1. Множества.

1.1 Множество

В основе рассматриваемых в математике логических теорий и моделей лежит теория множеств. Множество – простейшая математическая структура и конструкция, которая связывает несколько сущностей в целое. Связанные множеством сущности называют элементами этого множества. Если элемент связан множеством, то говорят, что элемент принадлежит этому множеству. Множество может быть задано явным перечислением всех его элементов либо некоторой формулой, которая описывает перечисляющую процедуру или разрешающую процедуру для этого множества, т.е. соответственно процедуру перечисления элементов этого множества или процедуру, которая для любой сущности может привести к ответу о том, является ли эта сущность элементом множества либо – нет. Например, следующее выражение читается так: множество таких и только таких элементов χ, для которых верно свойство, выраженное формулой $P(\chi)$.

$$\left(\left\{\chi \middle| P(\chi)\right\}\right)$$

Полагают, что для любого множества A верно, что любая сущность χ принадлежит этому множеству или не принадлежит. Это может быть записано следующим выражением.

$$\left((\chi \in A) \vee (\neg(\chi \in A))\right)$$

Допускается наличие кратных элементов во множестве, т.е. таких элементов, которые принадлежат множеству более, чем один раз; в этом случае множество называют мультимножеством. Также над множествами определены следующие отношения и операции.

1.1.1 Отношения над множествами.

Рассмотрим отношения подмножества и равенства множеств. Множество A, не имеющее кратных вхождений элементов, является подмножеством множества B тогда и только тогда, когда для любого элемента χ верно: что если этот элемент принадлежит множеству A, то он принадлежит множеству B.

$$\left((A \subseteq B) \sim \left(\forall \chi \left((\chi \in A) \to (\chi \in B)\right)\right)\right)$$

Любые два множества равны A и B тогда и только тогда, когда первое является подмножеством второго и второе является подмножеством первого.

$$\left((A = B) \sim \left((A \subseteq B) \wedge (B \subseteq A)\right)\right)$$

1.1.2 Операции над множествами.

Операции пересечения, объединения, разности и симметрической разности.

Пересечение двух множеств, не имеющих кратных вхождений элементов, равно множеству всех таких и только таких элементов, которые принадлежат первому множеству и второму.

$$\left((A \cap B) = \left\{\chi \middle| ((\chi \in A) \wedge (\chi \in B))\right\}\right)$$

Объединение двух множеств, не имеющих кратных вхождений элементов, равно множеству всех таких и только таких элементов, которые принадлежат первому множеству или второму.

$$\left((A \cup B) = \left\{\chi \middle| ((\chi \in A) \vee (\chi \in B))\right\}\right)$$

Разность двух множеств, не имеющих кратных вхождений элементов, равна множеству всех таких и только таких элементов, которые принадлежат первому множеству и не принадлежат второму.

$$\left((A/B) = \left\{\chi \middle| ((\chi \in A) \wedge (\neg(\chi \in B)))\right\}\right)$$

Симметрическая разность может быть выражена через понятие разности и объединения.

$$((A - B) = ((A/B) \cup (B/A)))$$

1.1.3 Мощность множества.

Множество может иметь конечное и бесконечное число элементов. В первом случае множество называется конечным, во втором случае – бесконечным. Если множество не имеет ни одного элемента, тогда такое множество называют пустым. Пустое множество будем обозначать следующим образом $\varnothing$.

Под мощностью конечного множества понимают число всех элементов этого множества.

Из множеств строятся другие абстракции, которые также используются в математической логике. Такими абстракциями являются вектора, отношения, функции, операции, математические модели, алгебры, формальные языки, формулы, формальные теории и пр.

1.2 Вектор

Понятие вектора и, в частности, пары является важным для многих математических теорий. Частным случаем вектора в теории множеств является пара, тройка, четвёрка, n-ка и т.д. Вектора состоят из компонентов. Вектора, компоненты которых являются символами, обычно называют строками. Разные авторы по-разному определяют или поясняют понятие вектора. Ниже это понятие поясним следующими примерами. Будем обозначать пару компонентов следующим образом $\langle \chi, \gamma \rangle$ и считать это обозначением множества, которое равно множеству вида $\{\{\{\varnothing,\{\chi\}\}\},\{\{\varnothing,\{\chi\}\},\{\gamma\}\}\}$. Отсюда следует, что верно равенство:

$$\left(\langle \chi, \gamma \rangle = \{\{\{\varnothing,\{\chi\}\}\},\{\{\varnothing,\{\chi\}\},\{\gamma\}\}\}\right)$$

Вектор из одного элемента $\langle \chi \rangle$ может быть представлена так:

$$\{\{\varnothing,\{\chi\}\}\}$$

Вектор из трёх $\langle \chi,\gamma,\lambda \rangle$ – так:

$$\left\{\left\{\left\{\left\{\varnothing,\{\chi\}\right\}\right\}\right\},\left\{\left\{\left\{\varnothing,\{\chi\}\right\},\{\gamma\}\right\}\right\},\left\{\left\{\varnothing,\left\{\left\{\varnothing,\{\chi\}\right\},\{\gamma\}\right\}\right\},\{\lambda\}\right\}\right\}$$

Другие вектора и строки – аналогично.

1.2.1 Отношения над векторами.

Как и для множеств для векторов определено отношение равенства. Другие отношение между векторами будут рассмотрены в следующих разделах.

1.2.2 Операции над векторами.

Над векторами определена операция конкатенации (соединения строк).

$$\left(\left(\langle \chi \rangle \oplus \langle \gamma,\lambda \rangle\right) = \langle \chi,\gamma,\lambda \rangle\right)$$

1.2.3 Размерность вектора.

Исходя из представления векторов, под размерностью вектора будем понимать мощность вектора.

1.3 Декартово произведение множеств. Соответствие. Функция

Важность понятия вектора подтверждается например тем, что декартово произведение n множеств определяется как множество всевозможных различных векторов размерности n, компонентами которых являются элементы этих множеств, а под *соответствием* – понимают подмножество декартова произведения пары множеств. Первое множество в этой паре называют областью определения соответствия, а второе – областью значений соответствия. Декартово произведение двух и трёх множеств будем обозначать соответственно $(A \times B)$ и $(A \times B \times C)$. Декартово произведение n-равных множеств A будем обозначать (A^n). В теоретико-множественном смысле, используя понятие пары, понимают и функцию, как однозначное соответствие, т.е. такое соответствие, в котором не найдётся двух пар таких, что первые компоненты этих пар различны, а вторые компоненты – совпадают. Множество полностью определённых функций, область которых есть множество A, а область значений – множество B будем обозначать B^A.

1.4 Отношение

Под n-арным отношением понимают подмножество декартова произведения n множеств. Среди отношений выделяют бинарные отношения, которые могут быть классифицированы в соответствии с наличием у этих отношений тех или иных свойств. Выделяют следующие свойства бинарных отношений: рефлексивность, арефлексивность, симметричность, антисимметричность, асимметричность, полноту, транзитивность и другие. Если читатель не знаком с определениями этих свойств, то соответствующие формулировки он сможет найти ниже, в соответствующих разделах.

1.5 Формальные языки

1.5.1 Формальный язык.

Известно, что языки можно классифицировать на два класса: естественные языки и формальные языки. Первые отличаются от вторых в основном тем, что правила построения текстов этих языков явно или жёстко не фиксируются для пользователей языка, язык развивается стихийно. Формальные же языки явно и строго описываются правилами, едиными для всех пользователей языка. Языки состоят из текстов, которые являются дискретными информационными конструкциями, т.е. состоят из конструкций, каждая из которых содержит конечное число элементов – знаков, которые используются для обозначения того, что описывает эта информационная конструкция.

Под формальным языком будем понимать множество текстов этого языка. Формальные языки могут быть линейные и графовые. Линейные языки состоят из строк, т.е. линейный язык – это множество строк. Строка является частным случаем множества и в данном случае рассматривается как вектор символов.

1.5.2 Отношения над строками.

Одни строки могут являться подстроками других. Учитывая рассмотренные примеры, также справедливо: если для любого элемента одной строки объединение всех пересечений одноэлементного множества этого элемента с элементом другой строки не равно пустому множеству, то первая строка является подстрокой второй строки. Если строка имеет размерность один и её пересечение с элементом второй строки не равно пустому множеству, то компонент первой строки является последним компонентом второй строки. Если один компонент является последним компонентом одной строки, и ещё один компонент является последним компонентом другой строки, размерность которой на единицу меньше, чем размерность первой, то строка из второго и первого компонентов, в которой первый компонент является последним, является подстрокой первой строки. Понятие подстроки является транзитивным отношением. Например, строка $\langle \chi, \gamma \rangle$ является подстрокой строки $\langle \chi, \gamma, \lambda \rangle$.

1.5.3 Метаописание языков.

Чтобы задать язык необходимо задать алфавит языка, синтаксические правила и семантические правила. Алфавит языка – это множество классов символов, из которых можно составлять тексты языка с помощью синтаксических правил. Семантические правила описывают механизм установки *соответствия* между каждым знаком в тексте языка и смыслом, значением этого знака. Формальные языки могут рассматриваться и при отсутствии семантических правил. Для описания языков используются метаязыки. Метаязыки, например, используются при описании синтаксических правил формального языка. Множество синтаксических правил является грамматикой языка. Примером такого метаязыка является язык Бэкуса-Наура. Далее, для описания синтаксических правил будем использовать следующий метаязык. Алфавит этого языка содержит классы так называемых метасимволов, соответствующие символам следующего множества.

$\{\langle,\rangle,[,],[\{,\}],::=,|\}$

Кроме этих символов в алфавите языка используются любые другие символы. Тексты этого языка строятся из строк. Любая строка символов, которая не содержит метасимволов, рассматривается как лексическая константа (константная лексема). Значением константной лексемы является она сама. Если строка заключена в угловые скобки, то она рассматривается как лексическая переменная (переменная лексема). Значением лексической переменной может являться строка другого языка. Можно выражать значение одной лексической переменной через другие переменные и лексические константы, используя специальный символ удвоенного двоеточия с равенством, который должен быть расположен непосредственно за лексической переменной. Если лексема заключена в квадратные скобки, то это значит, что значение этой лексемы можно не использовать. Если лексема заключена в пару квадратно-фигурных скобок, то это значит, что значение лексемы можно использовать ноль раз и больше. Если две лексемы разделены вертикальной чертой, то это значит, что выбирается значение только одной из них.

Опишем на этом языке его собственный синтаксис.

$\langle assigning \rangle ::= ::=$

$\langle left\ angle\ bracket \rangle ::= \text{б}$

$\langle right\ angle\ bracket \rangle ::= \text{с}$

$\langle left\ square\ bracket \rangle ::= [$

$\langle right\ square\ bracket \rangle ::=]$

$\langle left\ squared\ \ figure\ bracket \rangle ::= [\{$

$\langle right\ squared\ \ figure\ bracket \rangle ::= \}]$

$\langle splitter \rangle ::= |$

$\langle \textit{text} \rangle ::= \langle \textit{definition} \rangle \left[\{ \langle \textit{definition} \rangle \} \right]$

$\langle \textit{definition} \rangle ::= \langle \textit{lexical variable} \rangle \langle \textit{assigning} \rangle \langle \textit{lexical expression} \rangle$

$\langle \textit{lexical expression} \rangle ::= \langle \textit{lexical list} \rangle \mid \langle \textit{lexical simple expression} \rangle$

$\langle \textit{lexical simple expression} \rangle ::= \langle \textit{lexical unary expression} \rangle \mid \langle \textit{lexical splitted expression} \rangle$

$\langle \textit{lexical list} \rangle ::= \langle \textit{lexical unary expression} \rangle \left[\{ \langle \textit{lexical unary expression} \rangle \} \right]$

$\langle \textit{lexical unary expression} \rangle ::= \langle \textit{lexical atom} \rangle \mid \langle \textit{lexical bracket expression} \rangle$

$\langle \textit{lexical bracket expression} \rangle ::= \langle \textit{square bracket expression} \rangle \mid \langle \textit{figure bracket expression} \rangle$

$\langle \textit{square bracket expression} \rangle ::= \langle \textit{left square bracket} \rangle \langle \textit{lexical expression} \rangle \langle \textit{right square bracket} \rangle$

$\langle \textit{figure bracket expression} \rangle ::= \langle \textit{left squared figure bracket} \rangle \langle \textit{lexical expression} \rangle \langle \textit{right squared figure bracket} \rangle$

$\langle \textit{lexical splitted expression} \rangle ::= \langle \textit{lexical simple expression} \rangle \langle \textit{splitter} \rangle \langle \textit{lexical simple expression} \rangle$

$\langle \textit{lexical atom} \rangle ::= \langle \textit{lexical constant} \rangle \mid \langle \textit{lexical variable} \rangle$

$\langle \textit{lexical variable} \rangle ::= \langle \textit{left angle bracket} \rangle \langle \textit{lexical constant} \rangle \langle \textit{right angle bracket} \rangle$

$\langle \textit{lexical constant} \rangle$

Этот же самый язык Lmeta, тексты которого являются значениями лексической переменной $\langle \textit{text} \rangle$, при условии, что в качестве значений E и P берутся множества минимальные из возможных, а переменная $\langle \textit{lexical constant} \rangle$ принимает любые значения только из множества C, может быть описан на теоретико-множественном языке.

$$\mathrm{Lmeta} = \left\{ (\delta \oplus \tau) \,\middle|\, \left(\exists S \left(\left(\langle \delta, \tau \rangle \in \left((D \cap S) \times S \right) \right) \wedge \left(S \subset \mathrm{Lmeta} \right) \right) \right) \right\}$$

$$D = \left\{ \left((\delta \oplus \lambda) \oplus \tau \right) \,\middle|\, \left(\langle \delta, \lambda, \tau \rangle \in \left(V \times \{::=\} \times E \right) \right) \right\}$$

$$E = L \cup P$$

$$P = U \cup T$$

$$L = \left\{ (\delta \oplus \tau) \,\middle|\, \left(\exists S \left(\left(\langle \delta, \tau \rangle \in \left((U \cap S) \times S \right) \right) \wedge \left(S \subset L \right) \right) \right) \right\}$$

$$U = M \cup R$$

$$R = Q \cup F$$

$$Q = \left\{ \left((\delta \oplus \lambda) \oplus \tau \right) \,\middle|\, \left(\langle \tau, \lambda, \delta \rangle \in \left(\{\ ]\} \times E \times \{[\ \} \right) \right) \right\}$$

$$F = \left\{ \left((\delta \oplus \lambda) \oplus \tau \right) \,\middle|\, \left(\langle \tau, \lambda, \delta \rangle \in \left(\{\}]\} \times E \times \{[\{\} \right) \right) \right\}$$

$$T = \left\{ \left((\delta \oplus \lambda) \oplus \tau \right) \,\middle|\, \left(\langle \delta, \lambda, \tau \rangle \in \left(P \times \{|\} \times P \right) \right) \right\}$$

$$M = C \cup V$$

$$V = \left\{ \left((\delta \oplus \lambda) \oplus \tau \right) \,\middle|\, \left(\langle \tau, \lambda, \delta \rangle \in \left(\{\text{c}\} \times E \times \{\text{б}\} \right) \right) \right\}$$

2. Логика высказываний.

2.1 Высказывание

Математическая логика работает с логическими утверждениями, которые представляются в виде логических формул. В простейшем случае логическая формула может задавать некоторое высказывание (суждение). Высказывание – это некоторая сущность, являющаяся мысленной абстракцией, относительно которой можно сказать: истинна она или ложна. С формальной точки зрения высказывание – это функция, области значений которой принадлежат два понятия: истина и ложь. Для выражения этих понятий используются следующие символы, которые называются логическими константами.

ложь	истина
Л	*И*
0	1
F	T
false	*true*
⊥	•

Будем использовать последний вариант обозначения.

Текст, который обозначает некоторую функцию, будем называть формулой. Следовательно, высказывания представляются с помощью формул некоторого языка. Из простых формул с помощью логических связок составляются более сложные. В следующей таблице приведены основные логические связки и варианты их обозначений.

эквиваленция	отрицание	дизъюнкция	конъюнкция	импликация
equ	*not*	*or*	*and*	*imply*
↔	!	‖	&&	⊃
⇔		|	&	⇒
~	¬	∨	∧	→

С высказываниями часто можно встретиться при использовании естественных языков. Примерами высказывательных предложений естественного языка являются следующие фразы.

«За мгновение до своей смерти он был еще жив.»

«Если верно, что если идет дождь, то дорога мокрая, то справедливо также и следующее утверждение: если дорога не мокрая, то дождь не идёт.»

«Земля вертится.»

Чтобы убедиться в правильности первого предложения, достаточно понимать смысл слов: это предложение (в значительной степени) высказывает *истину языка.* Чтобы принять второе утверждение, достаточно понимать смысл некоторых слов (если ... то, нет), а также знать, что куски фразы «(не) идет дождь» и «дорога (не) мокрая» выражают *высказывания*, которые могут быть истинными или ложными. Высказывание, выраженное вторым предложением, останется истинным, если заменить эти два куска другими. В этом случае высказываемые истины языка называются *логическими истинами.* Напротив, третье предложение не высказывает истину языка, так как оно выражает некоторый факт (в данном случае — из физики и астрономии). Таким образом, это предложение высказывает *фактическую истину.* Элементарными высказываниями естественно считать такие, которые отражают единичный факт.

Итак, в логике высказываний используются формулы, а также такие понятия, как истина и ложь. Одной из основных задач математической логики и, в частности логики высказываний, является задача выяснения того, в каких тех или иных случаях истинна или ложна та или иная формула.

2.1.1 Язык логики высказываний

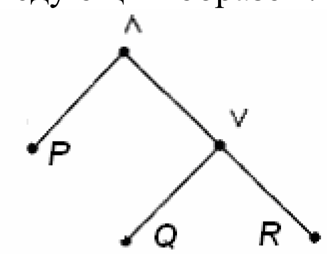
Язык логики высказываний является формальным языком. Этот язык будем далее обозначать языком $\mathbf{L}$. Зададим этот язык с помощью введённого ранее метаязыка. Вначале введём лексемы для символов, принадлежащих различным классам алфавита языка $\mathbf{L}$, т.е. опишем его алфавит.

$$\langle constant \rangle ::= \perp | \bullet$$
$$\langle non-zero\ digit \rangle ::= 1|2|3|4|5|6|7|8|9$$
$$\langle digit \rangle ::= 0 | \langle non-zero\ digit \rangle$$
$$\langle natural \rangle ::= \Big[\{\langle digit \rangle\}\Big]\langle non-zero\ digit \rangle$$
$$\langle symbol \rangle ::= A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z$$
$$\langle implication \rangle ::= \rightarrow$$
$$\langle conjunction \rangle ::= \wedge$$
$$\langle disjunction \rangle ::= \vee$$
$$\langle negation \rangle ::= \neg$$
$$\langle equivalence \rangle ::= \sim$$
$$\langle unary\ connective \rangle ::= \langle negation \rangle$$
$$\langle binary\ connective \rangle ::= \langle implication \rangle | \langle conjunction \rangle | \langle disjunction \rangle | \langle negation \rangle | \langle equivalence \rangle$$
$$\langle left\ bracket \rangle ::= ($$
$$\langle right\ bracket \rangle ::=)$$

Следующая грамматика описывает синтаксис языка $\mathbf{L}$.

$$\langle atom \rangle ::= \langle symbol \rangle \Big[\{\langle natural \rangle\}\Big]$$
$$\langle formula \rangle ::= \langle constant \rangle | \langle atom \rangle | \langle unary\ complex\ formula \rangle | \langle binary\ complex\ formula \rangle$$
$$\langle unary\ complex\ formula \rangle ::= \langle left\ bracket \rangle \langle unary\ connective \rangle \langle formula \rangle \langle right\ bracket \rangle$$
$$\langle binary\ complex\ formula \rangle ::= \langle left\ bracket \rangle \langle formula \rangle \langle binary\ connective \rangle \langle formula \rangle \langle right\ bracket \rangle$$

Подстрока другой строки называется подформулой тогда и только тогда, когда обе строки являются формулами. Отношение подформулы задаёт на множестве формул языка $\mathbf{L}$ древовидную структуру (или ориентированный граф без циклов). Например, для формулы $(P \wedge (Q \vee R))$ и её подформул эта структура будет выглядеть следующим образом.



Эта формула имеет три атомарные подформулы – P, Q и R – и две сложные подформулы: $(P \wedge (Q \vee R))$ и $(Q \vee R)$. Атомарная формула – это формула, которая не содержит логических связок и не является логической константной. Сложная формула – содержит логические связки.

2.2 Алгебра высказываний

2.2.1 Истинностные функции, операции и переменные.

Рассмотрим двуэлементное множество $(\mathrm{B} = \{\perp, \bullet\})$.

Будем называть n-арной операцией на множестве A функцию φ вида $\left(\varphi \in A^{\left(A^n\right)}\right)$.

