.
   - Связки: $\neg$, &, V, $\to$.
   - Аксиомы: $A \to (B \to A)$,
   - $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$,
   - $A \& B \to A$,
   - $A \& B \to B$,
   - $A \to (B \to (A \& B))$,
   - $A \to (A \vee B)$,
   - $B \to (A \vee B)$,
   - $(A \to C) \to ((B \to C) \to ((A \vee B) \to C))$,
   - $(A \to B) \to ((A \to \neg B) \to \neg A$,
   - $\neg\neg A \to A$.
   - Правило: Modus Ponens.

3. ИСЧИСЛЕНИЕ ПРЕДИКАТОВ

3.1. Логика и исчисление предикатов

Рассмотренное ранее исчисление высказываний не всегда позволяет выразить многие факты и суждения, в том числе используемые в повседневной жизни. В целях обобщения указанного исчисления в его предложения обычно вводят параметры. Дадим объекту дальнейшего обсуждения, а именно исчислению предикатов, нестрогое определение и проанализируем некоторые предложения.

Определение 3.1 (нестрогое). *Предикат* – это функция одного либо нескольких аргументов с булевскими значениями истина и ложь.

Рассмотрим следующие два предложения:

1) “Все комплектующие, которые выпускает фирма IBM, стоят менее 100 долларов”;

2) “Фирма IBM выпускает винчестеры”.

Отсюда нужно вывести, что винчестеры фирмы IBM стоят менее 100 долларов. Однако пока этого нельзя сделать, поскольку текст второго предложения не входит непосредственно в первое, даже если попытаться чуть изменить словесные формулировки.

Попробуем применить исчисление предикатов. Введём в предложения параметры, поместив их в скобках в качестве аргументов (по аналогии с обычной записью для функций). Получим следующее:

IBM_ выпускает (HardDisc) $\rightarrow$ *стоит_менее (HardDisc, 100)*

IBM_ выпускает (HardDisc).

Здесь “*IBM_ выпускает*“ и “*стоит_менее*” – предикаты.

При этом следует отметить, что, например, утверждение

IBM_ выпускает (Cofe) имеет значение ложь, и так же ложное значение имеет *стоит_менее (HardDisc, 0).*

Поскольку номенклатура выпускаемых фирмой IBM изделий не ограничивается винчестерами, смысл полученной записи из двух утверждений относительно *HardDisc* должен быть расширен, что может быть проделано при помощи кванторов.

Введём переменную x и квантор общности $\forall$, тогда получим всего одно утверждение

$\forall$ *x IBM_ выпускает (x)* $\rightarrow$ *стоит_менее (x, 100).*

Поскольку в полученное утверждение входит квантор общности, означающий “для всех”, “для каждого”, то при формальном прочтении оно звучит так: “Для каждого x, если IBM поставляет x, этот x стоит менее 100 долларов.”

Квантор существования $\exists$ применяется, когда нужно показать, что существует хотя бы одно значение переменной, для которой истинно данное утверждение.

При использовании обоих упомянутых кванторов и введения ещё одной переменной y предыдущее утверждение запишется в виде

$\forall$ *x IBM_ выпускает (x)* $\rightarrow \exists$ *y (стоит (x, y)* $\wedge$ *менее (y, 100)),*

что означает: “Для любого x, если IBM поставляет x, обязательно найдётся y, такой, что x стоит y долларов и y менее 100”.

Итак, сформулируем строгое определение:

Определение 3.2 (строгое). *Предикатом* называется отображение прямого произведения заданных множеств в множество значений истинности

P: $M_1 \times M_2 \times ... M_n \rightarrow \{И, Л\}$,

где $M_1, M_2,.., M_n$ – заданные множества,

И, Л – символы для обозначения соответственно истины и лжи.

Характерно, что в классической логике предикатов рассматривают только такие элементарные высказывания, которые обладают значениями либо “Л”, либо “И”, но не каким-то третьим значением.

Определение 3.3 *Исчисление предикатов* (первого порядка) – это формальная теория K, в которой определены следующие компоненты:

1. Алфавит:
   - связки основные – , →
   - дополнительные & , V
   - служебные символы (,)
   - кванторы всеобщности ∀
   - существования ∃
   - предметные константы a, b...a1, b1,…
   - переменные x, y…x_1, y_1…
   - предикаты P, Q,…
   - функторы f, g,…

С каждым предикатом и функтором связано некоторое натуральное число, которое называется *арностью* или *местностью*.

2. Формулы имеют следующий синтаксис:
   - <формула>:: = <атом> |
   - ¬<формула> |
   - (<формула>→<формула>) |
   - ∀<переменная><формула> |
   - ∃<переменная><формула>
   - <атом>::=<предикат>(<список термов>)
   - <список термов>::=<терм> |<терм>,<список термов>
   - <терм>::=<константа> |
   - <переменная> |
   - <функтор>(<список термов>)

При этом должны быть выполнены следующие условия:
   - функтор f в терме должен быть *n*-местным;
   - предикат p в формуле также должен быть *n*-местным.

Формулы вида A и $\neg A$, называются *литеральными формулами* (или литералами).

В формулах ∀ x A и ∃ x A подформула A называется *областью действия квантора* по x.

Обычно связки и кванторы упорядочивают по приоритету следующим образом: ¬, ∀, ∃, &, V, →.

Лишние скобки при этом отсутствуют.

Исчисление предикатов, которое не содержит предметных констант, функторов, предикатов и собственных аксиом, называется *чистым*.

Исчисление предикатов, которые содержат предметные константы и/или функторы и/или предикаты и связывающие их собственные аксиомы, называются *прикладными*.

Исчисление предикатов, в котором кванторы могут связывать только предметные переменные, но не могут связывать функторы или предикаты, называется *исчислением первого порядка*.

Исчисления, в которых кванторы могут связывать не только предметные переменные, но и функторы, предикаты или иные множества объектов, называются *исчислениями высшего порядка*.

