При написании этих лекций я использовал книгу:

Stanley N. Burris. Logic for Mathematics and Computer Science, Prentice Hall, 1998.

1 Резолюция в логике высказываний

Резолюция — это правило вывода, используемое для того, чтобы показать, что множество булевых формул вида

$$(\phi_1 \vee \phi_2 \vee \cdots \vee \phi_m), \tag{1}$$

где каждая из

$$\phi_1, \phi_2 \ldots \phi_m$$

либо является пропозициональной переменной, или является отрицанием пропозициональной переменной, не является выполнимым. Напомним, что булева формула называется выполнимой, если она истинна при некоторых значениях её переменных. Множество булевых формул называется выполнимым, если существуют такие значения входящих хотя бы в одну из этих формул переменных, что при этих значениях переменных каждая из формул этого множества истинна.

Истинность секвенции

$$\Phi_1 \ldots \Phi_m \vdash \Phi$$

равносильна, как известно, тождественной истинности формулы

$$((\Phi_1 \& \ldots \& \Phi_m) \to \Phi)$$

и, значит, равносильна невыполнимости формулы

$$\neg((\Phi_1 \& \ldots \& \Phi_m) \to \Phi).$$

Для последней формулы можно построить эквивалентную кнф. Невыполнимость этой кнф равносильна невыполнимости множества её конъюнктивных членов. Каждый же конъюнктивный член является элементарной конъюнкцией и, значит, имеет вид (1).

Заметим, что вместо элементарной дизъюнкции можно рассматривать множество её дизъюнктивных членов.

1.1 Клаузулы

Каждая пропозициональная переменная называется положительным литералом. Каждое отрицание пропозициальной переменной называется отрицательным литералом. Литерал — это положительный или отрицательный литерал.

Конечное множество литералов называется клаузулой. Литералы, входящие в клаузулу, называются её членами. При заданных значениях переменных клаузула называется истинной, если при этих значениях переменных хотя бы один член этой клаузулы истинен. Пустая клаузула (пустое множество литералов) не является истинным ни при каких значениях переменных. Множество клаузул истинно при заданных значениях переменных, если каждая клаузула, входящая в это множество, истинна при этих значениях переменных. Множество клаузул выполнимо (мы также иногда будем говорить совместно), если существуют такие значения переменных, при которых это множество истинно.

1.2 Резолюция

Пусть $\Diamond L$ есть переменная, если L есть отрицание этой переменной, и $\Diamond L$ есть отрицание переменной, если L есть эта переменная. Литерал $\Diamond L$ называется двойственным к L.

Для доказательства невыполнимости множества клаузул можно использовать следующее правило:

$$\frac{C \cup \{L\}, D \cup \{\Diamond L\}}{C \cup D},$$

в котором L — литерал, C и D — клаузулы, не содержащие ни L, ни $\Diamond L$. Смысл этого правила состоит в том, что имея совместное множество клаузул, содержащее клаузулы вида $C \cup \{L\}$ и $D \cup \{\Diamond L\}$, к этому множеству клаузул можно добавить клаузулу $C \cup D$, не теряя совместности. При этом $C \cup D$ называется резольвентой, а само правило называется резолюцией. Литерал L называется литералом этой резолюции.

Корректность этого правила следует из того, что если при заданных значениях переменных клаузулы $C \cup \{L\}$ и $D \cup \{\Diamond L\}$ истинны, то клаузула $C \cup D$ при этих значениях переменных тоже истинна.

Пример 1.1. Рассмотрим следующее множество клаузул:

$$\{A, \neg B\},\ \{B, C\},\ \{\neg A\},\ \{\neg C\}.$$

Применяя резолюции, добавляем резольвенты к заданному множеству.

1. $\{A, \neg B\}$,
2. $\{B, C\}$,
3. $\{A, C\}$, резолюция, 1,2,
4. $\{\neg A\}$,
5. $\{C\}$, резолюция, 3, 4,
6. $\{\neg C\}$,
7. $\{\}$, резолюция, 5, 6.