Истинностной n-местной функцией (функцией n аргументов) будем называть любую функцию являющуюся подмножеством n-арной операции φ вида $\left(\varphi \in \mathrm{B}^{\left(\mathrm{B}^n\right)}\right)$.

Алгебра, образованная k-элементным множеством вместе со всеми операциями на нем, называется алгеброй k-значной логики, а n-арные операции на k-элементном множестве называются k-значными логическими функциями n аргументов.

Таким образом, истинностные функции являются логическими функциями двузначной логики.

Аргумент i для истинностной функции φ называется несущественным (или фиктивным), если $\left(\varphi\left(\langle \chi 1,...,\chi i-1,\perp,\chi i+1,...,\chi n\rangle\right) = \varphi\left(\langle \chi 1,...,\chi i-1,\bullet\ ,\chi i+1,...,\chi n\rangle\right)\right)$ при любых значениях остальных аргументов, т. е. если изменение значения χi для любого набора значений $\langle \chi 1,...,\chi i-1,\chi i+1,...,\chi n\rangle$ не меняет значения функции φ.

Две функции φ и ψ называют равнозначными тогда и только тогда, когда существуют натуральные i и k, что для любых наборов верно $\left(\varphi\left(\langle \chi 1,...,\chi i-1,\chi i,...,\chi i+k,\chi i+k+1...,\chi n\rangle\right) = \psi\left(\langle \chi 1,...,\chi i-1,\chi i+k,...,\chi n\rangle\right)\right)$.

Для любой конечной совокупности функций всегда можно перейти к совокупности соответственно равнозначных функций зависящих и определённых для одного и того же множества аргументов (являющегося объединением множеств аргументов всех взятых функций), что часто бывает удобно.

Примеры логических функций. Истинностных функций одного аргумента — четыре.

$$(\varphi 0(\chi) = \perp)$$
$$(\varphi 1(\chi) = \chi)$$
$$(\varphi 2(\chi) = (\neg\chi))$$
$$(\varphi 3(\chi) = \bullet\ )$$

Функции $\varphi 0$ и $\varphi 3$ равнозначны константам (нульместным функциям) $\perp$ и $\bullet$ соответственно; их значения не зависят от значения аргумента, и, следовательно, аргумент χ для них несущественен. Функция $\varphi 1$ «повторяет» χ. Такие функции соответствуют переменным (логическим переменным). Функция $\varphi 2$ называется функцией *отрицания* χ (или «функцией НЕ») и обозначается: $(\neg\chi)$. Ее значение «противоположно» значению χ.

Функция $\psi 1$ называется ***конъюнктивной*** $\chi 1$ и $\chi 2$ и семантически обозначается: $(\chi 1 \wedge \chi 2)$. Она равна $\bullet$, только если $\chi 1$ и $\chi 2$ равны $\bullet$, поэтому ее называют часто «функцией И». Еще одно ее название — «логическое умножение», поскольку ее таблица взаимнооднозначно соответствует таблице обычного умножения для чисел 0 и 1.
Функция $\psi 7$ называется ***дизъюнктивной*** $\chi 1$ и $\chi 2$; её семантическое обозначение: $(\chi 1 \vee \chi 2)$. Она равна $\bullet$, если $\chi 1$ или $\chi 2$ равен $\bullet$ (хотя бы один из двух). Поэтому ее называют часто «функцией ИЛИ».

$\chi 1$	$\chi 2$	$\psi 0$	$\psi 1$	$\psi 2$	$\psi 3$	$\psi 4$	$\psi 5$	$\psi 6$	$\psi 7$	$\psi 8$	$\psi 9$	$\psi 10$	$\psi 11$	$\psi 12$	$\psi 13$	$\psi 14$	$\psi 15$
$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\bullet$	$\bullet$	$\bullet$	$\bullet$	$\bullet$	$\bullet$	$\bullet$	$\bullet$
$\perp$	$\bullet$	$\perp$	$\perp$	$\perp$	$\perp$	$\bullet$	$\bullet$	$\bullet$	$\bullet$	$\perp$	$\perp$	$\perp$	$\perp$	$\bullet$	$\bullet$	$\bullet$	$\bullet$
$\bullet$	$\perp$	$\perp$	$\perp$	$\bullet$	$\bullet$	$\perp$	$\perp$	$\bullet$	$\bullet$	$\perp$	$\perp$	$\bullet$	$\bullet$	$\perp$	$\perp$	$\bullet$	$\bullet$
$\bullet$	$\bullet$	$\perp$	$\bullet$	$\perp$	$\bullet$	$\perp$	$\bullet$	$\perp$	$\bullet$	$\perp$	$\bullet$	$\perp$	$\bullet$	$\perp$	$\bullet$	$\perp$	$\bullet$

Функция $\psi 6$ — это ***сложение по модулю 2*** («функция исключающего ИЛИ»). Она равна $\bullet$, когда значения ее аргументов различны, и равна $\perp$, когда они равны. Поэтому функцию $\psi 6$ иногда называют *неравнозначностью.*
Функция $\psi 9$ называется ***эквивалентностью****,* или *равнозначностью.* Ее семантическое обозначение с помощью связки эквиваленции: $(\chi 1 \sim \chi 2)$. Она равна $\bullet$, когда значения её аргументов равны, и равна $\perp$, когда они различны.
Еще три функции имеют свои названия:

- $\psi 13$ — ***импликативная***; обозначение: $(\chi 1 \to \chi 2)$, читается «если $\chi 1$, то $\chi 2$»;
- $\psi 8$ — функция ***стрелки Пирса***;
- $\psi 14$ — функция ***штриха Шеффера***.

Остальные функции специальных названий не имеют и, как будет показано позднее, легко выражаются через перечисленные ранее функции.
В функциях $\psi 3$ и $\psi 12$ аргумент $\chi 2$ фиктивен; из таблицы видно, что $(\psi 3(\langle \chi 1, \chi 2 \rangle) = \chi 1)$, $(\psi 12(\langle \chi 1, \chi 2 \rangle) = \chi 1)$. В функциях $\psi 5$ и $\psi 10$ фиктивен аргумент $\chi 1$: $(\psi 5(\langle \chi 1, \chi 2 \rangle) = \chi 2)$, $(\psi 10(\langle \chi 1, \chi 2 \rangle) = \chi 2)$.
Таким образом, из 16 функций двух аргументов шесть функций имеют фиктивные аргументы. С ростом n (числа аргументов) доля функций, имеющих фиктивные аргументы, убывает и стремится к нулю.

2.2.2 Высказывательностные функции, суперпозиции и интерпретации

Рассмотрим множество всевозможных истинностных функций $\mathsf{B\,fnc}$. Тогда высказывательностной функцией n аргументов будет любая функция φ тогда и только тогда, когда $\left((\varphi \subseteq \psi) \wedge \left(\psi \in \mathrm{B\ fnc}^{\left(\mathrm{B\ fnc}^{n}\right)}\right)\right)$. Любое значение высказывательностной функции ψ для набора аргументов $\langle \varphi 1, \ldots, \varphi m \rangle$ есть суперпозиция функций $\varphi 1, \ldots, \varphi m, \varphi m+1$, где истинностная функция $\varphi m+1$ может быть получена применением функции ψ, только к истинностным функциям, которые равнозначны функциям, обозначенными атомарными формулами. Если ψ есть суперпозиция функций $\varphi 0, \varphi 1, \ldots, \varphi m$ и $\varphi 0$ есть суперпозиция функций $\varphi m+1, \varphi m+2, \ldots, \varphi n$, то ψ является суперпозицией функций $\varphi 1, \ldots, \varphi n$. Если ψ есть суперпозиция функций $\varphi 0, \varphi 1, \ldots, \varphi m$ и $\varphi m+1$ равнозначна функции $\varphi 0$, то ψ является суперпозицией функций $\varphi 1, \ldots, \varphi m, \varphi m+1$.

Каждая формула языка L обозначает истинностную функцию, а подформулы в комплексе со связывающими их логическими связками в формулах соответствуют высказывательностным функциям, позволяя из формул, обозначающих одни истинностные функции, конструировать другие формулы, обозначающие и выражающие ту или иную суперпозицию соответствующих истинностных функций.

Всякая формула, выражающая функцию φ как суперпозицию исходных функций, задает способ ее вычисления (при условии, что известно, как вычислить исходные функции). Этот способ определяется следующим правилом: **значение обозначаемой формулой функции φ можно вычислить, если уже вычислены значения всех функций, обозначаемых подформулами этой формулы**. Фактически это правило может являетья основой для реализации соответствующей высказывательностной функции. Для того, чтобы применить это правило рассмотрим понятие интерпретации.

Интерпретацией формулы языка L называют функцию, отображающую множество всех атомарных подформул этой формулы, на множество B, таким образом, что для любой пары равных атомарных подформул этой формулы значения каждой подформулы из этой пары совпадают.

Например, для формулы $(P \wedge Q)$ можно рассмотреть четыре следующие интерпретации.

$$\{\langle P, \perp \rangle, \langle Q, \perp \rangle\}$$
$$\{\langle P, \perp \rangle, \langle Q, \bullet \rangle\}$$
$$\{\langle P, \bullet \rangle, \langle Q, \perp \rangle\}$$
$$\{\langle P, \bullet \rangle, \langle Q, \bullet \rangle\}$$

Эти интерпретации могут быть заданы таблично.

P	Q
$\perp$	$\perp$
$\perp$	$\bullet$
$\bullet$	$\perp$
$\bullet$	$\bullet$

Функция ψ, которую обозначает эта формула, является конъюнктивной функцией и имеет следующий вид.

$$\{\langle\langle\bot,\bot\rangle,\bot\rangle,\langle\langle\bot,\bullet\rangle,\bot\rangle,\langle\langle\bullet,\bot\rangle,\bot\rangle,\langle\langle\bullet,\bullet\rangle,\bullet\rangle\}$$

При помощи интерпретаций эта же функция может быть задана таблично, в виде таблицы истинности.

P	Q	$(P \wedge Q)$
$\bot$	$\bot$	$\bot$
$\bot$	$\bullet$	$\bot$
$\bullet$	$\bot$	$\bot$
$\bullet$	$\bullet$	$\bullet$

В данном случае как раз и применено описанное ранее правило. Действительно, формула $(P \wedge Q)$, обозначает конъюнктивную функцию, которая является суперпозицией пары функций, обозначенных подформулами P и Q и соответствующих переменным. Каждая интерпретация позволяет вычислить значения функций, обозначаемых формулами P и Q, и использовать эти значения при вычислении значения конъюнктивной функции. Для рассмотренной формулы элементом высказывательностной функции, соответствующим конъюнктивной связке $\wedge$ и подформулам P и Q, является следующая пара.

$$\langle\langle\{\langle\langle\bot,\bot\rangle,\bot\rangle,\langle\langle\bot,\bullet\rangle,\bot\rangle,\langle\langle\bullet,\bot\rangle,\bullet\rangle,\langle\langle\bullet,\bullet\rangle,\bullet\rangle\},\{\langle\langle\bot,\bot\rangle,\bot\rangle,\langle\langle\bot,\bullet\rangle,\bullet\rangle,\langle\langle\bullet,\bot\rangle,\bot\rangle,\langle\langle\bullet,\bullet\rangle,\bullet\rangle\}\rangle,\psi\rangle$$

Другим элементом этой высказывательностной функции может являться, например – для формулы $(P \wedge P)$, следующая пара.

$$\langle\langle\{\langle\bot,\bot\rangle,\langle\bullet,\bullet\rangle\},\{\langle\bot,\bot\rangle,\langle\bullet,\bullet\rangle\}\rangle,\{\langle\bot,\bot\rangle,\langle\bullet,\bullet\rangle\}\rangle$$

Рассмотрим ещё один пример применения этого правила для более сложной формулы $(P \wedge (Q \vee R))$.

P	Q	R	$(Q \vee R)$	$(P \wedge (Q \vee R))$
$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
$\bot$	$\bot$	$\bullet$	$\bullet$	$\bot$
$\bot$	$\bullet$	$\bot$	$\bullet$	$\bot$
$\bot$	$\bullet$	$\bullet$	$\bullet$	$\bot$
$\bullet$	$\bot$	$\bot$	$\bot$	$\bot$
$\bullet$	$\bot$	$\bullet$	$\bullet$	$\bullet$
$\bullet$	$\bullet$	$\bot$	$\bullet$	$\bullet$
$\bullet$	$\bullet$	$\bullet$	$\bullet$	$\bullet$

Полученную истинностную функцию φ

$$\left(\varphi = \{\langle\langle\bot,\bot,\bot\rangle,\bot\rangle,\langle\langle\bot,\bot,\bullet\rangle,\bot\rangle,\langle\langle\bot,\bullet,\bot\rangle,\bot\rangle,\langle\langle\bot,\bullet,\bullet\rangle,\bullet\rangle,\langle\langle\bullet,\bot,\bot\rangle,\bullet\rangle,\langle\langle\bullet,\bot,\bullet\rangle,\bot\rangle,\langle\langle\bullet,\bullet,\bot\rangle,\bot\rangle,\langle\langle\bullet,\bullet,\bullet\rangle,\bullet\rangle\}\right)$$

можно считать суперпозицией следующих функций.

$$\{\langle\bot,\bot\rangle,\langle\bullet,\bullet\rangle\}$$

$$\{\langle\langle\bot,\bot\rangle,\bot\rangle,\langle\langle\bot,\bullet\rangle,\bot\rangle,\langle\langle\bullet,\bot\rangle,\bot\rangle,\langle\langle\bullet,\bullet\rangle,\bullet\rangle\}$$

$$\{\langle\langle\bot,\bot\rangle,\bot\rangle,\langle\langle\bot,\bullet\rangle,\bullet\rangle,\langle\langle\bullet,\bot\rangle,\bullet\rangle,\langle\langle\bullet,\bullet\rangle,\bullet\rangle\}$$

Соответствующими элементами высказывательностных функций являются следующие две пары.

$$\left\langle\left\langle\left\{\left\langle\langle\perp,\perp\rangle,\perp\right\rangle,\left\langle\langle\perp,\bullet\rangle,\perp\right\rangle,\left\langle\langle\bullet,\perp\rangle,\bullet\right\rangle,\left\langle\langle\bullet,\bullet\rangle,\bullet\right\rangle\right\},\left\{\left\langle\langle\perp,\perp\rangle,\perp\right\rangle,\left\langle\langle\perp,\bullet\rangle,\bullet\right\rangle,\left\langle\langle\bullet,\perp\rangle,\perp\right\rangle,\left\langle\langle\bullet,\bullet\rangle,\bullet\right\rangle\right\}\right\rangle,\delta\right\rangle$$

$$\left(\delta = \left\{\left\langle\langle\perp,\perp\rangle,\perp\right\rangle,\left\langle\langle\perp,\bullet\rangle,\bullet\right\rangle,\left\langle\langle\bullet,\perp\rangle,\bullet\right\rangle,\left\langle\langle\bullet,\bullet\rangle,\bullet\right\rangle\right\}\right)$$

$$\left\langle\left\langle\left\{\left\langle\langle\perp,\perp,\perp\rangle,\perp\right\rangle,\left\langle\langle\perp,\bullet,\perp\rangle,\perp\right\rangle,\left\langle\langle\perp,\perp,\bullet\rangle,\perp\right\rangle,\left\langle\langle\perp,\bullet,\bullet\rangle,\perp\right\rangle,\left\langle\langle\bullet,\perp,\perp\rangle,\bullet\right\rangle,\left\langle\langle\bullet,\perp,\bullet\rangle,\bullet\right\rangle,\left\langle\langle\bullet,\bullet,\perp\rangle,\bullet\right\rangle,\left\langle\langle\bullet,\bullet,\bullet\rangle,\bullet\right\rangle\right\},\delta\right\rangle,\varphi\right\rangle$$

Таким образом, **формула каждой интерпретации ставит в соответствие значение функции и, следовательно, может служить наряду с таблицей способом задания и вычисления этой функции**. В частности, по формуле, вычисляя значения функции n аргументов для всех 2^n интерпретаций, можно восстановить таблицу этой функции.

2.2.3 Общезначимость

Формула называется общезначимой логической формулой тогда и только тогда, когда она обозначает функцию равнозначную константе $\bullet$.
Примером общезначимой формулы является формула $(P \to P)$.

P	$(P \to P)$	$\bullet$
$\perp$	$\bullet$	$\bullet$
$\bullet$	$\bullet$	$\bullet$

Ещё один пример общезначимой формулы $(A \to (B \to A))$.

A	B	$(B \to A)$	$(A \to (B \to A))$
$\perp$	$\perp$	$\bullet$	$\bullet$
$\perp$	$\bullet$	$\perp$	$\bullet$
$\bullet$	$\perp$	$\bullet$	$\bullet$
$\bullet$	$\bullet$	$\bullet$	$\bullet$

Формула называется необщезначимой тогда и только тогда, когда она не является общезначимой.

2.2.4 Противоречивость

Формула называется противоречивой логической формулой тогда и только тогда, когда она обозначает функцию равнозначную константе $\perp$.
Примером противоречивой формулы является формула $(P \sim \neg P)$.

P	$(P \sim \neg P)$	$\perp$
$\perp$	$\perp$	$\perp$
$\bullet$	$\perp$	$\perp$

2.2.5 Нейтральность

Формула, не являющаяся ни общезначимой, ни противоречивой, называется нейтральной.
Примером нейтральной формулы является формула $(\neg P)$.

P	$(\neg P)$
$\perp$	$\bullet$
$\bullet$	$\perp$

2.2.6 Выполнимость

Формула называется выполнимой (непротиворечивой) тогда и только тогда, когда она является нейтральной или общезначимой.
Примером выполнимой формулы является формула $(P \vee Q)$.

P	Q	$(P \vee Q)$
$\perp$	$\perp$	$\perp$
$\perp$	•	•
•	$\perp$	•
•	•	•

Формула называется невыполнимой тогда и только тогда, когда она не является выполнимой. Множество формул невыполнимо тогда и только тогда, когда конъюнкция всех этих и только этих формул невыполнима.

2.2.7 Следование и равносильность формул

Одна формула следует из другой тогда и только тогда, когда для любой интерпретации, для которой значение обозначаемой второй формулой функции равнозначно • , значение обозначаемой первой формулой функции также равнозначно • .
Например, формула $(P \to Q)$ следует из формулы $(P \wedge Q)$. Это записывается так $((P \wedge Q)\text{‘}\ (P \to Q))$. Если формула следует из любой формулы, т.е. является общезначимой, то это записывается так: $(\text{‘}\ (P \to P))$.

Формула следует из формулы тогда и только тогда когда множество невыполнимо.
Две формулы равносильны тогда и только тогда, когда следуют друг из друга. Например, формулы $(\neg P \vee Q)$ и $(P \to Q)$ равносильны, так как: $((\neg P \vee Q)\text{‘}\ (P \to Q))$ и $((P \to Q)\text{‘}\ (\neg P \vee Q))$. Равносильность формул записывается так: $((P \to Q) \Leftrightarrow (\neg P \vee Q))$.

Следовательно, две формулы равносильны тогда и только тогда, когда обозначают функции, для которых существует каждой из них равнозначная или равная функция.
Чтобы проверить являются ли две формул равносильными можно вычислить значения, обозначаемых ими функций для каждой из интерпретаций, что требует $2 * 2^n$ вычислений (если считать, что обе функции имеют n аргументов) и на практике оказывается слишком громоздким. Существуют и другие методы установления равносильности формул и получения новых формул, равносильных исходной.

2.2.8 Логический базис и алгебра высказываний

Логическим базисом называется множество истинностных функций тогда и только тогда, когда любая истинностная функция является суперпозицией всех или некоторых истинностных функций из этого множества.
Примерами логических базисов являются: множество, элементами которого являются импликативная функция и истинностная функция, равнозначная

логической константе $\perp$, множество, элементы которого – функция отрицания и конъюнктивная функция, одноэлементное множество, с элементом – функция стрелки Пирса, и другие.
Минимальным логическим базисом является множество функций, являющееся логическим базисом, ни одно строгое подмножество которого не является логическим базисом.
Алгебру высказываний задают парой, первым компонентом которой является множество всевозможных истинностных функций, являющееся носителем этой алгебры, а втором компонентом – некоторое множество операций, являющихся высказывательными функциями. Обычно в качестве операций выбирают высказывательные функции, которые позволяют вычислить любую истинностную функцию, как суперпозицию, используя только истинностные функции, обозначаемые атомарными формулами, в этом случае высказывательностные функции задают некоторый логический базис истинностных функций. Аналогичным образом, используя в качестве носителя множество формул и или множество классов формул можно построить соответственно алгебру формул логики высказываний или, например, алгебру Линденбаума — Тарского.