Все операции над значениями истинности можно обобщить до операций над множествами предикатов над заданным множеством M.

Если имеется предикат p над множеством M, то имеется непосредственная возможность превратить p в элементарное высказывание, задаваемое с помощью такой формы высказывания:

“Для всех $x \in M$ справедливо $p(x)$”

или

“По крайней мере, для одного $x \in M$ справедливо $p(x)$”.

Если первое из этих высказываний истинно, то предикат p называется *общезначимым*; если справедливо второе высказывание, то предикат p называется *выполнимым*.

Пусть t – есть формула с идентификаторами из семейства множеств идентификаторов X и пусть m – есть тип носителя M, который является множеством возможных значений для идентификатора $x \in M$, тогда:

квантор $\forall$ можно выразить через отрицание с помощью квантора $\exists$, и наоборот:

$$(\forall\ m\ x : t)=(\neg\exists\ m\ x : t),\ (\exists\ m\ x : t)=(\neg\forall\ m\ x : \neg t).$$

3.2. Правила вывода в логике предикатов первого порядка

Идентификаторы, такие как x в формуле логики предикатов $\forall$ m x : t, называются *связанными*.

Связанные идентификаторы могут быть переименованы без изменения терма логики предикатов. Имеет место следующий закон переименования:

$(\forall\ m\ x : t) = \forall\ m\ y : (t[y/x])$ – если y не входит в t как свободный идентификатор,

$(\exists\ m\ x : t) = \exists\ m\ y : (t[y/x])$ – если y не входит в t как свободный идентификатор.

Это означает, что формулы, получающиеся из других формул логики предикатов путем переименования связанных идентификаторов, семантически эквивалентны.

Правила подстановки для кванторизованных термов логики предикатов достаточно сложны. Подстановка в таком терме с кванторами описываются следующими равенствами, при условии если x и y – различные идентификаторы и x не является свободным в t`:

$(\forall\ m\ x : t)[t`/x] = \forall\ m\ x : t,$

$(\forall\ m\ x : t)[t`/y] = \forall\ m\ x : (t[t`/y])$

Если же x – свободный идентификатор в t`, то перед подстановкой связанного идентификатора $\forall$ m x : t с помощью приведенных выше законов переименования его необходимо переименовать, чтобы могла быть сделана подстановка $(\forall\ m\ x : t)[t`/y]$.

Справедливо следующее:

“И”, если для всех $\alpha \in M$ $(\alpha \neq \perp)$: $I_{B[\alpha/x]}[t]$ = “И”,

$I_B[\forall\ m\ x : t]$ = “Л”, в противном случае.

“И”, если для всех $\alpha \in M$ $(\alpha \neq \perp)$: $I_{B[\alpha/x]}[t]$ = “И”,

$I_B[\exists\ m\ x : t]$ = “Л”, в противном случае.

Теорема 3.1 (первая теорема Гёделя о неполноте).

Во всякой достаточно богатой теории 1-ого порядка существует такая истинная формула F, что ни F, ни $\neg$F не являются выводимыми в этой теории.

Теорема 3.2 (вторая теорема Гёделя о неполноте).

Во всякой достаточно богатой теории первого порядка формула F, утверждающая непротиворечивость этой теории, не является выводимой в ней.

Примечание. Вполне возможно, что непротиворечивость одной конкретной теории может быть установлена средствами другой, более мощной формальной теории. Но тогда возникает вопрос о непротиворечивости этой второй теории и т. д.

3.3. Метод резолюции для логики предикатов первого порядка

Метод (принцип) резолюций был предложен Ж. Эрбраном в 1930 г., а впервые реализован на ЭВМ Дж. Робинсоном в 1963 г.

В основе так называемого метода резолюций лежит идея «доказательства от противного».

Используемые в методе резолюций так называемые *резольвенты* представляют собой вершины дерева вывода, соответствующие выводимым дизъюнктам.

Определение 3.4 *Резолютивный вывод* Ф из множества дизъюнктов S

есть такая конечная последовательность $Ф_1,\ldots, Ф_k$ дизъюнктов, что $Ф_k$=Ф и каждый дизъюнкт $Ф_i$ или принадлежит S, или является резольвентой дизъюнктов, предшествующих $Ф_i$.

Определение 3.5 *Универсальным замыканием* формулы Ф ($x_1,\ldots, x_n$) называется предложение $\forall x_1\ldots\forall x_n$ Ф ($x_1,\ldots, x_n$).

Теорема 3.3 (о полноте метода резолюций). Если S – множество дизъюнктов, то множество универсальных замыканий формул из S невыполнимо тогда и только тогда, когда существует резолютивный вывод 0 из S.

В качестве противоречия F при доказательстве от противного по методу резолюций может быть использована пустая формула, обозначаемая □. Пустая формула по определению является противоречием, не имеет никакого значения и ни в какой интерпретации не является истинной.

Рассмотрим метод резолюций применительно к исчислению предикатов. Пусть C_1 и C_2 два предложения в исчислении предикатов. Правило вывода

$$\frac{C_1, C_2}{(C'_1 \vee C'_2)\sigma} R$$

называется *методом резолюций в исчислении предикатов*, если в предложениях C_1 и C_2 унифицируемые литералы P_1 и P_2, т. е. $C_1=P_1VC`_1$ и $C_2=\neg P_2VC`_2$.

В указанной форме записи метода резолюций предложения C_1 и C_2 являются *резольвируемыми*, а предложение $(C'_1 \vee C'_2)\sigma$, полученное из предложения $(C'_1 \vee C'_2)$ применением унификатора σ, *резольвентой*.

Метод резолюций служит основой языков логического программирования, главным отличием которых от так называемых "процедурных" языков является то, что программа не указывает, как что-либо следует делать для решения задачи, а описывает некоторые элементы и связи между ними и ставит конкретную цель. При этом компьютер самостоятельно ищет стратегию для решения поставленных вопросов.