Таким образом мы получаем пустую клаузулу в полученном множестве клаузул. Следовательно, полученное множество клаузул несовместно. Значит, и первоначальное множество клаузул было несовместным.

На самом деле, можно предложить более организованный алгоритм получения пустой клаузулы из заданного множества клаузул.

Клаузулу K назовём корректной, если она не содержит никакой пары двойственных членов.

Замечание 1.1. *Множество клаузул совместно тогда и только тогда, когда его подмножество, составленное из всех корректных клаузул, совместно.*

Доказательство. Достаточно заметить, что каждая некорректная клаузула истинна при любых значениях переменных. ∎

Множество клаузул называется обрабатываемым, если одна из клаузул этого множества содержит некоторый литерал и одна из клаузул этого же множества содержит двойственный литерал.

Пусть $A_1, \ldots, A_n$ — все переменные, встречающиеся в заданном обрабатываемом множестве клаузул M_1. Используя предыдущее замечание 1.1, можно предполагать, что все клаузулы из M_1 корректны. При этом мы предполагаем, что A_1 входит в некоторую клаузулу из M_1 и $\neg A_1$ входит в некоторую клаузулу из M_1.

Разобьём M_1 на три подмножества M_1^1, M_1^2 и M_1^3. В M_1^1 поместим все те клаузулы из M_1, членом которых является A_1. В M_1^2 поместим все те клаузулы из M_1, членом которых является $\neg A_1$. В M_1^3 поместим все те клаузулы из M_1, членом которых не является ни A_1, ни $\neg A_1$. Рассмотрим все корректные клаузулы, которые можно получить резолюциями с литералом A_1 из клаузулы, входящей в M_1^1, и клаузулы, входящей в M_1^2. Множество всех так полученных корректных клаузул обозначим через M_{22}. Пусть $M_2 = (M_1^3 \cup M_{22})$. Легко заметить, что M_2 составлено из корректных клаузул, не содержащих ни A_1, ни $\neg A_1$.

Теорема 1.1. *Множество клаузул M_2 совместно тогда и только тогда, когда множество клаузул M_1 совместно.*

Доказательство. Пусть M_2 совместно. M_1^1 и M_1^2 непусты. Рассмотрим такое распределение значений переменных $A_2,\ldots, A_n$, которое выполняет все клаузулы из M_2. Из замечания 1.1 следует, что это же распределение значений переменных $A_2,\ldots, A_n$ выполняет все клаузулы, которые можно получить резолюциями с литералом A_1 из клаузулы, входящей в M_1^1, и клаузулы, входящей в M_1^2. Рассмотрим клаузулы $C \cup \{A_1\}$ из M_1^1 и клаузулы $D \cup \{\neg A_1\}$ из M_1^2. Либо D не выполняется рассматриваемым распределением значений переменных для некоторой клаузулы $D \cup \{\neg A_1\}$ из M_1^2 и тогда C и тем более $C \cup \{A_1\}$ выполняются рассматриваемым распределением при любом значении A_1, в частности, если значение A_1 равно нулю, или D выполняется рассматриваемым распределением значений переменных для любой клаузулы $D \cup \{\neg A_1\}$ из M_1^2 и тогда все клаузулы из M_1 выполняются при рассматриваемом распределении значений переменных $A_2,\ldots, A_n$ и при значении A_1, равным 1. ∎

Если M_2 является обрабатываемым множеством клаузул, то далее рассматриваем M_2 вместо M_1 и некоторое такое переупорядочивание $A_{i_2},\ldots, A_{i_n}$ списка переменных $A_2,\ldots, A_n$, для которого A_{i_2} входит в некоторую клаузулу из M_2 и $\neg A_{i_2}$ входит в некоторую клаузулу из M_2, вместо списка $A_1,\ldots, A_n$ и повторяем шаг алгоритма.