2.2.9 Булева алгебра высказываний

Если в соответствующий логический базис выбрать операции, которые удовлетворяют следующим свойствам, тогда можно говорить о булевой алгебре.
Ассоциативность.
$$\left(\psi 0\left(\left\langle \chi, \psi 0\left(\left\langle \delta, \lambda \right\rangle\right)\right\rangle\right) = \psi 0\left(\left\langle \psi 0\left(\left\langle \chi, \delta \right\rangle\right), \lambda \right\rangle\right)\right)$$
$$\left(\psi 1\left(\left\langle \chi, \psi 1\left(\left\langle \delta, \lambda \right\rangle\right)\right\rangle\right) = \psi 1\left(\left\langle \psi 1\left(\left\langle \chi, \delta \right\rangle\right), \lambda \right\rangle\right)\right)$$
Коммутативность.
$$\left(\psi 0\left(\left\langle \chi, \lambda \right\rangle\right) = \psi 0\left(\left\langle \lambda, \chi \right\rangle\right)\right)$$
$$\left(\psi 1\left(\left\langle \chi, \lambda \right\rangle\right) = \psi 1\left(\left\langle \lambda, \chi \right\rangle\right)\right)$$
Дистрибутивность.
$$\left(\psi 0\left(\left\langle \chi, \psi 1\left(\left\langle \delta, \lambda \right\rangle\right)\right\rangle\right) = \psi 1\left(\left\langle \psi 0\left(\left\langle \chi, \delta \right\rangle\right), \psi 0\left(\left\langle \chi, \lambda \right\rangle\right)\right\rangle\right)\right)$$
$$\left(\psi 1\left(\left\langle \chi, \psi 0\left(\left\langle \delta, \lambda \right\rangle\right)\right\rangle\right) = \psi 0\left(\left\langle \psi 1\left(\left\langle \chi, \delta \right\rangle\right), \psi 1\left(\left\langle \chi, \lambda \right\rangle\right)\right\rangle\right)\right)$$
Идемпотентность.
$$\left(\psi 0\left(\left\langle \chi, \chi \right\rangle\right) = \chi\right)$$
$$\left(\psi 1\left(\left\langle \chi, \chi \right\rangle\right) = \chi\right)$$
Двойное отрицание.
$$\left(\psi 2\left(\psi 2\left(\chi\right)\right) = \chi\right)$$
Правила де Моргана.
$$\left(\psi 2\left(\left\langle \chi, \psi 0\left(\left\langle \delta, \lambda \right\rangle\right)\right\rangle\right) = \psi 1\left(\left\langle \psi 2\left(\left\langle \chi, \delta \right\rangle\right), \psi 2\left(\left\langle \chi, \lambda \right\rangle\right)\right\rangle\right)\right)$$
$$\left(\psi 2\left(\left\langle \chi, \psi 1\left(\left\langle \delta, \lambda \right\rangle\right)\right\rangle\right) = \psi 0\left(\left\langle \psi 2\left(\left\langle \chi, \delta \right\rangle\right), \psi 2\left(\left\langle \chi, \lambda \right\rangle\right)\right\rangle\right)\right)$$
Существуют такие O и I, что выполняются следующие свойства.

Свойства констант.
$\left(\psi 0\left(\langle\chi,O\rangle\right) = O\right)$
$\left(\psi 0\left(\langle\chi,I\rangle\right) = \chi\right)$
$\left(\psi 1\left(\langle\chi,O\rangle\right) = \chi\right)$
$\left(\psi 1\left(\langle\chi,I\rangle\right) = I\right)$
$\left(\psi 2\left(I\right) = O\right)$
$\left(\psi 2\left(O\right) = I\right)$
Закон противоречия.
$\left(\psi 0\left(\langle\chi,\psi 2(\chi)\rangle\right) = O\right)$
Закон «исключенного третьего».
$\left(\psi 1\left(\langle\chi,\psi 2(\chi)\rangle\right) = I\right)$
Примерами таких операций являются функции конъюнкции, дизъюнкции и отрицания, для которых $(O = \perp)$ и $(I = \bullet)$.
Таким образом, можно говорить о булевой алгебре множеств, булевой алгебре двоичных векторов, булевой алгебре логики, булевой алгебре высказываний.
Рассмотрим булеву алгебру истинностных функций одного аргумента, являющуюся подалгеброй булевой алгебры высказываний. Носитель этой алгебры есть

$$\{\varphi 0, \varphi 1, \varphi 2, \varphi 3\},$$

элементы которого являются следующими функциями.

$$\left(\varphi 0 = \{\langle\perp,\perp\rangle,\langle\bullet,\perp\rangle\}\right)$$
$$\left(\varphi 1 = \{\langle\perp,\perp\rangle,\langle\bullet,\bullet\rangle\}\right)$$
$$\left(\varphi 2 = \{\langle\perp,\bullet\rangle,\langle\bullet,\perp\rangle\}\right)$$
$$\left(\varphi 3 = \{\langle\perp,\bullet\rangle,\langle\bullet,\bullet\rangle\}\right)$$

Сигнатура этой алгебры состоит из следующих операций.

$$\begin{aligned}&\Big(\psi 0 = \{\langle\langle\varphi 0,\varphi 0\rangle,\varphi 0\rangle,\langle\langle\varphi 0,\varphi 1\rangle,\varphi 0\rangle,\langle\langle\varphi 0,\varphi 2\rangle,\varphi 0\rangle,\langle\langle\varphi 0,\varphi 3\rangle,\varphi 0\rangle,\\&\langle\langle\varphi 1,\varphi 0\rangle,\varphi 0\rangle,\langle\langle\varphi 1,\varphi 1\rangle,\varphi 1\rangle,\langle\langle\varphi 1,\varphi 2\rangle,\varphi 0\rangle,\langle\langle\varphi 1,\varphi 3\rangle,\varphi 1\rangle,\\&\langle\langle\varphi 2,\varphi 0\rangle,\varphi 0\rangle,\langle\langle\varphi 2,\varphi 1\rangle,\varphi 0\rangle,\langle\langle\varphi 2,\varphi 2\rangle,\varphi 2\rangle,\langle\langle\varphi 2,\varphi 3\rangle,\varphi 2\rangle,\\&\langle\langle\varphi 3,\varphi 0\rangle,\varphi 0\rangle,\langle\langle\varphi 3,\varphi 1\rangle,\varphi 1\rangle,\langle\langle\varphi 3,\varphi 2\rangle,\varphi 2\rangle,\langle\langle\varphi 3,\varphi 3\rangle,\varphi 3\rangle\}\Big)\end{aligned}$$

$$\begin{aligned}&\Big(\psi 1 = \{\langle\langle\varphi 0,\varphi 0\rangle,\varphi 0\rangle,\langle\langle\varphi 0,\varphi 1\rangle,\varphi 1\rangle,\langle\langle\varphi 0,\varphi 2\rangle,\varphi 2\rangle,\langle\langle\varphi 0,\varphi 3\rangle,\varphi 3\rangle,\\&\langle\langle\varphi 1,\varphi 0\rangle,\varphi 1\rangle,\langle\langle\varphi 1,\varphi 1\rangle,\varphi 1\rangle,\langle\langle\varphi 1,\varphi 2\rangle,\varphi 3\rangle,\langle\langle\varphi 1,\varphi 3\rangle,\varphi 3\rangle,\\&\langle\langle\varphi 2,\varphi 0\rangle,\varphi 2\rangle,\langle\langle\varphi 2,\varphi 1\rangle,\varphi 3\rangle,\langle\langle\varphi 2,\varphi 2\rangle,\varphi 2\rangle,\langle\langle\varphi 2,\varphi 3\rangle,\varphi 3\rangle,\\&\langle\langle\varphi 3,\varphi 0\rangle,\varphi 3\rangle,\langle\langle\varphi 3,\varphi 1\rangle,\varphi 3\rangle,\langle\langle\varphi 3,\varphi 2\rangle,\varphi 3\rangle,\langle\langle\varphi 3,\varphi 3\rangle,\varphi 3\rangle\}\Big)\end{aligned}$$

$$\left(\psi 2 = \{\langle\varphi 0,\varphi 3\rangle,\langle\varphi 1,\varphi 2\rangle,\langle\varphi 2,\varphi 1\rangle,\langle\varphi 3,\varphi 0\rangle\}\right)$$

Тогда все истинностные функции этой алгебры с помощью операций этой алгебры можно получить как суперпозиции, используя только функцию $\varphi 1$.
Действительно:

$$\left(\varphi 2 = \ \psi 3(\varphi 1)\right)$$
$$\left(\varphi 0 = \ \psi 0(\langle \varphi 1, \varphi 2 \rangle)\right)$$
$$\left(\varphi 3 = \ \psi 3(\varphi 0)\right)$$
$$\left(\varphi 3 = \ \psi 1(\langle \varphi 1, \varphi 2 \rangle)\right)$$

2.2.10 Нормальные формы и совершенные нормальные формы

Так как операции конъюнкции, дизъюнкции и отрицания образуют логический базис, то любая истинностная функция является их суперпозицией, т.е. может быть выражена через эти операции. Следовательно, для любой истинностной функции можно построить обозначающую её формулу языка логики высказываний. В связи с этим выделяют классы логических формул, называемых нормальными формами. Различают дизъюнктивную нормальную форму и конъюнктивную нормальную форму.

Элементарной конъюнкцией (конъюнктом) будем называть формулу, подформулами которой являются только конъюнктивные формулы, либо – атомарные формулы, либо – отрицания атомарных формул.

Элементарной дизъюнкцией (дизъюнктом) будем называть формулу, подформулами которой являются только дизъюнктивные формулы, либо – атомарные формулы, либо – отрицания атомарных формул.

Дизъюнктивной нормальной формой (ДНФ) будем назвать формулу, подформулами которой являются дизъюнктивные формулы, либо – элементарные конъюнкции.

Конъюнктивной нормальной формой (КНФ) будем назвать формулу, подформулами которой являются конъюнктивные формулы, либо – элементарные дизъюнкции.

Совершенная дизъюнктивная нормальная форма (СДНФ) – это ДНФ, каждая элементарная конъюнкция которой, содержит одинаковые множества равных атомарных подформул.

Совершенная конъюнктивная нормальная форма (СКНФ) – это КНФ, каждая элементарная дизъюнкция которой, содержит одинаковые множества равных атомарных подформул.

Любая необщезначимая формула имеет равносильную СКНФ.

Любая выполнимая формула имеет равносильную СДНФ.

Любая истинностная функция с конечным числом аргументов представима в виде СДНФ или СКНФ.

В качестве основы для доказательства этих утверждений можно предложить следующую процедуру построения СДНФ и СКНФ. Процедура заключается в следующем: если задана формула, то строим таблицу истинности для этой формулы, если функция известна, также представляем эту функцию в виде таблицы истинности.

Далее, для построения СКНФ рассматриваем все строки таблицы, ячейки которых содержат $\perp$ для значения рассматриваемой функции. Для каждой строки строим элементарную дизъюнкцию, которая включает в качестве недизъюнктивных подформул только атомарные формулы, имеющию в интерпретации, соответствующей этой строке, значение $\perp$, и отрицания атомарных формул, имеющей в той же интерпретации значение • . СКНФ получается как конъюнкция, неконъюнктивными подформулами которой являются только все такие элементарные дизъюнкции. Для построения СДНФ рассматриваются все строки таблицы, ячейки которых содержат • для значения рассматриваемой функции. Для каждой строки строится элементарная конъюнкция, которая включает в качестве неконъюнктивных подформул только атомарные формулы, имеющию в интерпретации, соответствующей этой строке, значение • , и отрицания атомарных формул, имеющей в той же интерпретации значение $\perp$. СДНФ получается как дизъюнкция, недизъюнктивными подформулами которой являются только все такие элементарные конъюнкции.

Рассмотрим пример.

P	Q	R	$(Q \vee R)$	$(P \wedge (Q \vee R))$
$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
$\perp$	$\perp$	•	•	$\perp$
$\perp$	•	$\perp$	•	$\perp$
$\perp$	•	•	•	$\perp$
•	$\perp$	$\perp$	$\perp$	$\perp$
•	$\perp$	•	•	•
•	•	$\perp$	•	•
•	•	•	•	•

СКНФ и СДНФ для этого примера выглядят соответственно следующим образом.

$$\left(\left(\left(\left((P \vee Q) \vee R\right) \wedge \left((P \vee Q) \vee (\neg R)\right)\right) \wedge \left(\left((P \vee (\neg Q)) \vee R\right) \wedge \left((P \vee (\neg Q)) \vee (\neg R)\right)\right)\right) \wedge \left(((\neg P) \vee Q) \vee R\right)\right)$$

$$\left(\left(\left((P \wedge (\neg Q)) \wedge R\right) \vee \left((P \wedge Q) \wedge (\neg R)\right)\right) \vee \left((P \wedge Q) \wedge R\right)\right)$$

2.2.11 Логические уравнения и системы логических уравнений

Как уже упоминалось ранее одной из основных задач логики является задача выяснения того, в каких тех или иных случаях истинна или ложна та или иная формула. Такая задача может быть задана логическим уравнением. Логическое уравнение состоит из двух формул, связанных знаком равенства. При решении задачи необходимо найти все такие интерпретации, для которых значения функций, заданных этими формулами равны. Если есть две формулы α и β, то уравнение выглядит следующим образом.

$$(\alpha = \beta)$$

Следует заметить, что задачи проверки формул на равносильность и выявления класса формулы легко сводятся или являются частным случаем задачи решения логического уравнения. Кроме того, задача решения логического уравнения легко сводится к задаче установления выполнимости логической формулы, а задача проверки равносильности формул $(\alpha \Leftrightarrow \beta)$ сводится к задаче проверки формулы на общезначимость (‘ $(\alpha \sim \beta)$).

Одним из способов решения логических уравнений в алгебре высказываний является табличный способ, сводящийся к построению истинностных таблиц. Так как этот способ может занимать большое время, применяются и другие способы. Например, так называемый, «способ решения логических уравнений», основанный на определениях логических связок. Рассмотрим этот способ на примере. Пусть есть следующее уравнение.

$((P \vee Q) = (P \wedge Q))$

Его решение сводится к следующему.

$$((P \vee Q) = (P \wedge Q)) \Leftrightarrow \left[\begin{array}{l} \left\{\begin{array}{l} ((P \vee Q) = \perp) \\ ((P \wedge Q) = \perp) \end{array}\right. \\ \left\{\begin{array}{l} ((P \vee Q) = \bullet) \\ ((P \wedge Q) = \bullet) \end{array}\right. \end{array}\right. \Leftrightarrow \left[\begin{array}{l} \left\{\begin{array}{l} \left\{\begin{array}{l} (P = \perp) \\ (Q = \perp) \end{array}\right. \\ \left[\begin{array}{l} (P = \perp) \\ (Q = \perp) \end{array}\right. \end{array}\right. \\ \left\{\begin{array}{l} \left[\begin{array}{l} (P = \bullet) \\ (Q = \bullet) \end{array}\right. \\ \left\{\begin{array}{l} (P = \bullet) \\ (Q = \bullet) \end{array}\right. \end{array}\right. \end{array}\right. \Leftrightarrow \left[\begin{array}{l} \left[\begin{array}{l} \left\{\begin{array}{l} (P = \perp) \\ (Q = \perp) \\ (P = \perp) \end{array}\right. \\ \left\{\begin{array}{l} (P = \perp) \\ (Q = \perp) \\ (Q = \perp) \end{array}\right. \end{array}\right. \\ \left[\begin{array}{l} \left\{\begin{array}{l} (P = \bullet) \\ (P = \bullet) \\ (Q = \bullet) \end{array}\right. \\ \left\{\begin{array}{l} (Q = \bullet) \\ (P = \bullet) \\ (Q = \bullet) \end{array}\right. \end{array}\right. \end{array}\right. \Leftrightarrow \left[\begin{array}{l} \left[\begin{array}{l} \left\{\begin{array}{l} (P = \perp) \\ (Q = \perp) \end{array}\right. \\ \left\{\begin{array}{l} (P = \perp) \\ (Q = \perp) \end{array}\right. \end{array}\right. \\ \left[\begin{array}{l} \left\{\begin{array}{l} (P = \bullet) \\ (Q = \bullet) \end{array}\right. \\ \left\{\begin{array}{l} (P = \bullet) \\ (Q = \bullet) \end{array}\right. \end{array}\right. \end{array}\right. \Leftrightarrow \left[\begin{array}{l} \left\{\begin{array}{l} (P = \perp) \\ (Q = \perp) \end{array}\right. \\ \left\{\begin{array}{l} (P = \bullet) \\ (Q = \bullet) \end{array}\right. \end{array}\right.$$

Таким образом, ответ, являющийся множеством всех интерпретаций, для которых выполняется равенство, выглядит так: $\{\{\langle P, \perp\rangle, \langle Q, \perp\rangle\}, \{\langle P, \bullet\rangle, \langle Q, \bullet\rangle\}\}$.

2.2.12 Равносильные преобразования

Третьим методом решения задач алгебры высказываний является так называемый метод равносильных или – в силу сводимости этой задачи к проверке формулы со связкой эквиваленции на общезначимость – эквивалентных преобразований. Этот метод использует основанные на определениях логических связок, свойствах операций булевой алгебры и следствиях из них следующие отношения равносильности между формулами, для замены одних подформул в формулах другими. Путём этой замены и выявляется, например, общезначимость или противоречивость формулы.

Ассоциативность.

$((\alpha \wedge (\beta \wedge \gamma)) \Leftrightarrow ((\alpha \wedge \beta) \wedge \gamma))$

$((\alpha \vee (\beta \vee \gamma)) \Leftrightarrow ((\alpha \vee \beta) \vee \gamma))$

Коммутативность.

$((\alpha \wedge \beta) \Leftrightarrow (\beta \wedge \alpha))$

$((\alpha \vee \beta) \Leftrightarrow (\beta \vee \alpha))$

Дистрибутивность.

$((\alpha \wedge (\beta \vee \gamma)) \Leftrightarrow ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma)))$

$((\alpha \vee (\beta \wedge \gamma)) \Leftrightarrow ((\alpha \vee \beta) \wedge (\alpha \vee \gamma)))$

Идемпотентность.

$((\alpha \wedge \alpha) \Leftrightarrow \alpha)$

$((\alpha \vee \alpha) \Leftrightarrow \alpha)$

Двойное отрицание.

$((\neg(\neg\alpha)) \Leftrightarrow \alpha)$

Правила де Моргана.

$((\neg(\alpha \wedge \beta)) \Leftrightarrow ((\neg\alpha) \vee (\neg\beta)))$

$((\neg(\alpha \vee \beta)) \Leftrightarrow ((\neg\alpha) \wedge (\neg\beta)))$

Свойства констант.

$((\alpha \wedge \perp) \Leftrightarrow \perp)$

$((\alpha \wedge \bullet) \Leftrightarrow \alpha)$

$((\alpha \vee \perp) \Leftrightarrow \alpha)$

$((\alpha \vee \bullet) \Leftrightarrow \bullet)$

$((\neg \perp) \Leftrightarrow \bullet)$

$((\neg \bullet) \Leftrightarrow \perp)$

Закон противоречия.

$((\alpha \wedge (\neg\alpha)) \Leftrightarrow \perp)$

Закон «исключённого третьего».

$((\alpha \vee (\neg\alpha)) \Leftrightarrow \bullet)$

Выражение связок.

$((\alpha \to \beta) \Leftrightarrow ((\neg\alpha) \to \beta))$

$((\alpha \sim \beta) \Leftrightarrow ((\alpha \to \beta) \wedge (\beta \to \alpha)))$

Следующие равносильные преобразования можно доказать используя предыдущие.

Поглощение.

$((\alpha \wedge (\alpha \vee \beta)) \Leftrightarrow \alpha)$

$((\alpha \vee (\alpha \wedge \beta)) \Leftrightarrow \alpha)$

Склеивание

$(((\alpha \vee \beta) \wedge (\alpha \vee (\neg\beta))) \Leftrightarrow \alpha)$

$(((\alpha \vee \beta) \vee (\alpha \wedge (\neg\beta))) \Leftrightarrow \alpha)$

и другие.

Обычно замену производят в следующем порядке: вначале избавляются от эквиваленций и импликаций заменяя их на другие связки. Потом, используя свойства и правила де Моргана, двойного отрицания, дистрибутивности, коммутативности, ассоциативности, идемпотентности и свойства констант, переходят к КНФ или ДНФ, а затем уже от них, если требуется, переходят к СКНФ или СДНФ.