Сформулируем основные положения метода резолюций.

1. Модель исследуемого "мира" представляется множеством аксиом, которые преобразуются в множество дизъюнктов S.

2. Для доказательства справедливости теорем в данном "мире" необходимо взять её отрицание и, преобразовав в форму дизъюнкнта, добавить к

множеству S. Если теорема верна, то новое множество дизъюнктов должно быть противоречиво.

3. Доказательство противоречивости сводится к доказательству того, что из данного множества дизъюнктов может быть выведен пустой дизъюнкт.

4. В техническом аспекте метод резолюций состоит из унификаций и получения множества резольвент до тех пор, пока не будет получена пустая резольвента.

5. Для уменьшения числа резольвент (и следовательно, для повышения эффективности вывода) очень существенна стратегия вывода.

6. Если множество дизъюнктов S противоречиво, то пустой дизъюнкт будет найден за конечное число шагов. При непротиворечивости множества дизъюнктов S процесс установления факта непротиворечивости может быть бесконечным.

4. МОДАЛЬНАЯ ЛОГИКА

4.1. Основные понятия модальной логики

Рассмотренные нами ранее суждения классической логики (как простые, так и сложные) подразумевали лишь факт их истинности либо ложности. Для моделирования многих технических задач такой формализм является неприемлемым. Поэтому были введены неклассические логики, включающие в себя так называемые ***модальные логики***, которые подразделяются на *временную*, *динамическую* и другие логики.

Название “модальная логика” связано с тем, что в модальные логические системы входят такие операторы над логическими формулами, которые позволяют модифицировать их интерпретацию.

Определение 4.1. *Модальная логика* – это область логики, посвящённая изучению модальностей, построению исчислений, в которых модальности применяются к высказываниям, наряду с логическими операциями, и сравнительному исследованию таких исчислений.

Модальные операции, такие как, ‘‘возможно’’, “необходимо”, и другие, могут относится как к высказываниям или предикатам, так и к словам, выражающим какие-либо действия или поступки.

В системах модальной логики, для которых справедливы принцип исключённого третьего и закон снятия двойного отрицания $\neg\neg A \supset A$, для операторов возможности $\Diamond$ и необходимости $\Box$ справедливы соотношения двойственности:

- $A \equiv \neg\Diamond\neg A$
- $A \equiv \neg\Box\neg A$,

вполне аналогичные законам де Моргана алгебры логики:

- $(A \vee B) \equiv (\neg A \& \neg B)$
- $(A \& B) \equiv (\neg A \vee \neg B)$.

Эта аналогичность сохраняется также для соответствующих соотношений логики предикатов для кванторов, связывающих операторы возможности ◊ и необходимости □ с отрицанием ⌉, а именно:

- $A \equiv \rceil \Diamond \rceil A$
- $A \equiv \rceil \Box \rceil A.$

В аксиоматических системах модальной логики в качестве исходной обычно вводят одну модальную операцию (используя какую-либо из этих эквивалентностей в качестве определения другой операции). Аналогично вводятся и другие модальные операции (не входящие в число логических операций и не выражаемых через них).

Системы модальной логики могут быть интерпретированы в терминах многозначной логики (простейшими системами модальной логики являются трёхзначные: " истина", "ложь", "возможно"). Это обстоятельство, а также возможность применения многозначной логики к построению теории т. н. “правдоподобных” выводов указывают на её глубокое родство с вероятностной логикой.

Помимо “абсолютных” модальностей, в модальной логике приходится иметь дело с т. н. “относительными”, т. е. связанными с каким-либо условием (“А возможно, если В”, и т. п.). Понятия всякого рода относительных модальностей удаётся легко формализовать, дополняя аппарат модальной логики аппаратом логики предикатов.

Введём три понятия, поясняющие связь между операторами необходимости и возможности.

Два суждения **противоречивы**, если они несовместимы ни по истинности, ни по ложности.

Контрарными являются суждения, совместимые по ложности, но несовместимые по истинности.

Два суждения находятся в отношении **подчинения**, лишь, если из первого следует второе, но из второго не следует первое.

Модальные формулы всегда можно записать посредством оператора $\Box$, поскольку справедливо:

$$\Diamond A \leftrightarrow \neg \Box \neg A.$$

Приведём одну важную зависимость, также выражаемую через оператор $\Box$:

$$\nabla A \equiv \Diamond A \wedge \Diamond \neg A,$$

что читается следующим образом: высказывание “случайно *A*” эквивалентно высказыванию “возможно *A* и возможно не-*A*”.

4.2. Синтаксис и семантика модальной логики

Определение 4.2 *Структурой* называется пара $F=(W, R)$, где W – непустое множество, а R – бинарное отношение на множестве W. Элементы такого множества называются *точками*.

Определение 4.3 Пусть P – множество высказываний модального языка L. Тогда *P-моделью* на структуре (W, R) называется тройка $M=(W, R, V)$, причём $V(p)$ интерпретируется как множество точек из W, в которых высказывание p истинно.

Пусть $w \in W$ и A – модальная формула. Запись $M \models_w p$ означает, что A истинна в точке w модели M. Тогда можно сформулировать следующие правила:

- $M \models_w p$, если $w \in V(p)$;
- $M \models_w A_1 \rightarrow A_2$, если из $M \models_w A_1$ следует $M \models_w A_2$;
- $M \models_w \neg A$, если $M \not\models_w A$;
- $M \models_w \Diamond A$, если $M \models_t A$ хотя бы для одного $t \in W$, такого, что wRt;
- $M \models_w \Box A$, если из wRt следует $M \models_t A$ для $\forall t \in W$.

Последнее правило означает, что формула $\Box$A истинна в точке w модели M, если она истинна во всех точках t, которые находятся в отношении R с точкой w.