Не более, чем через n шагов, мы либо получим необрабатываемое множество клаузул, или получим множество клаузул, содержащее пустую клаузулу. В последнем случае первоначальное множество клаузул несовместно.

Теорема 1.2. *Любое необрабатываемое множество клаузул совместно.*

Доказательство. Если переменная входит в одну из клаузул этого множества, придадим ей значение 1. Если же отрицание переменной входит в одну из клаузул этого множества, придадим этой переменной значение 0. ∎

Таким образом, если в результате работы алгоритма получится необрабатываемое множество клаузул, то первоначальное множество клаузул совместно.

1.3 Примеры

Пример 1.2. Рассмотрим секвенцию

$$(A \to B)\ (B \to C)\ (C \to D) \vdash (A \to D).$$

Ложность этой секвенции равносильна, как уже отмечалось, выполнимости формулы

$$\neg(\neg((\neg A \vee B)\&(\neg B \vee C)\&(\neg C \vee D)) \vee \neg A \vee D).$$

Последняя формула эквивалентна формуле

$$(((\neg A \vee B)\&(\neg B \vee C)\&(\neg C \vee D))\ \&\ A\ \&\ \neg D). \qquad (2)$$

Вместо формулы (2) можно рассматривать множество клаузул

$$\{\{\neg A,\ B\},\ \{\neg B,\ C\},\ \{\neg C,\ D\},\ \{A\},\ \{\neg D\}\}.$$

Таким образом для доказательства истинности рассматриваемой секвенции достаточно доказать невыполнимость полученного множества клаузул.

Докажем это, используя предложенный алгоритм.

$$\textbf{Шаг 1}.\ M_1 = \{\{\neg A,\ B\},\ \{\neg B,\ C\},\ \{\neg C,\ D\},\ \{A\},\ \{\neg D\}\};$$
$$M_1^1 = \{\{A\}\},\ \ M_1^2 = \{\{\neg A,\ B\}\},\ \ M_1^3 = \{\{\neg B,\ C\},\ \{\neg C,\ D\},\ \{\neg D\}\};$$
$$\textbf{Шаг 2}.\ M_2 = \{\{B\},\ \{\neg B,\ C\},\ \{\neg C,\ D\},\ \{\neg D\}\};$$
$$M_2^1 = \{\{B\}\},\ \ M_2^2 = \{\{\neg B,\ C\}\},\ \ M_2^3 = \{\{\neg C,\ D\},\ \{\neg D\}\};$$
$$\textbf{Шаг 3}.\ M_3 = \{\{C\},\ \{\neg C,\ D\},\ \{\neg D\}\};$$
$$M_3^1 = \{\{C\}\},\ \ M_3^2 = \{\{\neg C,\ D\}\},\ \ M_3^3 = \{\{\neg D\}\};$$
$$\textbf{Шаг 4}.\ M_4 = \{\{D\},\ \{\neg D\}\};$$
$$M_4^1 = \{\{D\}\},\ \ M_4^2 = \{\{\neg D\}\}, M_4^3 = \emptyset;$$
$$\textbf{Шаг 5}.\ M_5 = \{\{\}\}.$$

Пример 1.3. Рассмотрим множество клаузул

$$\{\{\neg A,\ B\},\ \{\neg E,\ B\},\ \{\neg C,\ D\},\ \{C,\ F,\ A\},$$
$$\{\neg D,\ F\},\ \{\neg B,\ \neg F\},\ \{F,\ \neg A,\ C\},\ \{B,\ \neg F,\ A,\ E\}\}.$$

Докажем невыполнимость рассматриваемого множества клаузул.

Докажем это, используя предложенный алгоритм.