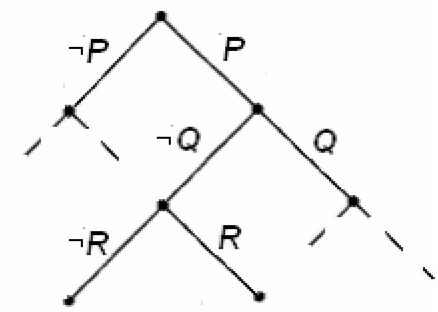
Применим метод равносильных преобразований для формулы $((P \vee Q) \sim (P \wedge Q))$.

$$((P\vee Q)\sim(P\wedge Q))\Leftrightarrow(((P\vee Q)\to(P\wedge Q))\wedge((P\wedge Q)\to(P\vee Q)))\Leftrightarrow$$
$$(((\neg(P\vee Q))\vee(P\wedge Q))\wedge((\neg(P\wedge Q))\vee(P\vee Q)))\Leftrightarrow(((\neg(P\vee Q))\vee(P\wedge Q))\wedge((\neg(P\wedge Q))\vee(P\vee Q)))\Leftrightarrow$$
$$((((\neg P)\wedge(\neg Q))\vee(P\wedge Q))\wedge(((\neg P)\vee(\neg Q))\vee(P\vee Q)))\Leftrightarrow$$
$$((((\neg P)\wedge(\neg Q))\vee(P\wedge Q))\wedge(((\neg P)\vee(\neg Q))\vee(Q\vee P)))\Leftrightarrow$$
$$((((\neg P)\wedge(\neg Q))\vee(P\wedge Q))\wedge((\neg P)\vee((\neg Q)\vee(Q\vee P))))\Leftrightarrow$$
$$((((\neg P)\wedge(\neg Q))\vee(P\wedge Q))\wedge((\neg P)\vee(((\neg Q)\vee Q)\vee P)))\Leftrightarrow$$
$$((((\neg P)\wedge(\neg Q))\vee(P\wedge Q))\wedge((\neg P)\vee((Q\vee(\neg Q))\vee P)))\Leftrightarrow$$
$$((((\neg P)\wedge(\neg Q))\vee(P\wedge Q))\wedge((\neg P)\vee(\bullet\vee P)))\Leftrightarrow((((\neg P)\wedge(\neg Q))\vee(P\wedge Q))\wedge((\neg P)\vee(P\vee\bullet)))\Leftrightarrow$$
$$((((\neg P)\wedge(\neg Q))\vee(P\wedge Q))\wedge((\neg P)\vee\bullet))\Leftrightarrow((((\neg P)\wedge(\neg Q))\vee(P\wedge Q))\wedge\bullet)\Leftrightarrow(((\neg P)\wedge(\neg Q))\vee(P\wedge Q))$$

2.2.13 Алгоритм Куайна

Поиск быстрых методов и алгоритмов решения логических уравнений является вторичной, но не менее важной задачей математической логики. В поисках более быстрых методов решения логических равнений, разрабатывались другие подходы. Например, следующий способ, основывается на рассмотрении дерева возможных решений (семантического дерева) и фактически является реализацией метода ветвей и границ для логических формул.

Рассмотрим алгоритм Куайна для проверки выполнимости формул на примере выполнимости формулы $(P\wedge(Q\vee R))$. Каждая ветвь в дереве решений соответствует тому или иному значению какой-либо атомарной подформулы. Будем отмечать ветви формулами, которые истинны при выбранном значении атомарной формулы.



На рисунке изображены те ветви, которые надо рассмотреть, чтобы выявить выполнимость формулы. Ветви, изображённые штриховыми линиями, анализировать не имеет смысла. Алгоритм Куайна как раз и заключается в рассмотрении только тех ветвей дерева, которые имеет смысл анализировать. В зависимости от порядка построения дерева количество ветвей может сильно варьироваться, и в худшем случае алгоритм будет ненамного эффективнее, чем алгоритм, использующий табличный способ.

2.2.14 Алгоритм Девиса-Патнема

Этот алгоритм рассматривается для формул, приведённых к КНФ, которая может быть получена в результате применения равносильных преобразований. Алгоритм Куайна сильно упрощается при применении к КНФ.

Допустим, что есть КНФ α, ни один дизъюнкт которой не содержит одновременно подформулы β и $(\neg\beta)$. Тогда для любой атомарной подформулы β формулы α множество всех дизъюнктов формулы α, разбивается на дизъюнкты, подформулой которых является β, на дизъюнкты подформулой которых является $(\neg\beta)$ и на остальные дизъюнкты. Алгоритм Куйана для КНФ сводится к следующему.

Рассматривается множество дизъюнктов S формулы α.

Выбирается атомарная подформула β формулы α.

Формируется множество дизъюнктов, содержащих β, множество дизъюнктов, содержащих $(\neg\beta)$, и множество остальных дизъюнктов C.

Из первых двух множеств формируется два новых множества дизъюнктов A и B, которые получаются путём исключения из каждого дизъюнкта соответствующих множеств формул β и $(\neg\beta)$ соответственно.

Тогда исходная формула α невыполнима тогда и только тогда, когда невыполнимы КНФ содержащие все дизъюнкты множеств $(A\cup C)$ и $(B\cup C)$. Если для A или B были соответственно исключены β и $(\neg\beta)$, то соответствующую КНФ следует считать невыполнимой. Алгоритм рекурсивен.

Алгоритм Девиса и Патнема базируется на использовании КНФ исследуемой формулы –для выбора исключаемых подформул β в оптимальном порядке. При этом не строятся бесполезные ветви дерева решений. Приоритетный выбор β удобен в двух следующих случаях:

- S содержит один из дизъюнктов с подформулой β или $(\neg\beta)$;
- только одна из формул β и $(\neg\beta)$ входит в S.

Например, предположим, что S содержит дизъюнкт β**.** Для множества A будет исключён этот дизъюнкт β и, следовательно, КНФ для $(A\cup C)$ будет невыполнимой. Таким образом, возможная невыполнимость для S сводится к невыполнимости для $(B\cup C)$. С другой стороны, если β входит в S, тогда как $(\neg\beta)$ не входит, то B пусто. Возможная невыполнимость для S сводится к невыполнимости КНФ для $(A\cup C)$.

Эта простая стратегия сильно увеличивает эффективность алгоритма. Всякий раз, когда она применима, она сводит проблему невыполнимости для S к проблеме невыполнимости только для одного из множеств $(A\cup C)$ и $(B\cup C)$, более простых, чем S.

2.2.15 Неклаузальное правило резолюций

Преобразование формулы к КНФ, предшествующее доказательству, может оказаться довольно трудоемким. Неклаузальная резолюция (т.е. резолюция, ориентированная для применения не только к нормальным формам) является ещё одним из подходов к решению логических задач и предназначена для распространения механизма доказательства на произвольные логические формулы. В частности, это позволяет использовать резолюцию как средство в доказательствах, проводимых по принципу дедукции, даже если гипотезы и. отрицание заключения не являются дизъюнктами.

Пусть α и β – формулы исчисления высказываний, возможно содержащие подформулу γ, что будем обозначать так: $\alpha(\gamma)$ и $\beta(\gamma)$. Тогда с учётом обозначения формулы $\alpha(\delta)$, в которой все имеющиеся подформулы γ заменяются на формулу δ, введем следующие определения операторов резолюции и согласия соответственно.

$$\left(\perp\left(\langle\alpha(\gamma),\beta(\gamma)\rangle\right) = \left(\alpha(\perp)\vee\beta(\bullet)\right)\right)$$

$$\left(\bullet\ \left(\langle\alpha(\gamma),\beta(\gamma)\rangle\right) = \left(\alpha(\perp)\wedge\beta(\bullet)\right)\right)$$

Оператор резолюции применяется для доказательства невыполнимости множества логических формул, оператор согласия – для доказательства общезначимости. Оба оператора позволяют из исходных формул получать новые, более простые, которые с помощью равносильных преобразований, описывающих свойства констант, в случае успеха, позволяют прийти к формуле $\perp$ или $\bullet$ соответственно.

2.2.16 Хорновские дизъюнкты, модель и минимальная модель

Для некоторых классов формул существуют довольно быстрые алгоритмы доказательства. К таким классам относится класс формул, называемых хорновскими дизъюнктами.

Хорновский дизъюнкт это такой дизъюнкт, который содержит в качестве подформулы не более одной атомарной формулы, отрицание которой не является подформулой этого дизъюнкта.

Для множеств таких дизъюнктов, в силу их структурных свойств, легко строить довольно быстрые алгоритмы, например, проверки выполнимости. Эти алгоритмы применяются в таких системах логического программирования как, например, Prolog.

Под моделью конъюнктивной формулы в логике высказываний будем понимать множество тех и только тех элементов области определения интерпретации формулы, для которой значение функции обозначаемой формулой принимает значение $\bullet$, и которые имеют значение $\bullet$ в этой интерпретации.

Под моделью множества формул будем понимать модель конъюнкции всех формул из этого множества и только этих формул.

Под минимальной моделью формулы или множества формул будем понимать модель соответственно этой формулы или этого множества формул, равную пересечению некоторого множества моделей этой формулы или этого множества формул соответственно.

Например, формула $(P\vee Q)$ имеет три модели, две первые из которых минимальные.

$$\{P\}$$
$$\{Q\}$$
$$\{P, Q\}$$

Любое выполнимое конечное множество хорновских дизъюнктов имеет ровно одну минимальную модель.

Таким обрзом, добавление новых хорновских дизъюнктов к исходному множеству хорновских дизъюнктов при сохраненнии его выполнимости приводит к тому, что новая минимальная модель является расширением старой (включает старую), что позволяет упростить её вычисление. Это свойство множеств хорновских дизъюнктов позволяет также применять при решении логических задач для множеств хорновских дизъюнктов гипотезу замкнутого мира, т.е. когда считаются истинными все высказывания, которые записаны явно, либо которые следуют записанных явно высказываний – все остальные высказывания считаются ложными. Это бывает часто удобно при разработке программных систем, использующих логический подход к решению задач. Гипотеза замкнутости мира узаконивает интересную семантику понятия отрицания в логическом программировании, позволяя упростить построение решения логической задачи. Однако, следует отметить, что гипотеза замкнутости мира противоположна такому необходимому свойству интеллектуальной системы как открытость.

2.3 Исчисление высказываний.

2.3.1 Формальные теории

Развитие методов решения, в частности, логических задач привело к тому, что решение логических задач может проводиться без апелляции к семантике рассматриваемых формул, т.е. без построения интерпретаций этих формул, ограничиваясь только рассмотрением структуры этих формул. Для этого вводится понятие исчисления (формальной аксиоматической теории), которое является подмножеством формул некоторого языка. Обычно это множество включает только общезначимые формулы. Исчисление задаётся четырьмя компонентами (вектора):

- алфавитом языка,
- грамматикой языка,
- множеством формул, называемых аксиомами,
- множеством правил (логического) вывода.

Множество аксиом является подмножеством исчисления. Так как исчисление может содержать бесконечное число формул, то правила вывода позволяют из аксиом вывести остальные формулы исчисления. Одно и то же исчисление может быть задано разными наборами аксиом и правил вывода. Например, множество аксиом может быть бесконечным (в этом случае множество аксиом задаётся с помощью так называемых аксиомных схем (аксиоматики)), и множество правил вывода содержать только одно правило вывода, либо –

множество аксиом может быть пустым, а множество правил вывода может содержать несколько правил вывода.
Исчисление высказываний задаётся алфавитом и грамматикой языка высказываний, которые были определены выше, а также множеством аксиом и правил вывода. Далее приведен один из вариантов множества аксиом и правил вывода.

2.3.2 Аксиоматика

Зададим с помощью аксиомных схем множество аксиом исчисления высказываний. Отличие аксиомной схемы от аксиомы в том, что, как это обычно бывает при определениях, аксиомная схема записывается на некотором метаязыке, формулы которого содержат метапеременные (лексические переменные). Любая аксиома может быть получена путём одновременной замены каждой метапеременной во всех её вхождениях в аксиомную схему на конкретную формулу соответствующего языка (в данном случае – языка логики высказываний). В данном случае метапеременные обозначены символами греческого алфавита.
Ниже приведены аксиомные схемы, задающие аксиомы для введения и удаления логических связок по отношению к формулам, являющимися подформулами этих аксиом.

	введение	удаление
$\rightarrow$	$(\alpha \rightarrow (\beta \rightarrow \alpha))$	$((\gamma \rightarrow \alpha) \rightarrow ((\gamma \rightarrow (\alpha \rightarrow \beta)) \rightarrow (\gamma \rightarrow \beta)))$
$\wedge$	$(\alpha \rightarrow (\beta \rightarrow (\alpha \wedge \beta)))$	$((\alpha \wedge \beta) \rightarrow \alpha)$ $((\alpha \wedge \beta) \rightarrow \beta)$
$\vee$	$(\alpha \rightarrow (\alpha \vee \beta))$ $(\alpha \rightarrow (\beta \vee \alpha))$	$((\alpha \rightarrow \gamma) \rightarrow ((\beta \rightarrow \gamma) \rightarrow ((\alpha \vee \beta) \rightarrow \gamma)))$
$\neg$	$((\alpha \rightarrow \beta) \rightarrow ((\alpha \rightarrow (\neg \beta)) \rightarrow (\neg \alpha)))$	$((\neg(\neg \alpha)) \rightarrow \alpha)$
$\sim$	$((\alpha \rightarrow \beta) \rightarrow ((\beta \rightarrow \alpha) \rightarrow (\alpha \sim \beta)))$	$((\alpha \sim \beta) \rightarrow (\alpha \rightarrow \beta))$ $((\alpha \sim \beta) \rightarrow (\beta \rightarrow \alpha))$

Названия этих аксиомных схем следующие: «аксиомная схема введения импликации», «аксиомная схема удаления импликации», «аксиомная схема введения конъюнкции» и т.д.

2.3.4 Правила вывода

Над формулами исчисления с помощью правил вывода задаётся отношение выводимости ((логического) вывода) $\mathfrak{h}$. Рассмотрим следующее правило вывода, которое называется правилом прямого заключения или правилом Modus Ponens и описано на метаязыке, в следующей формулировке.
Для любого множества G

$$G \subseteq \mathbf{L}$$

и для любых формул α и β языка $\mathbf{L}$, выполняется следующее:

$$((G \cup \{\alpha, (\alpha \rightarrow \beta)\}) \,\mathfrak{h}\, (G \cup \{\beta\})),$$

кроме вышеприведённого свойства отношение вывода является транзитивным отношением и обладает следующим свойством:

$$((\mathcal{G} \cup \mathcal{F}) \vdash \mathcal{F}),$$

где $\mathcal{G}$ и $\mathcal{F}$ произвольные множества формул языка $\mathbf{L}$.

2.3.5 Формальный вывод

Под формальным (логическим) выводом формулы β из множества посылок $\mathcal{G}$ понимается вектор (последовательность) логических формул, каждый i-й компонент которого является аксиомой или элементом множества $\mathcal{G}$, либо – формулой, которая выводима (получена путём применения правила вывода) из множества формул, состоящего из всех таких формул и только таких формул, которые являются k-ми компонентами этого вектора, где $(k < i)$, а формула β является последним компонентом этого вектора.

Формула β является теоремой (теоремой исчисления) тогда и только тогда, когда существует формальный вывод формулы β из множества посылок, являющегося подмножеством аксиом этого исчисления.

Далее будем использовать следующие эквивалентные обозначения отношения выводимости.

$((\vdash_{\mathcal{A}} \mathcal{F}) \sim (\mathcal{A} \vdash \mathcal{F}))$

$((\mathcal{G} \vdash_{\mathcal{A}} \mathcal{F}) \sim ((\mathcal{A} \cup \mathcal{G}) \vdash \mathcal{F}))$

Рассмотрим примеры формального вывода.

Пример формального вывода (доказательства) теоремы $(P \to P)$ в заданном исчислении высказываний.

$\Big\langle (P \to (P \to P)), ((P \to (P \to P)) \to ((P \to ((P \to P) \to P)) \to (P \to P))), (P \to ((P \to P) \to P)),$

$((P \to ((P \to P) \to P)) \to (P \to P)), (P \to P) \Big\rangle$

Оформим этот вывод следующим образом.

$1 : \langle + \to \rangle \vdash 1 (P \to (P \to P))$

$2 : \langle - \to \rangle \vdash ((P \to (P \to P)) \to ((P \to ((P \to P) \to P)) \to (P \to P)))$

$3 : \langle + \to \rangle \vdash (P \to ((P \to P) \to P))$

$4 : \langle 1, 2 \rangle \vdash ((P \to ((P \to P) \to P)) \to (P \to P))$

$5 : \langle 3, 4 \rangle \vdash (P \to P)$

Это означает, что первая и третья формулы получены из аксиомной схемы введения импликации, вторая – из аксимоной схемы удаления импликации, четвёртая и пятая – по правилу прямого заключения, используя соответственно пары формул $\langle 1,2 \rangle$ и $\langle 3,4 \rangle$.

Аналогичным образом можно получить формальные выводы для формул $(P \sim P)$, $(P \to (\neg(\neg P)))$, $(P \vee (\neg P))$, $(\neg(P \wedge (\neg P)))$ и других.

2.3.6 Метатеоремы вывода

Кроме теорем исчисления могут рассматриваться некоторые формулы метаязыка, описывающие закономерности между произвольными формулами и отношениями для этого исчисления. Такие формулы включают метапеременные и в случае – когда могут быть доказаны – называются метатеоремами. Метатеоремы могут формулировать свойства отношения выводимости, дополнительные правила вывода или другие закономерности. Некоторые метатеоремы могут оказаться теоремами некоторого другого исчисления.

Рассмотрим метатеоремы, формулирующие следующие свойства отношения выводимости для любых множеств формул G, F и E языка $\mathbf{L}$.

Рефлексивность отношения выводимости.

$$(G \vdash G)$$

Монотонность отношения выводимости.

$$\left((G \vdash E) \to ((G \cup F) \vdash E)\right)$$

Транзитивность отношения выводимости.

$$\left(((G \vdash F) \wedge (F \vdash E)) \to (G \vdash E)\right)$$

Кроме этого справедливы следующие теоремы, являющиеся метатеоремами.

Теорема дедукции.

$$\left(\left((G \cup \{\alpha\}) \vdash \{\beta\}\right) \to \left(G \vdash \{(\alpha \to \beta)\}\right)\right)$$

Теорема обратная теореме дедукции.

$$\left(\left(G \vdash \{(\alpha \to \beta)\}\right) \to \left((G \cup \{\alpha\}) \vdash \{\beta\}\right)\right)$$

Метатеоремы, формулирующие дополнительные правила вывода.

	введение	удаление
$\to$	$\left(\left((G \cup \{\alpha\}) \vdash \{\beta\}\right) \to \left(G \vdash \{(\alpha \to \beta)\}\right)\right)$	$\left(\left(G \cup \{\alpha, (\alpha \to \beta)\}\right) \vdash \left(G \cup \{\beta\}\right)\right)$
$\wedge$	$\left(\left(G \cup \{\alpha, \beta\}\right) \vdash \left(G \cup \{(\alpha \wedge \beta)\}\right)\right)$	$\left(\left(G \cup \{(\alpha \wedge \beta)\}\right) \vdash \left(G \cup \{\alpha\}\right)\right)$ $\left(\left(G \cup \{(\alpha \wedge \beta)\}\right) \vdash \left(G \cup \{\beta\}\right)\right)$
$\vee$	$\left(\left(G \cup \{\alpha\}\right) \vdash \left(G \cup \{(\alpha \vee \beta)\}\right)\right)$ $\left(\left(G \cup \{\alpha\}\right) \vdash \left(G \cup \{(\beta \vee \alpha)\}\right)\right)$	$\left(\left(G \cup \{(\alpha \vee \beta), (\alpha \to \gamma), (\beta \to \gamma)\}\right) \vdash \left(G \cup \{\gamma\}\right)\right)$
$\neg$	$\left(\left(G \cup \{(\alpha \to \beta), (\alpha \to (\neg\beta))\}\right) \vdash \left(G \cup \{(\neg\alpha)\}\right)\right)$	$\left(\left(G \cup \{(\neg(\neg\alpha))\}\right) \vdash \left(G \cup \{\alpha\}\right)\right)$
$\sim$	$\left(\left(G \cup \{(\alpha \to \beta), (\beta \to \alpha)\}\right) \vdash \left(G \cup \{(\alpha \sim \beta)\}\right)\right)$	$\left(\left(G \cup \{(\alpha \sim \beta)\}\right) \vdash \left(G \cup \{(\alpha \to \beta)\}\right)\right)$ $\left(\left(G \cup \{(\alpha \sim \beta)\}\right) \vdash \left(G \cup \{(\beta \to \alpha)\}\right)\right)$

Отметим, что можно построить исчисление высказываний, используя в качестве аксиом пустое множество и только вышеперечисленные правила вывода.

2.3.7 Выводимость и общезначимость

Существует следующая взаимосвязь между понятиями выводимости и общезначимости для исчисления высказываний, выражаемая следующими метатеоремами.

$\left((\vdash_{L} F) \to (\models F)\right)$

$\left((\models F) \to (\vdash_{L} F)\right)$

2.3.8 Принцип резолюций в исчислениий высказываний

Ещё одной метатеоремой является такое правило вывода, как правило (клаузальной) резолюции, которое является описанием метода резолюций в применении к дизъюнктам.