Сформулируем теперь понятие истинной формулы применительно к модели, структуре и общезначимой формуле.

Формула A истинна в модели M, если она истинна во всех точках этой модели, т. е. если $M|=_w A$ для всех $w \in W$, что обозначается как $M = {}_w A$.

Формула A истинна в структуре F=(W, R), если она истинна в любой модели (W, R, V), т. е. если $M|=_w A$ для всех моделей M=(W, R, V). Это обозначается как F |= A.

Формула общезначима, если она истинна во всех структурах F=(W, R). Это обозначается |=A.

4.3 Схемы модальных формул

Определение 4.4 Модальная логика называется *нормальной*, если она содержит:

- множество всех теорем логики высказываний, область действия которых распространяется на формулы модальной логики высказываний;
- схему аксиомы дистрибутивности:K: $\Box(A \to B) \to (\Box A \to \Box B)$, назовём её схемой K, что читается следующим образом: “если необходимо, что A влечёт B, то из необходимости A следует необходимость B”;
- модальные правила выводимости: $A \vdash \Box A$, т. е. A необходимо истинна при условии истинности A.

Классическими обозначениями для некоторых схем являются следующие:

- D: $\Box A \to \Diamond A$,
- T: $\Box A \to A$,
- $\Box A \to \Box\Box A$, (*)
- B: $A \to \Box\Diamond A$,

- $\Diamond A \to \Box\Diamond A$, (**)
- L: $\Box((A \wedge \Box A) \to B) \vee \Box((B \wedge \Box B) \to A)$,
- W: $\Box(\Box A \to A) \to \Box A$.

Приведём словесные названия некоторых из схем. Схему T называют *схемой аксиомы знания*. Схему (*) называют *схемой аксиомы положительной интроспекции*. Схема (**) называется *аксиомой негативной интроспекции*.

Необходимо отметить, что выбор модальной сжемы зависит от моделируемого понятия.

4.3. Бинарные отношения и семантика возможных миров

Наряду с алгебраическими семантиками в модальной логике построены так называемые семантики возможных миров.

Основным понятием в этих семантиках является понятие возможного мира. Под возможным миром понимается существующее или мыслимое положение дел либо возможный ход событий. Поэтому, когда речь идёт о достижимости из одного мира другого, под этим следует понимать некоторое отношение, переводящее одно состояние в другое.

Вспомним некоторые свойства, которыми может обладать бинарное отношение.

- Рефлексивность: $\forall s\ (sRs)$
- Транзитивность: $\forall s \forall t \forall u\ (sRt \wedge tRu \to sRu)$
- Симметричность: $\forall s \forall t\ (sRt \to tRs)$
- Евклидовость: $\forall s \forall t \forall u\ (sRt \wedge sRu \to tRu)$

Определим понятие возможных миров и их семантики. Точки $w \in W$ назовём *возможными мирами универсума*, а отношение R - *отношением достижимости*. Таким образом, структура состоит из множества W возможных миров, связанных отношением R. Если два мира s и t принадлежат универсуму W, то достижимость мира t из мира s обозначается sRt.

Приведённые определения целесообразно представить следующим рисунком (рис. 4.1).

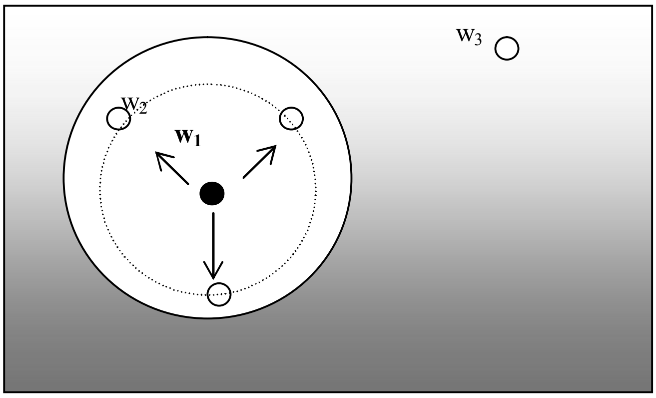

Рис. 4.1 Сфера миров, достижимых из w_1

Если представить универсум W возможных миров, то для каждого мира $w_i \in W$ множество достижимых из w_i миров можно изобразить в виде сферы миров. В частности, на этом рисунке мир w_2 достижим из мира w_1. Истинные формулы в w_2 возможно истинны в w_1. Однако мир w_3 недостижим, т. е. невозможен с позиций w_1. Разнообразные трактовки модальных формул в разных модальных логиках могут приводить к изменению форм отношений достижимости, определяемых через R.

4.5. Обзор других формально-логических моделей

4.5.1. Логика возможного

Определение 4.5 *Необходимо истинной в мире S* называется формула, которая подтверждается во всех мирах t, достижимых из S.

Формула вида $\Box A$ читается «A необходимо истинна».

Определение 4.6. *Возможно истинной в мире S* называется формула, которая подтверждается хотя бы в одном из миров, достижимых из S.

Чтение формулы $\Diamond A$ следующие: «возможно, что A истинна».

В универсуме один из миров можно было бы рассматривать в качестве «реального». Формула A, необходимо истинная в реальном мире, истинна и во всех мирах, достижимых из «реального мира», а, следовательно, и в самом реальном мире (свойство рефлективности).

Таким образом, формула вида $A\Box\rightarrow A$ истинна в рассматриваемой структуре.

Модальные понятия делятся на логические и физические (фактические). Например, положение дел может быть логически необходимо, логически возможно либо физически возможно, логически случайно либо физически случайно.

Логически возможно то, что не противоречит законам логики. Поэтому не всё то, что логически возможно, возможно физически.

Физически возможно то, что не противоречит законам физики, природы и общественным законам.

Логически необходимо то, что является законом логики.

Физически необходимы законы физики, природы и логические следствия из них.