Шаг 1. $M_1 = \{\{\neg A,\ B\},\ \{\neg E,\ B\},\ \{\neg C,\ D\},\ \{C,\ F,\ A\},$
$\{\neg D,\ F\},\ \{\neg B,\ \neg F\},\ \{F,\ \neg A,\ C\},\ \{B,\ \neg F,\ A,\ E\}\};$
$M_1^1 = \{\{C,\ F,\ A\},\ \{B,\ \neg F,\ A,\ E\}\},$
$M_1^2 = \{\{\neg A,\ B\}\},\ \{F,\ \neg A,\ C\},$
$M_1^3 = \{\{\neg E,\ B\},\ \{\neg C,\ D\},\ \{\neg D,\ F\},\ \{\neg B,\ \neg F\}\};$
Шаг 2. $M_2 = \{\{B,\ C,\ F\},\ \{C,\ F\},\ \{B,\ \neg F,\ E\},$
$\{\neg E,\ B\},\ \{\neg C,\ D\},\ \{\neg D,\ F\},\ \{\neg B,\ \neg F\}\};$
$M_2^1 = \{\{B,\ C,\ F\},\ \{B,\ \neg F,\ E\},\ \{\neg E,\ B\}\},$
$M_2^2 = \{\{\neg B,\ \neg F\}\},$
$M_2^3 = \{\{C,\ F\},\ \{\neg C,\ D\},\ \{\neg D,\ F\}\};$
Шаг 3. $M_3 = \{\{\neg F,\ E\},\ \{\neg F,\ \neg E\},\ \{C,\ F\},\ \{\neg C,\ D\},\ \{\neg D,\ F\}\};$
$M_3^1 = \{\{C,\ F\}\},\ M_3^2 = \{\{\neg C,\ D\}\},\ M_3^3 = \{\{\neg F,\ E\},\ \{\neg F,\ \neg E\},\ \{\neg D,\ F\}\};$
Шаг 4. $M_4 = \{\{F,\ D\},\ \{\neg F,\ E\},\ \{\neg F,\ \neg E\},\ \{\neg D,\ F\}\};$
$M_4^1 = \{\{F,\ D\}\},\ \ M_4^2 = \{\{F,\ \neg D\}\}, M_4^3 = \{\{\neg F,\ E\},\ \{\neg F,\ \neg E\}\};$
Шаг 5. $M_5 = \{\{F\},\ \{\neg F,\ E\},\ \{\neg F,\ \neg E\}\};$
$M_5^1 = \{\{E,\ \neg F\}\},\ \ M_5^2 = \{\{\neg F,\ \neg E\}\},\ \ M_5^3 = \{\{F\}\};$
Шаг 6. $M_6 = \{\{\neg F\},\ \ \{F\}\};\ \ M_6^1 = \{\{F\}\},\ \ M_6^2 = \{\{\neg F\}\},\ \ M_6^3 = \emptyset;$
Шаг 7.$M_7 = \{\{\}\}.$

1.4 Компактность

Теорема 1.3 (компактности). *Множество клаузул совместно тогда и только тогда, когда каждое его конечное подмножество совместно.*

Доказательство. Пусть каждое конечное подмножество множества клаузул T совместно. Тогда для каждого n совместно подмножество T_n этого множества клаузул, содержащее те и только те клаузулы, в которые не входят пропозициональные переменные, отличные от первых n. Набор ϵ_n значений переменных, делающий все клаузулы из T_n истинными, определяется неоднозначно. Выберем один из возможных наборов. Для каждого n можно выбрать такое бесконечное подмножество натуральных чисел ω_n, что $\omega_n \supseteq \omega_m$ для $n < m$ и значение, которое ϵ_i приписывает первым n переменным, одинаково для всех $i \in \omega_n$. Пусть $n(i)$ —

наименьший элемент в ω_i. Взяв теперь в качестве ϵ такой набор значений переменных, который i-ой переменной приписывает то же значение, что и $\epsilon_{n(i)}$. Очевидно, что, приписывая переменным набор значений ϵ, мы сделаем все клаузулы из T истинными. ∎

2 Унификация

2.1 Термы

Напомним, что для заданной сигнатуры $\mathcal{L}$ понятие терма определяется следующим образом.