Частными случаями этого правила являются:

$$\left(\left(G\cup\{(\alpha\vee\gamma)\}\cup\{(\beta\vee(\neg\gamma))\}\right)\mathfrak{h}\left(G\cup\{(\alpha\vee\beta)\}\right)\right)$$

$$\left(\left(G\cup\{(\alpha\vee\gamma)\}\cup\{(\neg\gamma)\}\right)\mathfrak{h}\left(G\cup\{\alpha\}\right)\right)$$

$$\left(\left(G\cup\{\gamma\}\cup\{(\alpha\vee(\neg\gamma))\}\right)\mathfrak{h}\left(G\cup\{\alpha\}\right)\right),$$

где G – произвольное множество формул языка $\mathbf{L}$. Формула, получаемая по правилу резолюции, называется резольвентой.

Алгоритм, основывающийся на этом правиле, быстр и удобен в применении к хорновским дизъюнктам.

2.3.9 Недостаточность исчисления высказываний

Рассмотрим следующее предложение.

«Не всё то золото, что блестит.»

Это предложение выражает высказывание, которое можно переформулировать так: «неверно, что если что-то блестит, то оно – золото». Трудность обозначения этого высказывания на языке логики высказываний в том, что если мы рассмотрим высказывательные предложения «что-то блестит» и «оно – золото» и обозначим их формулами соответственно P и Q, то попытка его записать приведёт к следующему:

$$(\neg(Q\to P)),$$

что равносильно формуле $(Q\wedge(\neg P))$. Однако с другой стороны нам известно, что что-то – золото и оно блестит: $(Q\wedge P)$ – исходя из взаимозаменяемости «оно» и «что-то». Как можно убедиться, конъюнкция этих формул невыполнима, однако с другой стороны для нас это противоречит нашему пониманию двух соответствующих высказываний, так как мы считаем и то и другое одновременно истинным. Значит, можно допустить, что мы ошиблись где-то при записи формулировок. В результате анализа можно прийти к выводу что, то золото, которое блестит, не является тем блестящим, которое не является золотом, иными словами одно блестящее и другое блестящее – это два разных блестящих. Тогда предложение можно попытаться записать так:

$$(\neg((Q1\to P1)\wedge(Q2\to P2))),$$

что теперь не противоречит формуле $(Q2\wedge P2)$. Однако, по мере того, как будет выясняться, что блестеть могут разные изотопы золота и т.п., число атомарных формул будет увеличиваться, и этот процесс может продолжаться до бесконечности. Длина (размерность) формулы не будет являться ограниченной. Чтобы упростить запись таких предложений, строится новый язык, в который дополнительно вводятся дополнительные средства (переменные и кванторы), которые позволяют сократить запись похожих формул, начинающую выглядеть следующим образом.

$$\left(\neg\left(\forall\chi\left(Q(\chi)\rightarrow P(\chi)\right)\right)\right)$$

3. Логика предикатов

3.1 Предикат

Предикатом n аргументов (n-местным предикатом) будем называть любую функцию являющуюся подмножеством n-арной операции φ вида $\left(\varphi\in \mathsf{A}^{\left(\mathrm{B}^{n}\right)}\right)$, где A – произвольное множество и $\left(\mathrm{B}=\{\perp,\bullet\}\right)$. Любой предикат двух и более аргументов характеризует отношение и называется характеристической функцией отношения. Например следующее отношение

$$\left(\mathsf{R}=\{\langle\gamma,\gamma\rangle,\langle\delta,\gamma\rangle,\langle\delta,\delta\rangle\}\right)$$

можно характеризовать следующим предикатом.

$$\left(R=\{\langle\langle\gamma,\gamma\rangle,\bullet\rangle,\langle\langle\gamma,\delta\rangle,\perp\rangle,\langle\langle\delta,\gamma\rangle,\bullet\rangle,\langle\langle\delta,\delta\rangle,\bullet\rangle\}\right)$$

Т.е. если предикат, характеризующий отношение, истинен для какого-либо аргумента, то этот аргумент принадлежит этому отношению, и если – ложен, то – не принадлежит.

3.2 Язык логики предикатов

3.2.1 Алфавит языка логики предикатов

По аналогии с языком логики высказываний зададим язык логики предикатов. Алфавит языка логики предикатов строится на основе алфавита языка логики высказываний и является его расширением.

$\langle small\ symbol\rangle ::= z\mid y\mid x\mid w\mid v\mid u\mid t$

$\langle variable\rangle ::= \langle small\ symbol\rangle\left[\{\langle digit\rangle\}\right]$

$\langle comma\rangle ::= ,$

$\langle for\ all\rangle ::= \forall$

$\langle exists\rangle ::= \exists$

$\langle quantor\rangle ::= \langle for\ all\rangle\mid\langle exists\rangle$

3.2.2 Синтаксис языка логики предикатов

Синтаксис языка логики предикатов строится независимо от синтаксиса языка логики высказываний и описывается следующей грамматикой.

$\langle atom \rangle ::= \langle symbol \rangle \left[\{ \langle natural \rangle \} \right]$

$\langle term \rangle ::= \langle variable \rangle$

$\langle term\ list \rangle ::= \langle left\ bracket \rangle \langle term \rangle \left[\{ \langle comma \rangle \langle term \rangle \} \right] \langle right\ bracket \rangle$

$\langle predicate\ term \rangle ::= \langle atom \rangle \left[\{ \langle term\ list \rangle \} \right]$

$\langle formula \rangle ::= \langle constant \rangle \mid \langle predicate\ term \rangle \mid \langle unary\ complex\ \ formula \rangle \mid \langle binary\ complex\ \ formula \rangle \mid \langle quantor\ \ formula \rangle$

$\langle unary\ complex\ \ formula \rangle ::= \langle left\ bracket \rangle \langle unary\ connective \rangle \langle formula \rangle \langle right\ bracket \rangle$

$\langle binary\ complex\ \ formula \rangle ::= \langle left\ bracket \rangle \langle formula \rangle \langle binary\ connective \rangle \langle formula \rangle \langle right\ bracket \rangle$

$\langle quantor\ \ formula \rangle ::= \langle left\ bracket \rangle \langle quantor \rangle \langle variable \rangle \langle formula \rangle \langle right\ bracket \rangle$

3.3 Интерпретации, модели и алгебраические системы

Модель логики предикатов задаётся множеством элементов (носителем) и множеством отношений на этом множестве элементов (сигнатурой). Любое множество предикатов задаёт хотя бы одну модель.
Каждая формула языка логики предикатов обозначает некоторый предикат. Интерпретацией формулы логики предикатов называется функция, которая каждой (предметной) переменной формулы ставит в соответствие её значение.
Как можно будет увидеть из последующих примеров число интерпретаций формулы в модели равно m^n, где n есть число свободных предметных переменных в формуле, а m – мощность носителя модели. Число моделей, которые можно построить равно $\prod_{i=1}^{p} 2^{m^{q_i}}$, где p – число предикатов (мощность сигнатуры модели), q_i – число аргументов (арность) предиката i, m – мощность носителя модели.
По аналогии с алгеброй высказываний можно рассматривать алгебраическую систему логики предикатов, носитель которой совпадает с носителем модели, а сигнатура которой является множеством предикатов.
В связи с тем, что число интерпретаций в логике предикатов быстро возрастает и может быть неограниченным, табличный метод решения логических задач неприемлем.

3.3.1 Кванторы

Кванторы (кванторные формулы) являются способом сокращённой записи конъюнктивных и дизъюнктивных формул. В этом смысле понятие квантора аналогично понятию суммы произвольного количества чисел и интеграла. Необходимость введения кванторов связана с тем, что область определения предикатов может быть бесконечной.
Например, если носитель модели состоит из элементов σi, то

$$\left(\left(\forall \chi P(\chi) \right) \Leftrightarrow \left(\left(\left(\left(P(\sigma 1) \wedge P(\sigma 1) \right) \wedge \ldots \right) \wedge P(\sigma i) \right) \wedge \ldots \right) \right)$$

$$\left(\left(\forall \chi P(\chi) \right) \Leftrightarrow \left(\left(\left(\left(P(\sigma 1) \vee P(\sigma 2) \right) \vee \ldots \right) \vee P(\sigma i) \right) \vee \ldots \right) \right)$$

аналогично тому, как:

$$\left(\sum_{i=1}^{n} i = (1+2+...+n)\right)$$
$$\left(\int_a^b f(\chi)d\chi = \int_a^c f(\chi)d\chi + \int_c^b f(\chi)d\chi\right).$$

Переменная (вхождение переменной) является в формуле свободной, если не существует кванторной формулы для этой переменной, которая является подформулой рассматриваемой формулы (включающей этой вхождение). Иначе переменная (вхождение переменной) называется связанной (соответствующим квантором).

Рассмотрим модель, носитель которой имеет два элемента σ и τ. Тогда следующая формула может в рамках $((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$ этой модели следующие интерпретации (в этом случае, предикат S может характеризовать рефлексивное бинарное отношение).

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	σ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

Для элементов σ и τ существуют различные модели, носитель которых содержит только эти два элемента, но различающиеся сигнатурой. Например, возможна модель, в которой могут быть построены следующие интерпретации.

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\perp$	$\perp$	$\perp$	$\bullet$	$\bullet$
τ	σ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

Другими примерами моделей для элементов σ и τ являются следующие.

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\perp$	$\bullet$	$\bullet$	$\perp$	$\bullet$
τ	σ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\perp$	$\bullet$	$\bullet$	$\bullet$	$\bullet$
τ	σ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\bullet$	$\perp$	$\bullet$	$\perp$	$\bullet$
τ	σ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\bullet$	$\perp$	$\bullet$	$\bullet$	$\bullet$
τ	σ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\bullet$	$\bullet$	$\bullet$	$\perp$	$\bullet$
τ	σ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\bullet$	$\bullet$	$\bullet$	$\bullet$	$\bullet$
τ	σ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	σ	$\perp$	$\perp$	$\perp$	$\bullet$	$\bullet$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\perp$	$\perp$	$\perp$	$\bullet$	$\bullet$
τ	σ	$\perp$	$\perp$	$\perp$	$\bullet$	$\bullet$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

χ	δ	$R(\chi,\delta,\sigma)$	$R(\chi,\delta,\tau)$	$(\exists\lambda R(\chi,\delta,\lambda))$	$S(\chi)$	$((\exists\lambda R(\chi,\delta,\lambda))\vee S(\chi))$
σ	σ	$\perp$	$\bullet$	$\bullet$	$\perp$	$\bullet$
τ	σ	$\perp$	$\perp$	$\perp$	$\bullet$	$\bullet$
σ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
τ	τ	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

и т.д.

3.3.2 Равносильные преобразования

В дополнение к равносильным преобразованиям логики высказываний в логике предикатов рассматриваются следующие равносильные преобразования над кванторами.

Коммутативность.
$$\left(\left(\forall\chi\left(\forall\lambda\alpha(\chi,\lambda)\right)\right)\Leftrightarrow\left(\forall\lambda\left(\forall\chi\alpha(\chi,\lambda)\right)\right)\right)$$
$$\left(\left(\exists\chi\left(\exists\lambda\alpha(\chi,\lambda)\right)\right)\Leftrightarrow\left(\exists\lambda\left(\exists\chi\alpha(\chi,\lambda)\right)\right)\right)$$
Дистрибутивность.
$$\left(\left(\forall\chi\left(\alpha(\chi)\wedge\beta(\chi)\right)\right)\Leftrightarrow\left(\left(\forall\chi\alpha(\chi)\right)\wedge\left(\forall\chi\beta(\chi)\right)\right)\right)$$
$$\left(\left(\exists\chi\left(\alpha(\chi)\vee\beta(\chi)\right)\right)\Leftrightarrow\left(\left(\exists\chi\alpha(\chi)\right)\vee\left(\exists\chi\beta(\chi)\right)\right)\right)$$
Вынос констант (γ).
$$\left(\left(\forall\chi\left(\alpha(\chi)\wedge\gamma\right)\right)\Leftrightarrow\left(\left(\forall\chi\alpha(\chi)\right)\wedge\gamma\right)\right)$$
$$\left(\left(\exists\chi\left(\alpha(\chi)\wedge\gamma\right)\right)\Leftrightarrow\left(\left(\exists\chi\alpha(\chi)\right)\wedge\gamma\right)\right)$$
$$\left(\left(\forall\chi\left(\alpha(\chi)\vee\gamma\right)\right)\Leftrightarrow\left(\left(\forall\chi\alpha(\chi)\right)\vee\gamma\right)\right)$$
$$\left(\left(\exists\chi\left(\alpha(\chi)\vee\gamma\right)\right)\Leftrightarrow\left(\left(\exists\chi\alpha(\chi)\right)\vee\gamma\right)\right)$$
Двойственность кванторов.
$$\left(\left(\neg\left(\forall\chi\alpha(\chi)\right)\right)\Leftrightarrow\left(\exists\chi\left(\neg\alpha(\chi)\right)\right)\right)$$
$$\left(\left(\neg\left(\exists\chi\alpha(\chi)\right)\right)\Leftrightarrow\left(\forall\chi\left(\neg\alpha(\chi)\right)\right)\right)$$

3.3.3 Сколемовские, предварённые и нормальные формы

Предварённой нормальной конъюктивной (дизъюнктивной) формой, называется формула, в которой нет некванторной подформулы, которая имеет кванторнуюподформулу, и в которой присутствует в качестве подформулы КНФ (ДНФ), которая не является подформулой формулы, не являющейся кванторной формулой или КНФ (ДНФ). Другими словами, предварённая нормальная форма – это формула, у которой все кванторы записаны слева, а оставшаяся подформула является КНФ (ДНФ), не включающей кванторов. Сколемовская (стандартная) форма получается из предварённой нормальной, путём преобразования последней и записью в виде текста расширения рассмотренного языка предикатов, допускающего константы и функциональные термы в качестве аргументов предикатов. Строится сколемовская стандартная форма следующим образом.
Если предварённая нормальная форма является кванторной формулой существования, то каждая во всех вхождениях переменная, связанная квантором заменяется, константой, которая отсутствует в формуле, и процесс повторяется для оставшейся подформулы.
Далее, если квантор существования расположен после кванторов общности, то соответствующая переменная, связанная квантором существования заменяется отсутствующим в исходной формуле функциональным термом, список аргументов в котором содержит все переменные, связанные расположенными до рассматриваемого квантора существования кванторами общности. Все кванторы общности исключатся из исходной формулы. Рассмотрим следующий пример.
Пусть есть формула $\left(\exists\chi\left(\forall\lambda\left(\forall\delta\left(\exists\gamma\left(\forall\sigma P(\chi,\lambda,\delta,\gamma,\sigma)\right)\right)\right)\right)\right)$, тогда её сколемовская стандартная форма выглядит следующим образом.

$$P\left(a,\lambda,\delta,\varphi\left(\langle\lambda,\delta\rangle\right),\sigma\right)$$

Использование сколемовских стандартных форм (ССФ) позволяет упростить применение метода резолюций к логике предикатов.

Эрбранова модель для формулы логики предикатов строится следующим образом. В носитель эрбрановой модели включаются все константы ССФ, если констант нету – туда включается фиктивная константа, далее для каждой функции включается в носитель её значение от имеющихся в модели констант и значений функций, процесс повторяется рекурсивно для всех значений. Таким образом, если присутствует хотя бы одна функция, то носитель эрбрановой модели является бесконечным. Если функции отсутствуют, то мощность носителя эрбрановой модели равна максимальному значению между числом констант ССФ и единицей. Для того чтобы доказать выполнимость формулы, необходимо и достаточно рассмотреть все возможные интерпретации всех возможных моделей, которые допускает носитель эрбрановой области. В случае, когда модель бесконечна, перебор всех интерпретаций невозможен и требуется использовать другие методы.

3.4 Исчисление предикатов

3.4.1 Аксиоматика

Рассмотрим аксиоматику исчисления предикатов, которая основывается на аксиоматике исчисления высказываний, используя все аксиомные схемы исчисления высказываний применительно к формулам языка логики предикатов, а также используя две следующие дополнительные аксиомные схемы для кванторов.

Здесь формула α, имеющая свободное вхождение переменной χ, обозначена как $\alpha(\chi)$. Формула $\alpha(\chi)$ не имеет свободных вхождений τ. Формула $\alpha(\tau)$ получена заменой всех свободных вхождений переменной χ на τ.

	введение	удаление
$\forall$		$\left(\left(\forall\chi\alpha(\chi)\right)\to\alpha(\tau)\right)$
$\exists$	$\left(\alpha(\chi)\to\left(\exists\chi\alpha(\chi)\right)\right)$	

3.4.2 Правила вывода

Кроме правила прямого заключения в логике предикатов используются ещё два правила вывода.

Для любого множества формул G языка логики предикатов, для любых формул $\alpha(\chi)$ и $\beta(\chi)$, содержащих свободную переменную χ, и любой формулы γ, которая не содержит свободной переменной χ, справедливы следующие правила (правило обобщения и конкретизации соответственно).

$$\left(\left(G\cup\left\{\left(\gamma\to\beta(\chi)\right)\right\}\right)\text{ђ}\left(G\cup\left\{\left(\gamma\to\left(\forall\chi\beta(\chi)\right)\right)\right\}\right)\right)$$

$$\left(\left(G\cup\left\{\left(\alpha(\chi)\to\gamma\right)\right\}\right)\text{ђ}\left(G\cup\left\{\left(\left(\exists\chi\alpha(\chi)\right)\to\gamma\right)\right\}\right)\right)$$

Наряду с этими правилами и исходя из них, как метатеоремы могут быть получены следующие правила вывода.
$\left(\left(G \cup \{\beta(\chi)\}\right) \mathfrak{h} \left(G \cup \{(\forall \chi \beta(\chi))\}\right)\right)$

	введение	удаление
$\forall$	$\left(\left(G \cup \{(\gamma \to \beta(\chi))\}\right) \mathfrak{h} \left(G \cup \{(\gamma \to (\forall \chi \beta(\chi)))\}\right)\right)$	$\left(\left(G \cup \{(\forall \chi \alpha(\chi))\}\right) \mathfrak{h} \left(G \cup \{\alpha(\tau)\}\right)\right)$
$\exists$	$\left(\left(G \cup \{\alpha(\chi)\}\right) \mathfrak{h} \left(G \cup \{(\exists \chi \alpha(\chi))\}\right)\right)$	$\left(\left(G \cup \{(\alpha(\chi) \to \gamma)\}\right) \mathfrak{h} \left(G \cup \{((\exists \chi \alpha(\chi)) \to \gamma)\}\right)\right)$

3.3 Исчисление секвенций

Для упрощения построения и анализа доказательств иногда рассматривают такую аксиоматическую теорию, как исчисление секвенций, которая аналогична исчислению предикатов, но использует другой, в некоторых отношениях более удобный, язык.

3.3.1 Язык исчисления секвенций

Язык исчисления секвенций строится из формул вида.

$$(\Gamma \Rightarrow \Delta)$$

Такая формула соответствует следующей формуле языка логике предикатов.

$$\left(\left(\left(\left(\bullet \ \wedge \alpha 1\right) \wedge \ldots\right) \wedge \alpha n\right) \to \left(\left(\left(\beta 1 \vee \ldots\right) \vee \beta m\right) \vee \bot\right)\right)$$

Соответственно формула языка исчисления секвенций

$$(\alpha \Gamma \Rightarrow \Delta \beta)$$

выражает следующую формулу языка логики предикатов.

$$\left(\left(\left(\left(\bullet \ \wedge \alpha\right) \wedge \ldots\right) \wedge \alpha n\right) \to \left(\left(\left(\beta 1 \vee \ldots\right) \vee \beta\right) \vee \bot\right)\right)$$

3.3.2 Аксиоматика

В качестве единственной аксиомной схемы исчисления секвенций рассматривают следующую аксиомную схему.

$$(\Delta \Rightarrow \Delta)$$

3.3.3 Правила вывода

Правила вывода исчисления секвенций делятся на два типа: логические правила вывода и структурные.
Логические правила вывода приведены ниже.