Например, если за V обозначить скорость света, то формула $\Box$ (V<300000 км/с) истинна в реальном мире при условии, что $\Box V$ – физическая необходимость. На самом деле полагают, что законы физики подтверждаются во всех мирах, достижимых из реального мира. Однако логически возможно, чтобы указанная формула была ложной.

4.5.2. Временные логики

Отношение достижимости, обозначение sRt, во временных логиках означает, что t следует за S. Следовательно, во временной логике возможные миры представляют состояния некоторого мира в различные моменты его эволюции. Интерпретация формулы $\Box A$ может быть такой: «во все будущие моменты произойдёт A», а формулы $\Diamond A$ такой: «по крайней мере в один из будущих моментов произойдёт A».

Если же sRt определено как «t предшествует S», то формулы $\Box A$ и $\Diamond A$ соответственно могут читаться так: «во все прошлые моменты было A» и «по крайней мере, в один из моментов было A».

Для временной логики естественными являются структуры (W, R), у которых W – это одно из множеств, а R – это отношение порядка ($<, \leq, \geq, >$).

4.5.3. Динамические логики

Динамическая логика основана на сопоставлении некоторого модального оператора каждой команде какого-то языка программирования. В этом контексте множество W интерпретируется как совокупность возможных состояний в ходе вычислений. Отношение sRt определяет, что вычисления начинаются в момент S и заканчиваются в момент t. Тогда недетерминированная программа может приводить к многочисленным результатам. В этом случае формула $\Box A$ будет означать «все выполнения программы оканчиваются тем, что A истинна», а формула $\Diamond A$ – «хотя бы одно выполнение программы заканчивается тем, что A истинна».

4.5.4. Логики веры и знания

Основной функцией модальной логики является формализация модальностей «необходимости» и «возможности». Другое направление преимущественного применения – это моделирование и анализ концептуальных категорий «знания» и «веры». При использовании в различных формальных языках операторов «веры» и «знания» оператор $\Box$ принимаетет соответственно значения «предполагается» и «известно», а оператор $\Diamond$ - соответственно «противоположное не предполагается» и «противоположное неизвестно».

Различные модальные логики являются расширениями классической логики первого порядка. Они заимствуют из неё аксиомы, правила, теоремы.

5. НЕМОНОТОННЫЕ РАССУЖДЕНИЯ И МЕТОДЫ ПОИСКА

5.1. Модифицируемые рассуждения и свойства немонотонных логик

Важной чертой нерационального рассуждения является способность вырабатывать здравые суждения, хотя они могут быть и недостоверными. В частности, при неполной, неточной и изменчивой информации рассуждения зачастую бывают предположительными, всего лишь правдоподобными и, следовательно, подлежащими пересмотру, то есть модификации.

Например, используя непрофессиональное, но распространенное представление о понятии «машины» как о некоторой системе, способной перемещаться в пространстве и зная, что «Кибер» - это машина, можно сделать вывод о возможности перемещения «Кибера». Такой вывод можно считать приемлемым. Однако его нельзя признать абсолютно корректным, так как не учитываются возможные использования этого термина в другой предметной области, где свойство перемещения может быть нехарактерным для машины. Если же будет сделано уточнение, что «Кибер» - это ЭВМ, то принятое рассуждение должно быть пересмотрено, путём опровержения допущения о возможности перемещения «Кибера».

Несмотря на простоту приведённого примера, его трудно формализовать в рамках классической логики.

Природа модифицируемых рассуждений базируется на двух фундаментальных разновидностях:

Рассуждения являются модифицируемыми, когда они являются предположительными и, следовательно, правдоподобными. Такие рассуждения неточны, так как зависят от неполной, неточной или изменчивой информации.

Рассуждения модифицируемы, когда они зависят от знаний, предполагаемых полными, но таковыми не являющихся либо перестающих быть таковыми.

Немонотонный вывод характеризуется тем, что множество теорем необязательно растёт с множеством аксиом. Модифицируемые рассуждения не являются в классическом смысле общезначимыми. Этому можно дать следующее пояснение: вывести p из множества посылок A и отказ от p, как только к A добавлена новая информация q, означает, что допустимо вывести p. Свойства монотонности и, как следствие, транзитивности, здесь не удовлетворяются.

Неадекватность систем дедукции классической логики для формализации модифицируемых рассуждений объясняется тем, что их правила являются позволяющими. Они имеют вид: q – теорема, если $p_1, p_2, \ldots, p_n$ – теоремы.

С одной стороны, моделирующая модифицируемые рассуждения система может обладать ограничивающими правилами вывода: q - теорема, если $p_1, p_2, \ldots, p_n$ – теоремы.

Таким образом, с помощью немонотонных логик предполагается моделирование общезначимых модифицируемых рассуждений.

5.2. Зацикливание немонотонных рассуждений и его преодоление

Существенным обстоятельством, усложняющим осуществление систем немонотонного вывода, является возможность зацикливания рассуждений.

Это объясняется необходимостью модификации правил вывода, так как немонотонная система должна обеспечивать возможность отказа от ранее сделанных выводов. С этой целью правила оснащаются условиями применения, проверка которых динамически изменяется с множеством посылок. Эти условия до вывода проверяют выполняемость некоторого утверждения вместе с выведенными ранее утверждениями из существующей системы посылок. Проблема здесь состоит в соотнесении выводимости и выполняемости.

Необходимо также отметить, что отдельные заключения могут быть получены лишь после того, как будет установлена невыводимость каких-то других результатов.

Практическая реализация алгоритмов модифицируемых рассуждений достаточно сложна.

В настоящее время применяются три семейства немонотонных логик.

I. Логики умолчаний.

Эти логики часто называют логиками типичного. Они введены и развиты Рейтором для формализации рассуждений, являющихся всего лишь выполнеными. Следовательно, немонотонность в них обусловлена необщезначимостью правил вывода. При неполной информации мы можем получить только правдоподобные предположительные заключения. Часто бывает, что правила истинные в большинстве случаев, мы принимаем как абсолютно общие, хотя в них и допускаются исключения.