В термах используются предметные переменные — переменные, принимающие значения из основного множества рассматриваемой алгебраической системы сигнатуры $\mathcal{L}$. Мы будем обозначать предметные переменные строчными латинскими буквами x, y, z, возможно, с индексами.

Определение 2.1 (терма).

а). Каждая предметная переменная и каждый нульместный символ операции сигнатуры $\mathcal{L}$ — терм сигнатуры $\mathcal{L}$. В предметную переменную входит она сама, а другие переменные не входят. В символ операции предметные переменные не входят.

б). Если f — n-местный символ операции сигнатуры $\mathcal{L}$, а $t_1, \ldots, t_n$ являются термами сигнатуры $\mathcal{L}$, то $f(t_1, \ldots, t_n)$ — тоже терм сигнатуры $\mathcal{L}$. Переменная входит в $f(t_1, \ldots, t_n)$ тогда и только тогда, когда она входит хотя бы в один из термов $t_1, \ldots, t_n$.

в). Каждая вещь является термом сигнатуры $\mathcal{L}$ только если это можно доказать, используя пункты а) и б).

г). Терм, не содержащий переменных, называется замкнутым.

Обычно при двухместном f вместо $f(t_1, t_2)$ пишут $(t_1 f t_2)$. Например, пишут $(x_1 + x_2)$ вместо $+(x_1, x_2)$.

Например, $((x_1 + x_2) \times x_3)$ — арифметический терм, в котором $+$ и $\times$ — символы двухместных операций сложения и умножения.

Заметим, что если сигнатура не содержит символов операций, то каждый терм является просто предметной переменной, так как в этом случае пункт б) определения нельзя использовать.

Если X - это множество всех переменных, входящих в терм t, то терм t называется термом над X.

Далее нам удобно использовать некоторую другую запись термов, называемую *прямой польской записью*.

Определение 2.2 (терма в прямой польской записи).

а). Каждая предметная переменная и каждый нульместный символ операции сигнатуры $\mathcal{L}$ *— терм сигнатуры* $\mathcal{L}$.

б). Если f *—* n*-местный символ операции сигнатуры* $\mathcal{L}$, *а* $t_1,\ldots, t_n$ *являются термами сигнатуры* $\mathcal{L}$, *то* $ft_1\ldots t_n$ *— тоже терм сигнатуры* $\mathcal{L}$.

В дальнейшем мы будем использовать только термы в прямой польской записи, не оговаривая это специально.

Пример 2.1. Пусть сигнатура $\mathcal{L}$ есть $\langle f^{(2)},\ g^{(3)}\rangle$. Тогда терм

$$g(f(g(x, y, z), f(u, u)), z, f(f(x, u), z))$$

в прямой польской записи имеет вид:

$$gfgxyzfuuzffxuz,$$

а терм $((x_1 + x_2) \times x_3)$ имеет вид: $\times + x_1x_2x_3$.

Польская запись обладает рядом полезных свойств.

Теорема 2.1. *(а). Если терм представлен в виде* $ft_1\ldots t_n$ *и в виде* $fs_1\ldots s_n$, *где* $t_1,\ldots, t_n$, $s_1,\ldots, s_n$ *— термы, а* f *—* n*-местный символ операции, то термы* $t_1,\ldots, t_n$ *совпадают с термами* $s_1,\ldots, s_n$.

(б). Никакое собственное начало терма не является термом.

(в). Каждый символ операции в записи терма является началом некоторого подтерма, единственного по (б).

Доказательство. (в) тривиально следует из (б). Будем оба утверждения (а) и (б) теоремы доказывать одновременно индукцией по длине терма.

Если ненульместных символов операций нет, то терм либо является переменной, либо нульместным символом операции. В этом случае теорема очевидна.

Пусть слова $ft_1\ldots t_n$ и $fs_1\ldots s_n$ совпадают. Пусть при этом $t_1,\ldots, t_i$ совпадают с $s_1,\ldots, s_i$, но t_{i+1} отличен от s_{i+1}. В этом случае либо t_{i+1} является собственным началом s_{i+1}, или s_{i+1} является собственным началом t_{i+1}. Это противоречит индукционному предположению, так как длина этих термов меньше, чем первоначального терма.