	введение	удаление
$\to$	$\left(\{(\alpha \Gamma \Rightarrow \Delta \beta)\} \mathfrak{h} \{(\Gamma \Rightarrow \Delta(\alpha \to \beta))\}\right)$	$\left(\{(\Gamma \Rightarrow \Delta \alpha), (\beta \Gamma \Rightarrow \Delta)\} \mathfrak{h} \{((\alpha \to \beta)\Gamma \Rightarrow \Delta)\}\right)$
$\wedge$	$\left(\{(\Gamma \Rightarrow \Delta \alpha), (\Gamma \Rightarrow \Delta \beta)\} \mathfrak{h} \{(\Gamma \Rightarrow \Delta(\alpha \wedge \beta))\}\right)$	$\left(\{(\alpha \beta \Gamma \Rightarrow \Delta)\} \mathfrak{h} \{((\alpha \wedge \beta)\Gamma \Rightarrow \Delta)\}\right)$
$\vee$	$\left(\{(\Gamma \Rightarrow \Delta \alpha \beta)\} \mathfrak{h} \{(\Gamma \Rightarrow \Delta(\alpha \vee \beta))\}\right)$	$\left(\{(\alpha \Gamma \Rightarrow \Delta), (\beta \Gamma \Rightarrow \Delta)\} \mathfrak{h} \{((\alpha \vee \beta)\Gamma \Rightarrow \Delta)\}\right)$
$\neg$	$\left(\{(\alpha \Gamma \Rightarrow \Delta)\} \mathfrak{h} \{(\Gamma \Rightarrow \Delta(\neg \alpha))\}\right)$	$\left(\{(\Gamma \Rightarrow \Delta \alpha)\} \mathfrak{h} \{((\neg \alpha)\Gamma \Rightarrow \Delta)\}\right)$
$\sim$	$\left(\{(\alpha \Gamma \Rightarrow \Delta \beta), (\beta \Gamma \Rightarrow \Delta \alpha)\} \mathfrak{h} \{(\Gamma \Rightarrow \Delta(\alpha \sim \beta))\}\right)$	$\left(\{((\alpha \to \beta)\Gamma \Rightarrow \Delta), ((\beta \to \alpha)\Gamma \Rightarrow \Delta)\} \mathfrak{h} \{((\alpha \sim \beta)\Gamma \Rightarrow \Delta)\}\right)$
$\forall$	$\left(\{(\Gamma \Rightarrow \alpha(\lambda))\} \mathfrak{h} \{(\Gamma \Rightarrow (\forall \chi \alpha(\chi)))\}\right)$	$\left(\{(\alpha(\lambda)\Gamma \Rightarrow \Delta)\} \mathfrak{h} \{((\forall \chi \alpha(\chi))\Gamma \Rightarrow \Delta)\}\right)$

$\exists$	$\left(\left\{\left(\Gamma \Rightarrow \alpha(\lambda)\right)\right\} \text{ђ} \left\{\left(\Gamma \Rightarrow \left(\exists \chi \alpha(\chi)\right)\right)\right\}\right)$	$\left(\left\{\left(\alpha(\lambda)\Gamma \Rightarrow \Delta\right)\right\} \text{ђ} \left\{\left(\left(\exists \chi \alpha(\chi)\right)\Gamma \Rightarrow \Delta\right)\right\}\right)$

Структурные правила вывода исчисления секвенций следующие.

$$\left(\left\{(\Gamma \Rightarrow \Delta)\right\} \text{ђ} \left\{(\Gamma\Pi \Rightarrow \Delta\Phi)\right\}\right)$$

$$\left(\left\{(\alpha\alpha\Gamma \Rightarrow \Delta)\right\} \text{ђ} \left\{(\alpha\Gamma \Rightarrow \Delta)\right\}\right)$$

$$\left(\left\{(\Pi\alpha\beta\Gamma \Rightarrow \Delta)\right\} \text{ђ} \left\{(\Pi\beta\alpha\Gamma \Rightarrow \Delta)\right\}\right)$$

$$\left(\left\{(\Gamma \Rightarrow \Delta\alpha),(\alpha\Pi \Rightarrow \Phi)\right\} \text{ђ} \left\{(\Gamma\Pi \Rightarrow \Delta\Phi)\right\}\right)$$

Доказательства (формальный вывод) в исчислении секвенций записываются в древовидном виде. В остальном, принцип построения доказательств аналогичен принципам построения выводов в логике предикатов.

4. Прикладные исчисления

Вышеперечисленные исчисления относят к «чистым» исчислениям. Прикладные исчисления, в отличие от «чистых» исчислений, содержат дополнительные предикатные константы в языке и дополнительные аксиомы, определяющие свойства этих констант. Прикладные исчисления можно разбить на абстрактные и предметные, т.е. те, которые описывают свойства «чистых» математических абстракций и те, которые описывают свойства физических объектов. Предметные прикладные исчисления обычно строятся с помощью языков логического программирования в рамках некоторой программной системы, решающей ту или иную прикладную задачу. Абстрактные прикладные исчисления встречаются чаще. Далее рассмотрим некоторые абстрактные прикладные исчисления.

4.1 Исчисление с равенством

Исчисление с равенством – исчисление, которое описывает, формализует свойства бинарного отношения эквивалентности (равенства).

4.1.1 Язык

Алфавит этого языка содержит дополнительный класс символов, включающий символ ÿ, использующийся в качестве предикатной константы. Синтаксис позволяет строить предикатные термы, используя этот символ.

4.1.2 Аксиоматика

Аксиоматика исчисления с равенством может быть выстроена на основе аксиоматики логики предикатов с добавлением следующей аксиомной схемы и аксиомы. Здесь формула $\alpha(\chi,\lambda)$ получена из формулы $\alpha(\chi,\chi)$ заменой некоторых свободных вхождений переменной χ на λ.

$\left(\left(\chi \,\text{ÿ}\, \lambda\right) \to \left(\alpha(\chi,\chi) \to \alpha(\chi,\lambda)\right)\right)$

$\left(\forall x\left(x \,\text{ÿ}\, x\right)\right)$

4.1.3 Теоремы

Теоремами исчисления с равенством являются свойства симметричности и транзитивности соответственно.

$$\left(\forall x\left(\forall y\left(\left(x \text{ ў } y\right) \rightarrow \left(y \text{ ў } x\right)\right)\right)\right)$$

$$\left(\forall x\left(\forall y\left(\forall z\left(\left(\left(x \text{ ў } y\right) \wedge \left(y \text{ ў } z\right)\right) \rightarrow \left(x \text{ ў } z\right)\right)\right)\right)\right)$$

4.2 Исчисление порядка

Исчисление с равенством – исчисление, которое описывает, формализует свойства бинарных отношений порядка. В дополнение к аксиома логики исчисления с равенством, следующие две аксиомы используются во всех исчислениях порядка, описывающих отношение (предпорядка) порядка $\textbf{Ъ}$, специальный символ для обозначения которого вводится, как предикатная константа, в алфавит языка исчисления порядка. Этими аксиомами соответственно являются аксиома антисимметричности и аксиома транзитивности.

$$\left(\forall x\left(\forall y\left(\left(\left(x \textbf{ Ъ } y\right) \wedge \left(y \textbf{ Ъ } x\right)\right) \rightarrow \left(y \text{ ў } x\right)\right)\right)\right)$$

$$\left(\forall x\left(\forall y\left(\forall z\left(\left(\left(x \textbf{ Ъ } y\right) \wedge \left(y \textbf{ Ъ } z\right)\right) \rightarrow \left(x \textbf{ Ъ } z\right)\right)\right)\right)\right)$$

В качестве правил вывода используются все правила исчисления предикатов.

4.3 Исчисление нестрогого порядка

Включение в аксиоматику исчисления порядка (для отношения $\textbf{Ъ}$) следующей аксиомы приводит к исчислению нестрогого порядка.

$$\left(\forall x\left(x \textbf{ Ъ } x\right)\right)$$

В качестве правил вывода используются все правила исчисления предикатов.

4.4 Исчисление строгого порядка

Включение в аксиоматику исчисления порядка (для отношения $\textbf{Ъ}$) следующей аксиомы приводит к исчислению строгого порядка.

$$\left(\neg\left(\forall x\left(x \textbf{ Ъ } x\right)\right)\right)$$

В качестве правил вывода используются все правила исчисления предикатов.

4.5 Исчисление частичного порядка

Включение в аксиоматику исчисления порядка (для отношения $\textbf{Ъ}$) следующей аксиомы приводит к исчислению частичного порядка.

$$\left(\neg\left(\forall x\left(\forall y\left(\left(x \textbf{ Ъ } y\right) \vee \left(y \textbf{ Ъ } x\right)\right)\right)\right)\right)$$

В качестве правил вывода используются все правила исчисления предикатов.

4.6 Исчисление линейного порядка

Включение в аксиоматику исчисления порядка (для отношения $\textbf{Ъ}$) следующей аксиомы приводит к исчислению полного порядка.

$$\left(\forall x\left(\forall y\left(\left(x \textbf{ Ъ } y\right) \vee \left(y \textbf{ Ъ } x\right)\right)\right)\right)$$

В качестве правил вывода используются все правила исчисления предикатов.

4.7 Исчисление арифметики

Исчисление арифметики описывает натуральные числа и закономерности операций над натуральными числами. В алфавит языка исчисления арифметики дополнительно вводятся классы следующих символов $0, ', +, *$. Аксиоматика кроме аксиомных схем исчисления предикатов содержит шесть аксиом и одну аксиомную схему, формализующую метод математической индукции.

$$\left(\forall x\left(\neg\left(x'=0\right)\right)\right)$$

$$\left(\forall x\left(\forall y\left(\left(x'=y'\right)\to\left(x=y\right)\right)\right)\right)$$

$$\left(\forall x\left(\left(x+0\right)=x\right)\right)$$

$$\left(\forall x\left(\forall y\left(\left(x+y'\right)=\left(x+y\right)'\right)\right)\right)$$

$$\left(\forall x\left(\left(x*0\right)=0\right)\right)$$

$$\left(\forall x\left(\forall y\left(\left(x*y'\right)=\left(\left(x*y\right)+x\right)\right)\right)\right)$$

$$\left(\left(\alpha\left(0\right)\wedge\left(\forall x\left(\alpha\left(x\right)\to\alpha\left(x'\right)\right)\right)\right)\to\left(\forall x\alpha\left(x\right)\right)\right)$$

4.7.1 Теоремы

Примерами теорем исчисления арифметики являются следующие формулы.

$$\left(\forall x\left(\forall y\left(\left(x+y\right)=\left(y+x\right)\right)\right)\right)$$

$$\left(\forall x\left(\forall y\left(\left(x*y\right)=\left(y*x\right)\right)\right)\right)$$

4.8 Временные логики

К временным логикам относятся исчисления, которые формализуют свойства временных (темпоральных) отношений и зависимость истинности тех или иных предикатов, от временных переменных.

4.8.1 Интервальная временная логика

Интервальная временная логика описывает свойства и отношения временных интервалов, и предикатов заданных для этих интервалов. Множество всех возможных временных интервалов интервальной временной логики обозначим $\mathbf{I}$. Введём следующие сокращения.

$$\left(\left(\forall i.\alpha\left(i\right)\right)\sim\left(\forall i\left(\left(i\in\mathbf{I}\right)\to\alpha\left(i\right)\right)\right)\right)$$

$$\left(\left(\exists i.\alpha\left(i\right)\right)\sim\left(\exists i\left(\left(i\in\mathbf{I}\right)\wedge\alpha\left(i\right)\right)\right)\right)$$

Рассмотрим свойство отношение непосредственного следования интервалов $:$.Запись $(i:j)$ читается как: «интервал i непосредственно предшествует интервалу j».

Другие отношения над интервалами могут быть определены следующим образом.

Отношение «строго до» (отношение опосредованного предшествования).

$$\left(\left(i\prec j\right)\sim\left(\exists m.\left(\left(i:m\right)\wedge\left(m:j\right)\right)\right)\right)$$

Отношение «нестрого до» (отношение предшествования).

$$\left(\left(i \prec: j\right) \sim \left(\left(i \prec j\right) \vee \left(i : j\right)\right)\right)$$

Отношение включения интервалов.

$$\left(\left(i \text{ ф } j\right) \sim \left(\exists k.\left(\exists l.\left(\left(\left(k : j\right) \wedge \left(j : l\right)\right) \wedge \left(\left(k \prec: i\right) \wedge \left(i \prec: l\right)\right)\right)\right)\right)\right)$$

Отношение строгого включения интервалов.

$$\left(\left(i \text{ p } j\right) \sim \left(\left(i \text{ ф } j\right) \wedge \left(\neg\left(i = j\right)\right)\right)\right)$$

Отношение разъединённости.

$$\left(\left(i >< j\right) \sim \left(\left(i \prec: j\right) \vee \left(j \prec: i\right)\right)\right)$$

Аксиоматика интервальной временной логики основывается на аксиоматике исчисления с равенством. Следующие пять аксиом описывают свойства временных интервалов и отношения непосредственного предшествования.

$$\left(\forall i.\left(\exists j.\left(\exists k.\left(\left(j : i\right) \wedge \left(i : k\right)\right)\right)\right)\right)$$

$$\left(\forall i.\left(\forall j.\left(\forall k.\left(\forall l.\left(\left(\left(i : j\right) \wedge \left(j : k\right)\right) \wedge \left(k : l\right)\right) \rightarrow \left(\exists m.\left(\left(i : m\right) \wedge \left(m : l\right)\right)\right)\right)\right)\right)\right)$$

$$\left(\forall i.\left(\forall j.\left(\forall k.\left(\forall l.\left(\left(\left(\left(i : j\right) \wedge \left(i : k\right)\right) \wedge \left(l : j\right)\right) \rightarrow \left(l : k\right)\right)\right)\right)\right)\right)$$

$$\left(\forall i.\left(\forall j.\left(\forall k.\left(\forall l.\left(\left(\left(\left(k : i\right) \wedge \left(k : j\right)\right) \wedge \left(\left(i : l\right) \wedge \left(j : l\right)\right)\right) \rightarrow \left(i = j\right)\right)\right)\right)\right)\right)$$

$$\left(\forall i.\left(\forall j.\left(\forall k.\left(\forall l.\left(\left(\left(i : j\right) \wedge \left(k : l\right)\right) \rightarrow \left(\left(\neg\left(i : l\right)\right) \sim \left(\exists m.\left(\left(\left(k : m\right) \wedge \left(m : j\right)\right) \sim \left(\left(i : m\right) \wedge \left(m : l\right)\right)\right)\right)\right)\right)\right)\right)\right)\right)$$

Следующие формулы определяют свойства слабого и сильного отрицаний в интервальной временной логике.

$$\left(\forall t.\left(\left(\sim \left(\sim \alpha(t)\right)\right) \sim \alpha(t)\right)\right)$$

$$\left(\forall t.\left(\left(\neg\alpha(t)\right) \sim \left(\forall t1.\left(\left(t1 \text{ ф } t\right) \rightarrow \left(\sim \alpha(t1)\right)\right)\right)\right)\right)$$

Интервальная логика рассматривает так называемые однородные предикаты. Любой однородный предикат по интервально переменной $t1$, обозначенный формулой α, удовлетворяет следующей аксиомной схеме однородности.

$$\left(\forall x1 \ldots \left(\forall xn \left(\forall t.\left(\forall t1.\left(\left(\alpha\left(x1,\ldots xn,t\right) \wedge \left(t1 \text{ ф } t\right)\right) \rightarrow \alpha\left(x1,\ldots xn,t1\right)\right)\right)\right)\right)\right)$$

Также для любого однородного предиката выполняется свойство дискретной вариативности (изменчивости), задаваемое следующей аксиомной схемой.

$$\left(\forall t.\left(\left(\sim \alpha(t)\right) \rightarrow \left(\exists t1.\left(\left(t1 \text{ ф } t\right) \wedge \left(\neg\alpha(t)\right)\right)\right)\right)\right)$$

Следует отметить, что сильное отрицание однородного предиката, в общем случае не является однородным предикатом по той же интервальной (временной) переменной, это означает, что понятие сильного отрицания является относительным и темпорально независимым от предиката, который оно отрицает.

Теоремы интервальной временной логики являются следующие формулы.

$$\neg\left(\forall t.\left(\left(\neg\left(\neg\alpha(t)\right)\right) \sim \alpha(t)\right)\right)$$

$$\left(\forall t.\left(\left(\sim \left(\sim \left(\sim \alpha(t)\right)\right)\right) \sim \left(\sim \alpha(t)\right)\right)\right)$$

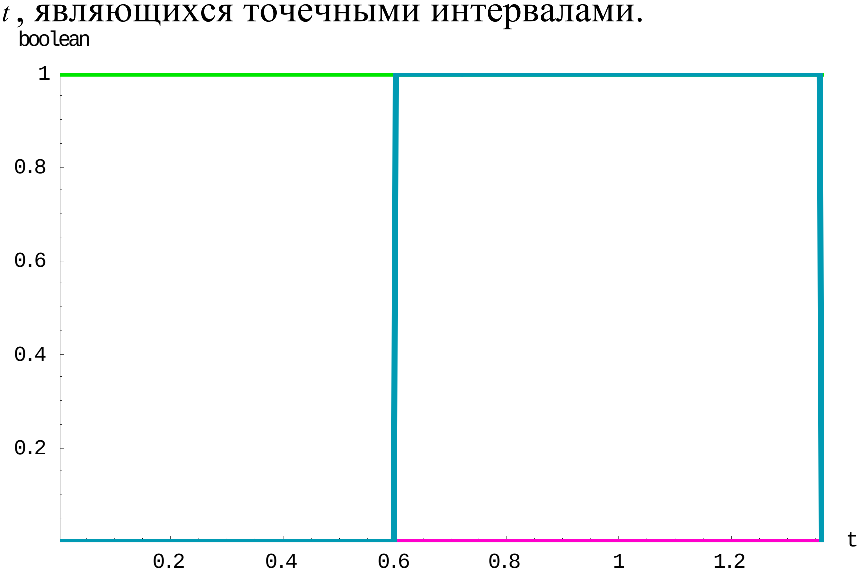
Для однородных по переменным t и $t1$ предикатов также верны следующие теоремы.

$$\left(\forall t.\left(\forall t1.\left(\left(\alpha(t) \wedge \left(\neg\alpha(t1)\right)\right) \rightarrow \left(t >< t1\right)\right)\right)\right)$$

$\left(\forall t.\left(\left(\neg\left(\neg\left(\neg\alpha(t)\right)\right)\right)\sim\left(\neg\alpha(t)\right)\right)\right)$

$\left(\forall t.\left(\left(\neg\left(\sim\alpha(t)\right)\right)\sim\left(\neg\left(\neg\alpha(t)\right)\right)\right)\right)$

Ниже приведена зависимость некоторого предиката от значений переменной t, являющихся точечными интервалами.



4.8.2 Логика ветвящегося времени

Рассматривается (временная) модель T, которая состоит из множества временных моментов и отношения предшествования $\prec$. Отношение предшествования обладает в этой логике следующими свойствами.

Свойство отношения строгого порядка.

$\left(\left(i \preceq j\right)\sim\left(\left(i \prec j\right)\vee\left(i=j\right)\right)\right)$

Свосйство линейноти в прошлое.

$\left(\forall x\left(\forall y\left(\forall z\left(\left(\left(y \prec x\right)\wedge\left(z \prec x\right)\right)\rightarrow\left(\left(y \preceq z\right)\vee\left(z \preceq y\right)\right)\right)\right)\right)\right)$

Свойство связности.

$\left(\forall x\left(\forall y\left(\left(\left(x\in\mathsf{T}\right)\wedge\left(y\in\mathsf{T}\right)\right)\rightarrow\left(\exists z\left(\left(z\in\mathsf{T}\right)\wedge\left(\left(z \preceq x\right)\wedge\left(z \preceq y\right)\right)\right)\right)\right)\right)\right)$

Семантика временных кванторов в логике ветвящегося времени определяется следующим образом. Вводится понятие формулы, общезначимой в модели, задаваемой тремя величинами T,σ,χ, которые соответственно являются: временной моделью T, элементом множества всех ветвей $Branches(\mathsf{T})$ временной модели T и микромоделью (состоянием) – элементом носителя модели T. Фактически эти три величины соответствуют неявным переменными в формуле α, обозначающей временной (зависящий от значений этих переменных) предикат.

$\left(\left(\mathsf{T},\sigma,\chi\models\alpha\right)\sim\left(\left(\models\alpha\left(\mathsf{T},\sigma,\chi\right)\right)\wedge\left(\left(\chi\in\sigma\right)\wedge\left(\sigma\in Branches\left(\mathsf{T}\right)\right)\right)\right)\right)$

Задана некоторая функция φ, которая позволяет вычислить общезначимость любой формулы α, принадлежащей множеству атомарных формул A, в микромодели χ.