Логики умолчаний позволяют формировать рассуждения в виде правил вида:

$$\frac{\alpha : M\beta}{\gamma}.$$

Смысл состоит в следующем. Если мы допускаем истинность α и если β выполнимо при сделанных допущениях α, то можно допустить и истинность γ.

В приведённом правиле символ M – метасимвол, отражающий возможность (выполнимость) некоторого факта. Здесь следует обратить внимание на то, что в связи с возникающими алгоритмическими и вычислительными трудностями, по данному направлению неизвестны какие-либо значительные практические реализации.

II. Модальные немонотонные логики Мак-Дермотта.

Они основаны на аксиоматических системах модальной логики и предназначены для характеризации множеств взаимно выполняемых утверждений, которые можно вывести из какого-то задаваемого множества посылок.

Немонотонные логики Мак-Дермотта предлагают независимую от области применения (и в этом смысле универсальную) аксиоматическую систему оценки выполняемых множеств утверждений. Для формализации модифицируемых рассуждений используется модальная логика. Кроме того, в рамках этой логики предусмотрена возможность устранения зацикливаний при задании правил немонотонного вывода. Здесь имеет место ситуация, когда опровергнутый ранее вывод, на следующем шаге при поступлении дополнительной информации множеств вновь оказаться непротиворечивым. Поскольку история предыдущих доказательств не сохраняется, по существу доказательство уже пройденного пути начинается с самого начала. Таким образом, можно сделать вывод о невысокой эффективности таких систем.

III. Автоэпистемические логики.

Они позволяют осуществлять формализацию уравнения вида: «если я не предполагаю, что p подтвердится, то подтверждается q».

Под идеально разумными понимаются рассуждения, содержащие следующие идеализации: можно выводить только ожидаемые идеологические следствия из исходного множества предположений и все эти логические следствия надо принять во внимание. Это важное обстоятельство ставит под сомнение реализуемость задач, основанных на этих логиках, поскольку для рассуждений нужны неограниченные ресурсы.

5.3. Стратегии немонотонного вывода в глубину и ширину

Для задач, сформулированных в терминах пространства состояний, существуют различные подходы к проблеме поиска решающего пути.

Пространство состояний – это граф, вершины которого соответствуют ситуациям, встречающимся в задаче, а решение задачи сводится к поиску пути на графе.

Вершины графа соответствуют проблемным ситуациям, а дуги – разрешённым переходам из одних состояний в другие. Отыскание плана решения задачи эквивалентно построению пути между заданной начальной ситуацией и некоторой указанной заранее конечной ситуацией, которая называется целевой вершиной.

Основными стратегиями поиска являются поиск в глубину и поиск в ширину. Для обеих стратегий допустимы следующие допущения:

- процесс вывода заключения интерпретируется как дедуктивный процесс доказательства теории;
- логический вывод по существу сводится к достижению цели G на основе последовательного процесса доказательства истинности (или ложности) частичных целей.

Проход в глубину (рис. 5.1) образует путь, который может и не достигать цели, поэтому должен быть предусмотрен механизм возврата для поиска альтернативы, вновь ведущей вглубь. При этом история альтернатив должна быть сохранена для того, чтобы при повторном поиске в глубину «не проходить» старый безуспешный путь.

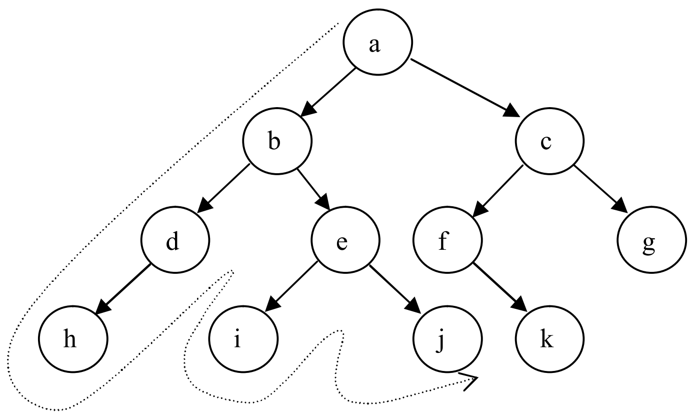

Рис. 5.1. Поиск в глубину

Другой подход к построению системы немонотонных рассуждений основан на использовании стратегии поиска в ширину. Принцип стратегии поиска в ширину заключается в следующем.

В противоположность поиску в глубину стратегия поиска в ширину предусматривает переход в первую очередь к вершинам, ближайшим к стартовой вершине.

В результате процесс поиска имеет тенденцию развиваться более в ширину, чем в глубину, что проиллюстрировано на рис. 5.3.

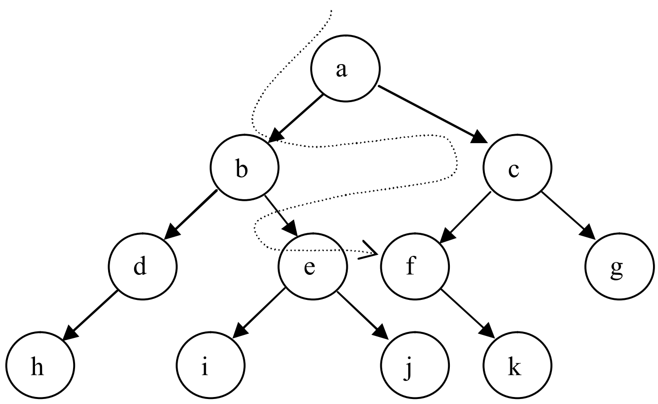

Рис. 3. Поиск в ширину

Поиск в ширину реализуется алгоритмически не так легко, как поиск в глубину. Причина этого в том, что приходится сохранять всё множество альтернативных вершин-кандидатов, а не только одну вершину, как при поиске в глубину. Кроме того, если в процессе поиска необходимо получить решающий путь, то одного множества вершин не достаточно. Поэтому хранят множество путей-кандидатов. Для того, чтобы выполнить поиск в ширину при заданном множестве путей-кандидатов, необходимо: если голова первого пути – это целевая вершина, то взять путь в качестве решения, в противном случае удалить первый путь из множества кандидатов и породить множество всех возможных продолжений этого пути на один шаг, множество продолжений добавить в конец множества кандидатов, а затем выполнить поиск в ширину с полученным новым множеством.