Пусть теперь терм s является собственным началом терма $ft_1\ldots t_n$. Тогда s имеет вид $fs_1\ldots s_n$, так как s начинается с f и f является n-местным. Повторяя рассуждение предыдущего абзаца, доказываем, что это невозможно. ∎

2.2 Подстановка и замена

Зафиксируем некоторую сигнатуру $\mathcal{L}$ и под термом будем понимать терм сигнатуры $\mathcal{L}$. Зафиксируем также некоторый список переменных $X = x_1, \ldots, x_n$. Будем предполагать, что переменные $x_1,\ldots, x_n$ попарно различны. Все рассматриваемые термы предполагаются термами над X.

Понятие одновременной подстановки в терм t термов $t_1,\ldots, t_n$ вместо переменных $x_1,\ldots, x_n$ было определено ранее. Напомним это определение.

Определение 2.3 (результата подстановки).

Пусть $x_1, \ldots, x_n$ — попарно различные переменные, а $t_1, \ldots, t_n$ — термы.

а). Если x отлична от каждой из переменных $x_1, \ldots, x_n$, то

$$(x)^{x_1,\ldots,x_n}_{t_1,\ldots,t_n}$$

есть x.

б). Если $i \in \{1, \ldots, n\}$, то

$$(x_i)^{x_1,\ldots,x_n}_{t_1,\ldots,t_n}$$

есть t_i.

в). Если f — нульместный символ операции, то $(f)^{x_1,\ldots,x_n}_{t_1,\ldots,t_n}$ есть f.

г). Если f — m-местный символ операции, а $t'_1, \ldots, t'_m$ — термы, то

$$(ft'_1 \ldots t'_m)^{x_1,\ldots,x_n}_{t_1,\ldots,t_n}$$

есть

$$f(t'_1)^{x_1,\ldots,x_n}_{t_1,\ldots,t_n} \ldots (t'_m)^{x_1,\ldots,x_n}_{t_1,\ldots,t_n}.$$

Операцию перехода от терма t к терму $(t)^{x_1,\ldots,x_n}_{t_1,\ldots,t_n}$ будем называть подстановкой в терм t термов $t_1,\ldots, t_n$ вместо переменных $x_1,\ldots, x_n$. Для этой подстановки σ через $\sigma(t)$ будем обозначать $(t)^{x_1,\ldots,x_n}_{t_1,\ldots,t_n}$. По определению, если f — m-местный символ операции, а $t'_1, \ldots, t'_m$ — термы, то

$$\sigma(ft'_1 \ldots t'_m) = f\sigma(t'_1) \ldots \sigma(t'_m).$$

Подстановка σ термов $t_1,\ldots, t_n$ вместо переменных $x_1,\ldots, x_n$ далее задаётся системой равенств

$$\sigma(x_1) = t_1, \ldots, \sigma(x_n) = t_n.$$

Подстановка σ называется тривиальной, если

$$\sigma(x_1) = x_1, \ldots, \sigma(x_n) = x_n.$$

Композицией подстановок τ и ρ будем называть такую подстановку σ, что

$$\sigma(t) = \rho(\tau(t))$$

для любого терма t. Композиция подстановок τ и ρ обозначается через $\rho\tau$.

Определение 2.4 (более общей подстановки).

Подстановка τ называется более общей, чем подстановка σ, если существует такая подстановка ρ, что $\sigma = \rho\tau$.

Пример 2.2. Пусть для арифметической сигнатуры

$$\sigma(x_1) = +x_1x_2,\ \sigma(x_2) = *x_1x_2.$$

Пусть $\tau(x_1) = +00,\ \tau(x_2) = *00$. Тогда подстановка σ является более общей, чем подстановка τ. Действительно, если $\rho(x_1) = 0,\ \rho(x_2) = 0$, то $\tau = \rho\sigma$.