$\left(\alpha\in\mathsf{A}\right)\rightarrow\left(\left(\mathsf{T},\sigma,\chi\models\alpha\right)\sim\left(\chi\in\varphi\left(\alpha\right)\right)\right)$

Атомарные формулы предполагаются не являющимися явно открытыми, т.е. не содержащими явных вхождений свободных переменных. Тогда общезначимость отрицания формулы и конъюнкции формул устанавливается соответственно следующим образом.

$$\big((\mathsf{T},\sigma,\chi \text{ ‘ } (\neg\alpha)) \sim (\neg(\mathsf{T},\sigma,\chi \text{ ‘ } \alpha))\big)$$

$$\big((\mathsf{T},\sigma,\chi \text{ ‘ } (\alpha\wedge\beta)) \sim ((\mathsf{T},\sigma,\chi \text{ ‘ } \alpha)\wedge(\mathsf{T},\sigma,\chi \text{ ‘ } \beta))\big)$$

Общезначимость формул, содержащих временные (темпоральные) кванторы, устанавливается для кванторов постоянства в будущем G, постоянства в прошлом H и квантора необходимости $,$ соответственно.

$$\big((\mathsf{T},\sigma,\chi \text{ ‘ } (G\alpha)) \sim \forall\lambda((\chi \prec\!\!: \lambda)\to(\mathsf{T},\sigma,\lambda \text{ ‘ } \alpha))\big)$$

$$\big((\mathsf{T},\sigma,\chi \text{ ‘ } (H\alpha)) \sim \forall\lambda((\lambda \prec\!\!: \chi)\to(\mathsf{T},\sigma,\lambda \text{ ‘ } \alpha))\big)$$

$$\Big((\mathsf{T},\sigma,\chi \text{ ‘ } (,\ \alpha)) \sim \forall\pi\big(((\chi\in\pi)\wedge(\pi\in Branches(\mathsf{T})))\to(\mathsf{T},\pi,\chi \text{ ‘ } \alpha)\big)\Big)$$

Двойственные кванторы случайности в будущем, случайности в прошлом и возможности определяются через ранее определённые.

$$\big((F\alpha)\sim(\neg(G(\neg\alpha)))\big)$$

$$\big((P\alpha)\sim(\neg(H(\neg\alpha)))\big)$$

$$\big((\mathrm{a}\ \alpha)\sim(\neg(,\ (\neg\alpha)))\big)$$

Аксиоматика логики с ветвящимся временем может быть построена на аксиоматике, как минимум, исчисления высказываний и дополнительно может включать какие-либо из следующих.

$$\big((G(\alpha\to\beta))\to((G\alpha)\to(G\beta))\big)$$

$$\big((G\alpha)\to(G(G\alpha))\big)$$

$$\big(\alpha\to(G(P\alpha))\big)$$

$$\big((F\alpha)\to(G(((F\alpha)\vee\alpha)\vee(P\alpha)))\big)$$

$$\big((H(\alpha\to\beta))\to((H\alpha)\to(H\beta))\big)$$

$$\big((H\alpha)\to(H(H\alpha))\big)$$

$$\big(\alpha\to(H(F\alpha))\big)$$

$$\big((P\alpha)\to(H(((F\alpha)\vee\alpha)\vee(P\alpha)))\big)$$

$$\big((,\ (\alpha\to\beta))\to((,\ \alpha)\to(,\ \beta))\big)$$

$$\big((,\ \alpha)\to(,\ (,\ \alpha))\big)$$

$$\big((,\ \alpha)\to\alpha\big)$$

$$\big(\alpha\to(,\ (\mathrm{a}\ \alpha))\big)$$

Для любой атомарной формулы γ верно $(\gamma\to(,\ \gamma))$.

$$\big((,\ (H\alpha))\sim(H(,\ \alpha))\big)$$

$$\big((P(,\ \alpha))\to(,\ (P\alpha))\big)$$

$$\big((,\ (G\alpha))\to(G(,\ \alpha))\big)$$

$$\big((G\perp)\to(,\ (G\perp))\big)$$

Множество правил вывода логики ветвящегося времени кроме правил вывода исчисления высказываний включает следующие.

$$\left(\left(G \cup \{\alpha\}\right) \mathfrak{h} \left(G \cup \{G\alpha\}\right)\right)$$

$$\left(\left(G \cup \{\alpha\}\right) \mathfrak{h} \left(G \cup \{H\alpha\}\right)\right)$$

$$\left(\left(G \cup \{\alpha\}\right) \mathfrak{h} \left(G \cup \{,\ \alpha\}\right)\right)$$

Для любой атомарной формулы γ и формулы α, подформулой которой не является γ верно $\left(\left(G \cup \left\{\left(\left(\gamma \wedge \left(H\left(\neg\gamma\right)\right)\right) \rightarrow \alpha\right)\right\}\right) \mathfrak{h} \left(G \cup \{\alpha\}\right)\right)$.

Одним из обобщающих аналогов логик ветвящегося времени, для которого также могут использоваться правила немонотонного вывода, являются логики, которые получили название семантик возможных миров.

Формула является общезначимой в модели, тогда и только тогда когда она общезначима в любой микромодели.

Формула является общезначимой в структуре (временной модели), тогда и только тогда когда она общезначима во всех микромоделях этой структуры (временной модели).

4.9 Модальные логики

Кванторы, которые выражают в том или ином виде временную (темпоральную) семантику, а именно: необходимость, возможность, будущность (будущее), прошлое, известность, допустимость, вера – получили называние модальных.

Модальные кванторы могут обладать следующими свойствами. Модальные кванторы всегда двойственны, т.е. модальный квантор имеет всегда двойственный ему.

$$\left(\left(''\ \alpha\right) \sim \left(\neg\left(!\ \left(\neg\alpha\right)\right)\right)\right)$$

Каждая аксиомная схема для того или иного квантора при добавлении к аксиоматике исчисления предикатов порождает ту или иную модальную систему (исчисление).

Так называемая модальная K система для модального квантора $''$ порождается следующей аксиомной схемой (схемой дистрибутивности).

$$\left(\left(!\ \left(\alpha \rightarrow \beta\right)\right) \rightarrow \left(\left(!\ \alpha\right) \rightarrow \left(!\ \beta\right)\right)\right)$$

Модальная система T для модального квантора $''$ порождается «аксиомной схемой знания».

$$\left(\left(!\ \alpha\right) \rightarrow \alpha\right)$$

Модальная система 4 для модального квантора $''$ порождается «аксиомной схемой позитивной интроспекции».

$$\left(\left(!\ \alpha\right) \rightarrow \left(!\ \left(!\ \alpha\right)\right)\right)$$

Модальная система 5 для модального квантора $''$ порождается «аксиомной схемой позитивной интроспекции».

$$\left(\left(''\ \alpha\right) \rightarrow \left(!\ \left(''\ \alpha\right)\right)\right)$$

Можно рассматривать модальные системы, являющиеся комбинациями вышеперечисленных. Так, например, модальные системы $KT4$ и $KT45$ имеют обозначения **S4** и S5 соответственно.
Во всех этих системах наряду с правилами вывода исчисления предикатов используется следующее модальное правило вывода.
$((G \cup \{\alpha\})$ ђ $(G \cup \{!\ \alpha\}))$
Семантика квантора знания обычно определяется следующим образом.
$((\text{Т},\gamma,\chi\ ‘\ (!\ \alpha)) \sim \forall\lambda((\chi:\lambda)\to(\text{Т},\delta,\lambda\ ‘\ \alpha)))$

5. Неклассические логики и другие приложения

К неклассическим логикам относят логики, которые имеют качественно меньший набор аксиом, чем «классическое» исчисление предикатов, качественно иные правила вывода, либо оперируют с большим числом значений истинности, чем два (истина и ложь).

5.1 Многозначные логики

Многозначные логики используют три и более значений истинности. Для каждой многозначной логики можно определить разные наборы логических операций, поэтому могут существовать разные логики равной значности

5.1.1 Трёхзначная

Рассмотрим логические операции трёхзначной логики Лукасевича (со связкой импликации Лукасевича). Операции, выражаемые логическими связками и модальными кванторами, трёхзначной логики Лукасевича заданы следующими истинностными таблицами. Значения этой логики семантически интерпретируются как: «необходимо ложно», «проблематично» и «необходимо истинно» (0,1,2), что во временной семантике означает, что формула ложна во все моменты времени, формула истинна для определённого множества моментов времени (например, чётные моменты времени) и ложна в остальные моменты времени (например, нечётные моменты времени), и – формула истинна во все моменты времени.
Таблицы истинности для отрицания и модальных кванторов.

α	$(\neg\alpha)$	(a α)	(, α)
0	2	0	0
1	1	0	2
2	0	2	2

Таблица истинности конъюнкции.

$(\alpha\wedge\beta)$	0	1	2
0	0	0	0
1	0	1	1
2	0	1	2

Таблица истинности дизъюнкции.

$(\alpha\vee\beta)$	0	1	2
0	0	1	2

1	1	1	2
2	2	2	2

Таблица истинности импликации.

$(\alpha \to \beta)$	0	1	2
0	0	1	2
1	1	1	2
2	2	2	2

Таблица истинности эквиваленции.

$(\alpha \sim \beta)$	0	1	2
0	2	1	0
1	1	2	1
2	0	1	2

5.1.2 Четырёхзначная

По аналогии рассмотрим пример четырёхзначной логики. Значения рассматриваемой четырёхзначной логики называются «необходимо ложно», «случайно ложно, «случайно истинно», «необходимо истинно» (0,1,2,3), что семантически интерпретируются, что формула ложна во все моменты времени, формула ложна сейчас и истинна потом, что формула истинна сейчас и ложна потом, и – формула истинна во все моменты времени. Отметим, что второй либо третий тип формул этой четырёхзначной логики в трёхзначной логике Лукасевича мог бы рассматриваться как проблематичный.

Таблицы истинности для отрицания и модальных кванторов.

α	$(\neg\alpha)$	$(\text{a}\ \alpha)$	$(,\ \alpha)$
0	3	0	0
1	2	0	3
2	1	0	3
3	0	3	3

Таблица истинности конъюнкции.

$(\alpha \wedge \beta)$	0	1	2	2
0	0	0	0	0
1	0	1	0	1
2	0	0	2	2
3	0	1	2	3

Таблица истинности дизъюнкции.

$(\alpha \vee \beta)$	0	1	2	2
0	0	1	2	3
1	1	1	3	3
2	2	3	2	3
3	3	3	3	3

Таблица истинности импликации.

$(\alpha \to \beta)$	0	1	2	2

0	3	3	3	3
1	2	3	2	3
2	1	1	3	3
3	0	1	2	3

Таблица истинности эквиваленции.

$(\alpha \sim \beta)$	0	1	2	2
0	3	2	1	0
1	2	3	0	1
2	1	0	3	2
3	0	1	2	3

5.2 Нечёткая логика

Нечёткая логика оперирует с континуальным (несчётным) множеством значений истинности, и подавляющее множество многозначных логик могут иметь гомоморфное вложение в нечёткую логику.

5.2.1 Множества

Нечёткое множество определяется так:

$$\left(\mathrm{M} \in [0,1]^{\mathrm{S}}\right),$$

где $[0,1]$ является отрезком на множестве от числа ноль до числа один, элементы которого являются значениями степени нечёткой принадлежности, а S – произвольное множество.

Для нечётких множеств определены следующие отношения (нечёткой принадлежности т, нечёткого подмножества) и операции (нечёткого объединения, нечёткого пересечения, нечёткой разности, нечёткой симметрической разности, нечёткого дополнения).

$$\left((\chi \text{т} \mathrm{A}) \vee (\neg(\chi \text{т} \mathrm{A}))\right)$$

$$\left((\chi \text{т} \mathrm{A}) \sim (\mathrm{A}(\chi) > 0)\right)$$

$$\left((\mathrm{A}\ \Phi\ \mathrm{B}) \sim (\forall \chi (\mathrm{A}(\chi) \le \mathrm{B}(\chi)))\right)$$

$$\left(((\mathrm{A}\ \text{У}\ \mathrm{B})(\chi)) = \min(\{\mathrm{A}(\chi), \mathrm{B}(\chi)\})\right)$$

$$\left(((\mathrm{A}\ \mathrm{T}\ \mathrm{B})(\chi)) = \max(\{\mathrm{A}(\chi), \mathrm{B}(\chi)\})\right)$$

$$\left(((\mathrm{A} // \mathrm{B})(\chi)) = \max(\{0, \mathrm{A}(\chi) - \mathrm{B}(\chi)\})\right)$$

$$\left((\mathrm{A}\ \text{«}\ \mathrm{B}) = ((\mathrm{A} // \mathrm{B})\ \mathrm{T}\ (\mathrm{B} // \mathrm{A}))\right)$$

$$\left(\tilde{\mathrm{A}}(\chi) = (1 - \mathrm{A}(\chi))\right)$$

Мощность нечёткого множества определяется следующим образом.

$$\left(\|\mathrm{A}\| = \left(\sum\nolimits_{\chi} \mathrm{A}(\chi)\right)\right)$$

Мера нечёткости, которая может быть построена в частности на определении мощности нечёткого множества, выражает нечёткость множества.

$$\frac{\|\mathrm{A}\ \text{У}\ \tilde{\mathrm{A}}\|}{\|\mathrm{A}\ \mathrm{T}\ \tilde{\mathrm{A}}\|}$$

5.2.2 Отношения

Частным видом нечётких множеств являются нечёткие отношения. Среди них выделяют бинарные нечёткие отношения, для которых определены следующие свойства и операции.

Рефлексивность.

$(\mathrm{A}(\chi,\chi) = 1)$

Арефлексивность

$(\mathrm{A}(\chi,\chi) = 0)$

Симметричность.

$(\mathrm{A}(\chi,\lambda) = \mathrm{A}(\lambda,\chi))$

Антисимметричность.

$((\mathrm{A}(\chi,\lambda) = \mathrm{A}(\lambda,\chi)) \to ((\neg(\lambda = \chi)) \to (\mathrm{A}(\lambda,\chi) = 0)))$

Асимметричность.

$(\neg(\mathrm{A}(\chi,\lambda) = \mathrm{A}(\lambda,\chi)))$

Строгая асимметричность.

$(\min(\{\mathrm{A}(\chi,\lambda),\mathrm{A}(\lambda,\chi)\}) = 0)$

Полнота.

$(\max(\{\mathrm{A}(\chi,\lambda),\mathrm{A}(\lambda,\chi)\}) = 1)$

Транзитивность.

$\left(\mathrm{A}(\chi,\lambda) \le \max\left(\left\{\min(\{\mathrm{A}(\chi,\gamma),\mathrm{A}(\gamma,\lambda)\})|\gamma\right\}\right)\right)$

Композиция.

$\left(\mathrm{A}(\chi,\lambda) = \max\left(\left\{\min(\{\mathrm{B}(\chi,\gamma),\mathrm{C}(\gamma,\lambda)\})|\gamma\right\}\right)\right)$

5.2.3 Предикаты. Треугольные нормы

Нечёткий предикат – это нечёткое множество, значения которого интерпретируются как значения истинности. Над нечёткими предикатами определены нечёткие операции отрицания, конъюнкции, дизъюнкции, импликации и другие. В зависимости от предметной области эти операции могут отличаться, т.е. могут существовать разные виды нечёткой конъюнкции, нечёткой дизъюнкции, нечёткой импликации. Эти виды определяются зависимостью выражаемых предикатами свойств и явлений: например явления могут быть причинно-зависимыми, независимыми и альтернативными. Однако, не смотря на различие зависимостей нечёткие логические операции сохраняют некоторые общие свойства для любой предметной области: так например операция нечёткой конъюнкции удовлетворяет свойствам операции, которую называют треугольной нормой или t-нормой. Эти свойства следующие.

$((\mathrm{A} \tilde{\wedge} 0) = 0)$

$((\mathrm{A} \tilde{\wedge} 1) = \mathrm{A})$

$((\mathrm{A} \tilde{\wedge} \mathrm{B}) = (\mathrm{B} \tilde{\wedge} \mathrm{A}))$

$(((\mathrm{A} \tilde{\wedge} \mathrm{B}) \tilde{\wedge} \mathrm{C}) = (\mathrm{A} \tilde{\wedge} (\mathrm{B} \tilde{\wedge} \mathrm{C})))$

$$\left((B \le C) \to \left((A \tilde{\wedge} B) \le (A \tilde{\wedge} C)\right)\right)$$

Когда задана операция нечёткой конъюнкции (треугольная норма), тогда операция нечёткой дизъюнкции (s-норма) может быть выражена через неё с помощью операции нечёткого отрицания.

$$\left(\left(;\ \left(A(\chi) \tilde{\wedge} B(\chi)\right)\right) = \left(\left(;\ A(\chi)\right) \tilde{\vee} \left(;\ B(\chi)\right)\right)\right)$$

Примерами операций нечёткой конъюнкции и нечёткой дизъюнкции являются операции нечёткого пересечени и нечёткого объединения множеств соответственно.

$$\left(\left(A(\chi) \tilde{\wedge} B(\chi)\right) = \min\left(\left\{A(\chi), B(\chi)\right\}\right)\right)$$

$$\left(\left(A(\chi) \tilde{\vee} B(\chi)\right) = \max\left(\left\{A(\chi), B(\chi)\right\}\right)\right)$$

5.2.4 Меры возможности и необходимости

Для множеств событий, явлений, проявляемых свойств можно ввести меры необходимости и возможности проявления этих свойств. Для любых множеств событий мера необходимости N и мера возможности P удовлетворяют следующим соотношениям соответственно.

$$N(A \cap B) \le \min\left(\left\{N(A), N(B)\right\}\right)$$

$$P(A \cap B)\ \mathrm{i}\ \min\left(\left\{P(A), P(B)\right\}\right)$$

5.2.5 Прямой нечёткий вывод

Задача прямого вывода подразумевает известность некоторой пары нечётких предикатов, один из которых рассматривается как посылка, а второй – как правило, обычно первый предикат является унарным, а второй – бинарным. Тогда задача прямого вывода сводится к нахождению композиции межу этими двумя нечёткими предикатами. Результат (следствие) также является нечётким предикатом.

$$\left(B(\gamma) = A(\chi) \tilde{\circ} R(\chi, \gamma)\right)$$

$$B(\gamma) = ?$$

В зависимости от выбранного правила и вида операции композиции результат может соответствовать мере необходимости, либо мере возможности нечёткого логического следствия, либо некоторой другой, например, усреднённой мере. Это вызвано тем, что правило, обычно нельзя построить однозначным образом для зависимостей причин и следствий по известным фактам. Правило обычно строится как некоторая импликация, которая выражает зависимость между наблюдаемыми причинами и следствиями. В силу вида нечётких операций над предикатами таких правил может быть несколько, поэтому такая неоднозначность повышает степень нечёткости результатов нечёткого логического вывода, тогда для представления более полного заключения при прямом нечётком логическом выводе необходимо использовать нечёткие предикаты и множества более высоких порядков. В случае, когда рассматривается правило импликативного вида, исходя из целей получения меры возможности для заключения, можно рассчитать предикат, выражающий правило на основании известных причины и следствия следующим образом.

$$\left(\mathrm{A}(\chi) \tilde{\to} \mathrm{B}(\gamma) = \sup\left(\left\{\delta \middle| \left(\left(\left(\mathrm{A}(\chi) \tilde{\wedge} \delta\right) \le \mathrm{B}(\gamma)\right) \wedge (\delta \le 1)\right)\right\}\right)\right)$$

Затем это уже правило может быть использовано для получения заключения, когда в качестве причины выбирается тот же или другой нечёткий предикат.

$$\left(\mathrm{B}(\gamma) = \sup\left(\left\{\mathrm{A}(\chi) \tilde{\wedge} \mathrm{R}(\chi,\gamma)\right\}\right)\right)$$

5.2.6 Нечёткие множества высших порядков

Нечёткое множество второго порядка может быть определено следующим образом, т.е. областью определения для него выступает множество первого порядка.

$$\left(\mathrm{M} \in [0,1]^{[0,1]^S}\right)$$

Таким образом, областью определения множества следующего порядка является множество предыдущего порядка.

5.2.7 Обратный нечёткий вывод

Задача обратного нечёткого логического вывода является обратной задачей к задаче прямого логического вывода. В качестве исходных данных здесь выступают два нечётких предиката – правило и заключение. Найти требуется множество посылок, которые могут при применении данного правила привести к указанному заключению. Задача обратного нечёткого вывода сложнее задачи прямого нечёткого логического вывода и не всегда имеет решение.

$$\left(\mathrm{B}(\gamma) = \mathrm{A}(\chi) \tilde{\circ} \mathrm{R}(\chi,\gamma)\right)$$

$$\mathrm{A}(\chi) = \ ?$$

Искомые посылки могут быть найдены как нечёткий предикат (множество) первого порядка и выше, либо наиболее общие случаи для посылок могут быть заданы парами минимального и максимального значений для каждого аргумента посылки.

5.2.8 Деффузификация

Деффузификация заключается в переходе от неточных множеств к точным. Зачастую при решении задачи требуется однозначный ответ или точный, поэтому существуют методы, которые позволяют выделить из нечёткого множества тот элемент или такое чёткое множество элементов, которые были бы наиболее подходящими для окончательного ответа. Примером такого метода является выделение центрального элемента множества, которое описывается следующими выражениями, если нечёткое множество определено на векторном пространстве.

$$\left(\hat{\chi} = \frac{\sum \mathrm{A}(\chi) * \chi}{\sum \mathrm{A}(\chi)}\right)$$

$$\left(\hat{\chi} = \frac{\int \mathrm{A}(\chi) * \chi}{\int \mathrm{A}(\chi)}\right)$$

Обратная задача называется задачей фуззификации.