6. ЭЛЕМЕНТЫ НЕЧЁТКОЙ ЛОГИКИ

6.1. Основные понятия и определения

Определение 6.1 Множества объектов, обладающие нечёткими свойствами, называются *нечёткими* или расплывчатыми.

Определение 6.2 Пусть X – произвольное непустое множество. Множество $\overline{A}$ называется *нечётким* множеством в множестве X, если каждый элемент множества $\overline{A}$ есть пара, на первом месте которой стоит значение функции принадлежности $M_{\overline{A}}$: X→[0, 1], а на втором – элемент x∈X, для которого определена эта функция.

Другими словами, при задании множества $\overline{A}$ каждому x∈X приписывается число $0 \le M_{\overline{A}}(x) \le 1$, определяющее степень принадлежности этого элемента множеству $\overline{A}$.

Определение 6.3 *Носителем нечёткого множества* $\overline{A}$ называется подмножество A множества X, для которого значения функции принадлежности $M_{\overline{A}}(x)$ больше нуля.

Определение 6.4 Нечёткое множество $\overline{A}$ в X называется *пустым* ∅, если для всех x∈X величина $M_{\overline{A}}(x)=0$.

Определение 6.5 *Нечётким высказыванием* называется предложение, относительно которого можно судить о его истинности или ложности в настоящий момент.

Таким образом, степень истинности или степень ложности нечёткого высказывания принимает значения из интервала [0, 1].

Нечёткое высказывание, имеющее значение степени истинности, равное 0.5, назовём индифферентностью, поскольку оно в той же мере истинно, в коей и ложно.

Например, нечёткими будут такие высказывания:

«2 – маленькое число», «Студент Иванов не опаздывает на занятия», «Москвич – хороший автомобиль».

6.2. Нечёткие логические формулы

Под нечёткой высказывательной переменной $\overline{x}_i$ будем понимать нечёткое высказывание, степень истинности которого может принимать значения из [0, 1].

Определение 6.6 *Нечёткой логической формулой* $\overline{A}(\overline{x}_1, \overline{x}_2, ..., \overline{x}_n)$ где $n \geq 1$, называется: любая нечёткая переменная или константа из [0, 1]; или выражение $\overline{A}(\overline{x}_1, \overline{x}_2, ..., \overline{x}_n)$, полученное из нечётких логических формул $\overline{A_1}(\overline{x}_1, \overline{x}_2, ..., \overline{x}_n)$ и $\overline{A_2}(\overline{x}_1, \overline{x}_2, ..., \overline{x}_n)$ применением к ним любого конечного числа логических операций.

Для задания нечётких логических формул используются аналитические функции, поскольку таблицы степеней истинности невозможны из-за их бесконечности.

Важно подчеркнуть, что нечёткие логические формулы, имеющие на одних и тех же наборах одинаковые степени истинности, не равносильны, а имеют некоторую степень равносильности. Эта степень обычно больше либо равна 0,5, но всегда меньше либо равна единице.

Поэтому для одних и тех же формул можно говорить о степени их неравносильности, которая определяется как $1 - \mu(\tilde{A}_1, \tilde{A}_2)$.

Степень равносильности нечётких формул $\overline{A_1}(\overline{x}_1, \overline{x}_2, ..., \overline{x}_n)$ и $\overline{A_2}(\overline{x}_1, \overline{x}_2, ..., \overline{x}_n)$ обозначается $M(\overline{A_1}, \overline{A_2})$ и определяется выражением:

$$M(\overline{A}_1, \overline{A}_2) = \mathrm{I}\ \overline{A}_1(\overline{x}_1, \overline{x}_2, ..., \overline{x}_n) \leftrightarrow \overline{A}_2(\overline{x}_1, \overline{x}_2, ..., \overline{x}_n)$$

,где знаки эквивалентности «↔» и конъюнкции $\cap$ определяются на основе соответствующих числовых характеристик.

Рассмотрим понятия *нечётко истинных* и *нечётко ложных формул*. Пусть $\tilde{A}(\tilde{x}_1,\tilde{x}_2,...,\tilde{x}_n)$ - формула. Если при всех определённых значениях степеней истинности нечётких переменных $(\tilde{x}_1,\tilde{x}_2,...,\tilde{x}_n)$ значение степени истинности нечёткой логической формулы $\tilde{A}(\tilde{x}_1,\tilde{x}_2,...,\tilde{x}_n)$ больше или равно 0,5, то такую формулу можно назвать нечётко истинной и обозначать её $\tilde{И}$.

Если на этих же наборах степень истинности формулы $\tilde{A}(\tilde{x}_1,\tilde{x}_2,...,\tilde{x}_n)$ меньше или равна 0,5, то такую формулу можем считать нечётко ложной и обозначать $\tilde{Л}$.