5.3 Теория вычислимости

С целью формального описания понятия (числовой) функции была разработана теория вычислимости (или общерекурсивных функций). Язык и формулы, выражающие всевозможные числовые функции, в этой теории строятся на основе исчисления арифметики, поэтому эту теорию можно рассматривать как развитие исчисления арифметики.

Определяются термы, задающие базовые функции (функция нуля, прибавления единицы, проективные функции).

$$(Z(\chi) = 0)$$

$$(\chi' = (\chi + 1))$$

$$(Un \backslash i(\chi 1, ..., \chi n) = \chi i)$$

Далее с помощью следующих правил и введённых ранее термов строятся остальные функции. Применяются следующие правила.

Правила построения примитивно-рекурсивных функций (суперпозиция и примитивная-рекурсия).

$$(\varphi(\chi 1, ..., \chi n) = \psi(\varphi 1(\chi 1, ..., \chi n), ..., \varphi m(\chi 1, ..., \chi n)))$$

$$\varphi(\chi 1, ..., \chi n, \lambda) = \begin{cases} \phi(\chi 1, ..., \chi n) \big| (\lambda = 0) \\ \psi(\chi 1, ..., \chi n, \delta, \varphi(\chi 1, ..., \chi n, \delta)) \big| ((\lambda = \delta + 1) \wedge (\delta = 0)) \end{cases}$$

Правило построения общерекурсивных функций (из уже построенных) – μ-оператор, который возвращает минимальное из всех значений λ для заданной функции φ при условии, что выполняется равенство, и λ существует для любых $\chi 1, ..., \chi n$.

$$\mu\lambda(\varphi(\chi 1, ..., \chi n, \lambda) = 0) | \forall \chi 1 ... \forall \chi n \exists \lambda$$

Много известных задач могут быть сведены к задаче вычисления той или иной функции. Принимается тезис (тезис Чёрча), что любая вычислимая функция, может быть вычислена как общерекурсивная функция. Формально этот тезис может быть записан на языке интуиционистской логики.

5.4 Теория алгоритмов

При решении задач часто используют понятие алгоритма. Теория алгоритмов вводят формальные средства для уточнения этого понятия, и решает вопрос о существовании алгоритма (алгоритмического решения) для решения той или иной задачи. Одним из формальных способов уточнения понятия алгоритма является такая абстрактная модель как машина Тьюринга.

Машина Тьюринга может быть задана семью компонентами

$$\langle \mathsf{A}, s0, \mathsf{Q}, q0, q1, \mathsf{T}, \tau \rangle,$$

соответственно – конечным алфавитом, пустым символом алфавита, конечным множеством внутренних состояний, начальным состоянием, конечным состоянием, множеством переходов (влево, вправо и никуда), программой, а также – бесконечной лентой (вектором), которая хранит символы алфавита в каждой ячейке, относительно которых совершаются переходы.

Программа машины Тьюринга имеет следующий вид.

$$\left(\tau \in \mathsf{A} \times \mathsf{T} \times \mathsf{Q}^{\mathsf{A} \times (\mathsf{Q} / \{q0\})}\right)$$

Если соответствие является функции, то машину Тьюринга называют детерминированной машиной Тьюринга, иначе – недетерминированной. Принимается тезис, что любая вычислимая функция может быть вычислена на машине Тьюринга.

Понятие машины Тьюринга может быть формализовано в рамках исчисления арифметики. При такой формализации символы алфавита, состояния, переходы, программа машины Тьюринга представляются натуральными числами, или формулами, которые содержат натуральные числа. Это возможно в силу того, что все компоненты машины Тьюринга являются счётными, так же как и множество натуральных чисел. Вследствие этого в исчислении арифметики можно описывать свойства относительно машины Тьюринга в виде формул. Оказалось, что не для всех формул, выражающих свойства относительно машины Тьюринга можно построить доказательство (формальный вывод этих формул). Например, предикат $(\exists x T(a,a,x))$, который выражает существование момента (шага) x, в который должна остановиться та или иная машина Тьюринга a на исходных данных a, не может быть разрешим, т.е. построено доказательство этого предиката в рамках исчисления арифметики высказываний. В силу этого исчисление высказываний является неполным. Свойство этого исчисления формулируется в теореме Гёделя.

Теорема Гёделя. Непротиворечивая формальная теория (исчисление), формализующая арифметику, неполна.

5.5 Теория сложности

Теории сложности рассматривает свойства алгоритмов (алгоритмический решений задач). К таким свойствам могут относиться, требуемые алгоритмам ресурсы – память, время и другие. Алгоритмы работают с исходными данными, поэтому вначале в теории сложности следует рассмотреть понятие кодировки, с помощью которой кодируются данные для той или иной задачи.

Любая кодировка ε удовлетворяет следующим свойствам:

$$\left(\varepsilon \in \left(\mathsf{L}^{\mathsf{C}} \cap \mathsf{C}^{\mathsf{L}}\right)\right)$$

$$\left(\exists \rho \left(\forall I \left((I \in \mathsf{C}) \to \left(t\left(\varepsilon^{-1}\left(\varepsilon(I)\right)\right) < \rho\left(\left|\varepsilon(I)\right|\right)\right)\right)\right)\right),$$

где L – язык, C – класс (множество) задач, ρ – полином.

Кодировка называется неизбыточной тогда и только тогда, когда выполняется следующее.

$$\left(\forall\varepsilon 1\left(\exists\rho\left(\rho\left(\left|\varepsilon(I)\right|<\rho\left(\left|\varepsilon 1(I)\right|\right)\right)\right)\right)\right)$$

Классы полиномиально разрешимых и недетерминировано-полиномиально разрешимых задач и соответствующих им алгоритмов решения определяются следующим образом:

$$\mathsf{P} = \left\{\mathsf{C}\,\middle|\,\exists\varepsilon\exists\tau\exists\rho\forall n\left(Td\left(\langle\tau,\mathsf{C},n\rangle\right)<\rho(n)\right)\right\}$$

$$\mathsf{NP} = \left\{\mathsf{C}\,\middle|\,\exists\varepsilon\exists\tau\exists\rho\forall n\left(Tnd\left(\langle\tau,\mathsf{C},n\rangle\right)<\rho(n)\right)\right\}$$

где $Td\left(\langle\tau,\mathsf{C},n\rangle\right)$ – максимальное из времён успешного выполнения алгоритма τ на детерминированной машине Тьюринга для каждого слова длины n, кодирующего задачу из класса C в кодировке ε, а $Tnd\left(\langle\tau,\mathsf{C},n\rangle\right)$ – максимальное из времён успешного выполнения алгоритма τ на недетерминированной машине Тьюринга для каждого слова длины n, кодирующего задачу из класса C в кодировке ε.

5.6 Интуиционистская логика

В связи с ограниченностью рассмотренных, например, в исчислении арифметики средств формализации и в связи с более точной формализацией понятий разрешимых и неразрешимых задач математическая логика получила развитие в неклассических направлениях. Одним из результатов движения в этом направлении является интуиционистская логика. Интуиционистская логика отличается от классической в исчислении высказываний тем, что в ней отсуствует аксиома или теорема удаления двойного отрицания (отсутствет соответствующее правило), как следствие, в интуиционистком исчислении высказываний недоказуем закон «исключённого третьего». Однако, не смотря на кажущуюся слабость такого исчисления, все аксиомы и доказумые утверждения классического исчисления предикатов могут быть «погружены» в интуиционистскую логику. Также следует отметить, что интуиционистская логика не отказывается от принципа сохранности (когда истинность формулы не зависит от времени).

Акоматика интуиционисткого исчисления предикатов следующая.

	введение	удаление
$\rightarrow$	$(\alpha\rightarrow(\beta\rightarrow\alpha))$	$((\gamma\rightarrow\alpha)\rightarrow((\gamma\rightarrow(\alpha\rightarrow\beta))\rightarrow(\gamma\rightarrow\beta)))$
$\wedge$	$(\alpha\rightarrow(\beta\rightarrow(\alpha\wedge\beta)))$	$((\alpha\wedge\beta)\rightarrow\alpha)$ $((\alpha\wedge\beta)\rightarrow\beta)$
$\vee$	$(\alpha\rightarrow(\alpha\vee\beta))$ $(\alpha\rightarrow(\beta\vee\alpha))$	$((\alpha\rightarrow\gamma)\rightarrow((\beta\rightarrow\gamma)\rightarrow((\alpha\vee\beta)\rightarrow\gamma)))$
$\neg$	$((\alpha\rightarrow\beta)\rightarrow((\alpha\rightarrow(\neg\beta))\rightarrow(\neg\alpha)))$	
$\sim$	$((\alpha\rightarrow\beta)\rightarrow((\beta\rightarrow\alpha)\rightarrow(\alpha\sim\beta)))$	$((\alpha\sim\beta)\rightarrow(\alpha\rightarrow\beta))$ $((\alpha\sim\beta)\rightarrow(\beta\rightarrow\alpha))$
$\forall$		$((\forall\chi\alpha(\chi))\rightarrow\alpha(\tau))$

$\exists$	$(\alpha(\chi)\rightarrow(\exists\chi\alpha(\chi)))$	

Правила вывода такие же, какие были использованы в определении классического исчисления предикатов: правило прямого заключения, правило обобщения и правило конкретизации.

В интуиционистской логике, исходя из её семантики, корректно формализуются тезисы о вычислимости, например – тезис Чёрча.

Тезис Чёрча.

$$((\forall\chi(\exists\gamma(\alpha(\chi,\gamma))))\rightarrow(\exists\varphi(\forall\chi(\exists\gamma(((\varphi(\chi)=\gamma))\wedge\alpha(\chi,\gamma))))))$$

5.7 Немонотонный вывод

Другими результатами движения в направлении неклассических логик являются логические теории (немонотонные логики), в которых используется немонотонный вывод. Немонотонность вывода связана с тем, что вывод необходимо делать исходя из неполной информации, поэтому при немонотонном выводе допустим вывод не только общезначимых формул, но и возможно только нейтральных, т.е. немонотонные правила вывода позволяют выводить выполнимые формулы. Немонотонное отношение выводимости $\mid\!\sim$ не обладает свойством монотонности, т.е. существуют такие множества формул G, F, E, что выполняется следующее свойство.

Немонотонность отношения выводимости.

$$\neg((G\mid\!\sim E)\rightarrow((G\cup F)\mid\!\sim E))$$

Однако отношение немонотонного вывода может иногда удовлетворять свойству полумонотонности, когда для любых множеств G, F, E формул верно

$$(((G\mid\!\sim E)\wedge(G\mid\!\sim F))\rightarrow((G\cup F)\mid\!\sim E)).$$

5.7.1 Логики умолчаний

Одними из логических теорий, в которых используются немонотонный логический вывод, являются теории с умолчаниями. Теория с умолчаниями задаётся парой – множеством замкнутых логических формул и множеством умолчаний – правил немонотонного вывода.

В общем виде молчание выглядит следующим образом:

$$(\{\alpha\}/\{(\neg\beta)\}\mid\!\sim\{\gamma\}),$$

где α – требование, β – обоснование, γ – следствие.

Среди умолчаний выделяют нормальные и полунормальные как и среди теорий с умолчаниям.

Нормальное умолчание.

$$(\{\alpha\}/\{(\neg\beta)\}\mid\!\sim\{\beta\})$$

Полунормальное умолчание.

$$(\{\alpha\}/\{(\neg(\beta\wedge\gamma))\}\mid\!\sim\{\beta\})$$

Умолчания теории с умолчаниями позволяют строить расширение такой теории.

Пусть Ldflt – язык, на котором записываются формулы теории с умолчаниями, тогда расширением теории Δ с умолчаниями, заданной парой $\langle F, D\rangle$, будет такое множество E тогда и только тогда, когда выполняется следующее:

$$\left(Th(\langle L, X\rangle) = \left\{\rho \middle| \left(\left(\left(\rho \in L\right) \wedge \left(X \models \{\rho\}\right)\right) \wedge \left(X \subseteq L\right)\right)\right\}\right)$$

$$\left(\left(\Gamma(S) = S\right) \to \left(\neg\left(\exists Q\left(\left(Q = \Gamma(Q)\right) \wedge \left(Q \subset S\right)\right)\right)\right)\right)$$

$$\left(F \subseteq \Gamma(S)\right)$$

$$\left(Th\left(\langle Ldflt, \Gamma(S)\rangle\right) = \Gamma(S)\right)$$

$$\left(\left(\left(\left(\{\alpha\}/\{(\neg\beta)\} \mid\!\sim \{\gamma\}\right) \in D\right) \wedge \left(\alpha \in \Gamma(S)\right) \wedge \left(\neg\left((\neg\beta) \in S\right)\right)\right) \to \left(\gamma \in \Gamma(S)\right)\right)$$

$$\left(\Gamma(E) = (E \cap Ldflt)\right),$$

где ρ – замкнутая формула.

Доказательство множества формул $\{f\}$ (формулы f), в теории Δ с умолчаниями $(\Delta = \langle D, F\rangle)$ определяется следующим образом: $\langle D0, D1, \ldots, Dk\rangle$, при условии, что верно:

$$(Di \in D)$$

$$(f \in Ldflt)$$

$$\left(\left(F \cup \{CC(D0)\}\right) \models \{f\}\right)$$

$$\left(\left(F \cup \{CC(Di)\}\right) \models \{DC(Di-1)\}\right),$$

$$(Dk = \varnothing)$$

$$\left(\neg\left(\left\{\left(F \cup \{CC(Di)\}\right) \middle| 0 \le i \le k\right\} \models \hat{}\ \right)\right)$$

где $CC(Di)$ - конъюнкция следствий умолчания Di, $DC(Di-1)$ – конъюнкция требований умолчания $Di-1$.

5.7.2 Немонотонная логика Мак-Дермотта

Немонотонная логика Мак-Дермотта может быть построена на базе одной из модальных систем, используя или не используя дополнительную аксиомную схему («схему Баркан»).

$$\left(\left(\forall\chi\left(!\ \alpha(\chi)\right)\right) \to \left(!\ \left(\forall\chi\alpha(\chi)\right)\right)\right)$$

Множество теорем B логики Мак-Дермотта, следующих из множества посылок A и множества аксиом S, где Lnm – язык логики Мак-Дермотта, а M – модальный квантор допущения, определяется следующим образом.

$$(B \subseteq Lnm)$$

$$\left(Th(\langle S, A\rangle) = \left\{\rho \middle| \left(\left(\rho \in Lnm\right) \wedge \left(A \models_S \{\rho\}\right)\right)\right\}\right)$$

$$\left(Hyp(\langle A, B\rangle) = \left(\left\{\mathrm{M}\delta \middle| \left(\left(\delta \in Lnm\right) \wedge \left(\neg\left((\neg\delta) \in B\right)\right)\right)\right\} / \left(\{\mathrm{M}\delta(\chi)\} \cup Th(\langle S, A\rangle)\right)\right)\right)$$

$$\left(TH(\langle S, A\rangle) = Lnm \cap \bigcap\left\{B \middle| \left(B = Th\left(\left\langle S, \left(A \cup Hyp(\langle A, B\rangle)\right)\right\rangle\right)\right)\right\}\right)$$

Для немонотонной логики Мак-Дермотта справедливо следующее.

$$\left(\left(A \mid\!\sim_{S5} B\right) \to \left(A \models_{S5} B\right)\right)$$

5.8 Вывод по аналогии.

Ещё одним примером неклассического логического вывода является вывод по аналогии.

Правило прямого вывода (заключения) по аналогии формально может выглядеть так:

$$\left(\left(\left(G\cup\{\alpha,\varphi\}\right)\hbar\left(G\cup\{\beta\}\right)\right)\to\left(\left(G\cup\{\sigma(\alpha,\lambda),\sigma(\varphi,\psi),\sigma(\beta,\gamma),\alpha,\varphi,\lambda\}\right)\hbar\left(G\cup\{\gamma\}\right)\right)\right),$$

где σ – бинарное отношение подобия на множестве термов и формул.

6. Формализация математики

Одним из приложений в рамках аксиоматического подхода является формализация понятий в рамках всей математики. Построено несколько различных аксиоматических систем (теорий), которые позволяют формализовывать довольно большое множество математических абстракций. Одной из таких систем является аксиоматическая система Цермело-Френкеля.

6.1 Аксиоматика Цермело–Френкеля.

Вначале определим (переопределим для этой системы) некоторые обозначения.

Оператор i.

$$\left(\left\{\chi\middle|\alpha(\chi)\right\}=\mathrm{i}\lambda\left(\forall\chi\left((\chi\in\lambda)\sim\alpha(\chi)\right)\right)\right)$$

Пустое множество.

$$\left(\varnothing=\mathrm{i}\chi\left(\forall\gamma\left(\neg(\gamma\in\chi)\right)\right)\right)$$

Двуэлементное множество.

$$\left(\{\chi,\gamma\}=\left\{\lambda\middle|\left((\lambda=\chi)\vee(\lambda=\gamma)\right)\right\}\right)$$

Объединение двух множеств.

$$\left((\chi\cup\gamma)=\left\{\lambda\middle|\left((\lambda\in\chi)\vee(\lambda\in\gamma)\right)\right\}\right)$$

Пересечение двух множеств.

$$\left((\chi\cap\gamma)=\left\{\lambda\middle|\left((\lambda\in\chi)\wedge(\lambda\in\gamma)\right)\right\}\right)$$

Отношение подмножества.

$$\left((\chi\subseteq\gamma)\sim\forall\lambda\left((\lambda\in\chi)\to(\lambda\in\gamma)\right)\right)$$

Отметим, что понятие вектора из двух элементов было определено на примерах в первой главе.

Декартово произведение двух множеств.

$$\left((\chi\times\gamma)=\left\{\lambda\middle|\left(\exists\delta\left(\exists\sigma\left((\lambda=\langle\delta,\sigma\rangle)\wedge(\delta\in\chi)\wedge(\sigma\in\gamma)\right)\right)\right)\right\}\right)$$

Понятие функции.

$$\left(Fnc(\delta)\sim\left(\left(\exists\gamma\left(\delta\subseteq(\gamma\times\gamma)\right)\right)\wedge\left(\forall\sigma\left(\forall\chi\left(\forall\lambda\left(\left((\langle\sigma,\chi\rangle\in\delta)\wedge(\langle\sigma,\lambda\rangle\in\delta)\right)\to(\chi=\lambda)\right)\right)\right)\right)\right)\right)$$

Понятие образа функции δ при аргументе χ.

$$\left(\delta(\chi)=\mathrm{i}\gamma\left(\langle\chi,\gamma\rangle\in\delta\right)\right)$$

Понятие бесконечности.

$$\left(Inf(\lambda) \sim \left((\varnothing \in \lambda) \wedge \left(\forall \chi \left((\chi \in \lambda) \to (\chi \cup \{\chi\} \in \lambda)\right)\right)\right)\right)$$

Тогда аксиоматика Цермело-Френкеля может быть записана следующим образом.

Аксиома объёмности.

$$\left(\forall \chi \left(\left((\chi \in \gamma) \sim (\chi \in \lambda)\right) \to (\gamma = \lambda)\right)\right)$$

Аксиома пары.

$$\left(\exists \delta \left(\forall \chi \left((\chi \in \delta) \sim \left((\chi = \lambda) \vee (\chi = \gamma)\right)\right)\right)\right)$$

Аксиома суммы.

$$\left(\exists \delta \left(\forall \chi \left((\chi \in \delta) \sim \exists \gamma \left((\gamma \in \lambda) \wedge (\chi \in \gamma)\right)\right)\right)\right)$$

Аксиома степени.

$$\left(\exists \delta \left(\forall \chi \left((\chi \in \delta) \sim (\chi \subseteq \lambda)\right)\right)\right)$$

Аксиома выделения.

$$\left(\exists \delta \left(\forall \chi \left((\chi \in \delta) \sim \left((\chi \in \lambda) \wedge \alpha(\chi)\right)\right)\right)\right)$$

Аксиома бесконечности.

$$\left(\exists \delta Inf(\delta)\right)$$

Аксиома выбора.

$$\left(\forall \gamma \left(\exists \delta \left(Fnc(\delta) \wedge \left(\forall \chi \left(\left((\chi \in \lambda) \wedge \left(\neg(\chi = \varnothing)\right)\right) \to \left(\delta(\chi) \in \chi\right)\right)\right)\right)\right)\right)$$

Аксиома фундирования.

$$\left(\forall \chi \left(\left(\neg(\chi = \varnothing)\right) \to \left(\exists \lambda \left((\lambda \in \chi) \wedge \left((\lambda \cap \chi) = \varnothing\right)\right)\right)\right)\right)$$

Аксиома подстановки Френкеля.

$$\left(\exists \delta \left(\forall \chi \left((\chi \in \delta) \sim \exists \gamma \left((\gamma \in \lambda) \vee \left(\chi = \mathrm{i}\tau\alpha(\tau, \gamma)\right)\right)\right)\right)\right)$$