6.3. Основные операции над нечёткими множествами и их свойства

Пусть $\overline{A}$ и $\overline{B}$ - нечёткие множества, причём:

$$\overline{A}=\{<M_A(x)\,|\,x>\} \qquad \overline{B}=\{<M_B(x)\,|\,x>\},x\in X.$$

Определение 6.7 *Пересечением множеств* $\overline{A}$ и $\overline{B}$ называется нечёткое множество, определяемое следующим образом:

$\overline{A}\wedge\overline{B}=\{<M_{A\cup B}(x)\,|\,x>\},x\in X$,где: $M_{A\wedge B}(x)=\min(M_A(x),M_b(x)).$

Определение 6.8 *Объединением нечётких множеств* $\overline{A}$ и $\overline{B}$ называется нечёткое множество, обозначаемое $\overline{A}\,V\,\overline{B}$ и определяемое как:

$\overline{A}V\overline{B}=\{<M_{AVB}(x)|x>\},x\in X$,где: $M_{AVB}(x)=\max(M_A(x),M_b(x)).$

Определение 6.9 *Дополнением множества* $\overline{A}$ называется и через $\neg\overline{A}$ обозначается нечёткое множество:

$\neg\overline{A}=\{<M_{\neg A}(x)\,|\,x>\},x\in X$, где: $M_{\neg A}(x)=1-M_A(x)$

Определение 6.10 *Разностью множеств* $\overline{A}$ и $\overline{B}$ называется множество

$\overline{A} \mid \overline{B} = \{< M_{A/B}(x), x > \mid x \in X\}$,где: $M_{A\setminus B}(x) = M_A(x) \wedge \neg M_B(x)$.

Определение 6.11 *Степень нечёткости* ρ($\overline{A}$) множества $\overline{A}$ в X есть отрицание степени равенства между множеством $\overline{A}$ и его носителем А, то есть: $\rho(\overline{A})=1\text{-}M(\overline{A},A)$.

Для произвольных нечётких множеств $\overline{A}, \overline{B}, \overline{C}$ *справедливы следующие свойства:*

$\neg(\neg\overline{A}) \approx \overline{A}$ (инволюция)

$\overline{A}\, V\, \overline{A} \approx \overline{A}$ (идемпотентность)

$\overline{A}\, \Lambda\, \overline{A} \approx \overline{A}$

$\overline{A}\, V\, \overline{B} \approx \overline{B}\, V\, \overline{A}$ (коммутативность)

$\overline{A}\, \Lambda\, \overline{B} \approx \overline{B}\, \Lambda\, \overline{A}$

$\overline{A}\, V(\overline{B}\, V\, \overline{C}) \approx (\overline{A}\, V\, \overline{B})\, V\, \overline{C} \approx \overline{A}\, V\, \overline{B}\, V\, \overline{C}$

(ассоциативность)

$\overline{A}\, \Lambda(\overline{B}\, \Lambda\, \overline{C}) \approx (\overline{A}\, \Lambda\, \overline{B})\, \Lambda\, \overline{C} \approx \overline{A}\, \Lambda\, \overline{B}\, \Lambda\, \overline{C}$

$\overline{A}\, V(\overline{B}\, \Lambda\, \overline{C}) \approx (\overline{A}\, \Lambda\, \overline{B})\, V(\overline{A}\, \Lambda\, \overline{C})$

$\neg(\overline{A}\, V\, \overline{B}) \approx \neg\overline{A}\, \Lambda \neg\overline{B}$

(закон де Моргана)

$\neg(\overline{A}\, \Lambda\, \overline{B}) \approx \neg\overline{A}\, V \neg\overline{B}$

$\overline{A}\, V \neg\overline{A} \approx \overline{B}\, V \neg\overline{B}$

$\overline{A}\, \Lambda \neg\overline{A} \approx \overline{B}\, \Lambda \neg\overline{B}$

$\overline{A}\, V \neg\overline{A}\, V\, \overline{B} \approx \overline{B}\, V \neg\overline{B}\, V\, \overline{A}$

Отметим особенность понятия прямого (декартового) произведения нечётких множеств.

Определение 6.12 *Прямым (декартовым) произведением нечётких множеств* $\tilde{A}$ и $\tilde{B}$ называется расплывчатое множество $\tilde{A} \times \tilde{B}$ в $X \times Y$, определяемое как:

$\tilde{A} \times \tilde{B} = \{< \mu_{A\times B}(x,y) \mid (x,y) >\}, (x,y) \in X \times Y$,

где $\mu_{A\times B}(x, y)=\mu_A(x) \wedge \mu_B(y)=\min(\mu_A(x), \mu_B(y))$.

Л И Т Е Р А Т У Р А

1. Андреева Е., Фалина И. Системы счисления и компьютерная арифметика. – М.: Лаборатория Базовых Знаний, 1999. – 256 с.
2. Ашинянц Р. А. Логические методы в искусственном интеллекте. – М.: МГАПИ, 2001. – 223 с.
3. Брой М. Информатика. Основополагающее введение: В 4-х ч. Ч. I. / Пер. с нем. – М.: Диалог-МИФИ, 1996. – 299 с.
4. Карпов Ю.Г. Теория автоматов – СПб.: Питер,2002.– 224с.,ил.
5. Курант Р., Роббинс Г. Что такое математика? – Ижевск, НИЦ “Регулярная и хаотическая динамика”, 2001. – 592 с.
6. Лыскова В. Ю., Ракитина Е. А. Логика в информатике. – М.: Лаборатория Базовых Знаний, 2001. – 160 с.
7. Новиков Ф. А. Дискретная математика для программистов. СПб.: Питер, 2001. – 304 с.
8. Судоплатов С. В., Овчинникова Е. В. Элементы дискретной математики: Учебник. – М.: ИНФРА-М, Новосибирск: Изд-во НГТУ, 2002. – 280 с.

Учебное издание

Ульянов Михаил Васильевич, Шептунов Максим Валерьевич

Математическая логика и теория алгоритмов
Часть I
Математическая логика
Учебное пособие

Подписано в печать 02.10.2002
Формат 60 х 80 1/16
Объем 3,0 п.л. Тираж 300 экз. Заказ № 128

Отпечатано в типографии Московской государственной академии
приборостроения и информатики
107846, Москва, ул. Стромынка, 20