Подтермом называется подслово терма, которое само является термом. Замена в терме фиксированного подтерма на некоторый терм определяется естественным образом. Более формально, заменив первый терм на второй, получим второй терм. Если же в терме $ft'_1 \ldots t'_m$ заменяем некоторый подтерм терма t'_i на другой терм и получаем при этом из t'_i терм t''_i, то из $ft'_1 \ldots t'_i \ldots t'_m$ при этой замене получим терм $ft'_1 \ldots t''_i \ldots t'_m$.

Легко доказывается индукцией по числу символов операций в терме следующая теорема.

Теорема 2.2. *Если в терме заменить некоторый фиксированный подтерм на терм, то получим терм.*

2.3 Унификация двух термов

Определение 2.5 (унификации).

Подстановка τ называется унификацией термов t и s, если $\tau(t) = \tau(s)$. Если для термов t и s существует унификация, эти термы называются унифицируемыми. Унификация термов t и s называется самой общей, если эта унификация является более общей, чем любая другая унификация термов t и s.

Пример 2.3. Рассмотрим термы $gxfyy$ и $gxfyx$, в которых оба символа операций бинарны. Подстановка σ, для которой $\sigma(x) = y,\ \sigma(y) = y$, унифицирует эти термы. Однако эти же термы могут быть унифицируемы и подстановками τ, для которой $\tau(x) = fxy,\ \tau(y) = fxy$, и ρ, для которой $\rho(x) = gxy,\ \rho(y) = gxy$. Легко однако заметить, что первая унификация является самой общей.

Следующий алгоритм позволяет найти самую общую унификацию двух термов t и s.

Алгоритм 1 (унификации).

Если эти термы одинаковы, их унифицирует тривиальная подстановка.

Рассмотрим первую слева позицию, различающую эти термы. Назовём эту позицию критической.

Если оба терма содержат в этой позиции символы операций, то эти термы неунифицируемы.

Чтобы термы t и s были унифицируемы, по крайней мере один из этих термов, скажем, t должен иметь в этой позиции символ предметной переменной x.

Если другой из этих термов содержит в этой позиции символ предметной переменной y, то рассмотрим подстановку y вместо x, не меняющую отличных от x переменных. Применяя эту подстановку к обоим термам, получим термы, которые уже не содержат x и для которых первая слева позиция, различающая полученные термы, продвинулась вправо.

Если другой из этих термов s содержит в критической позиции символ операции, то рассмотрим подтерм r терма s, начинающийся с этой критической позиции.

Если этот подтерм содержит x, то эти термы t и s неунифицируемы.

В противном случае рассмотрим подстановку r вместо x, не меняющую отличных от x переменных. Применяя эту подстановку к обоим термам t и s, получим термы, которые уже не содержат x и для которых первая слева позиция, различающая полученные термы, продвинулась вправо.

Так как после каждого шага работы алгоритма уменьшается число переменных, входящих хотя бы в один из унифицируемых термов, алгоритм закончит работу не более, чем через n шагов.

Пример 2.4. Пусть f, g и h — унарный и два бинарных символа операций. Рассмотрим термы

$$ghgxyfuz$$
$$gvhyhfwguu$$

этой сигнатуры и применим к ним алгоритм унификации.

На первом шаге критической является вторая позиция. Возникает подстановка $\sigma_1(v) = hgxyfu$. После этой подстановки получаем термы

$$ghgxyfuz$$
$$ghgxyfuhyhfwguu$$

и переходим к следующему шагу.

На втором шаге критической является восьмая позиция. Возникает подстановка $\sigma_2(z) = hyhfwguu$. После этой подстановки получаем термы

$$ghgxyfuhyhfwguu$$
$$ghgxyfuhyhfwguu$$

и заканчиваем работу, так как нет критических позиций.

В результате получаем одинаковые термы. Их унифицирует тривиальная подстановка. Таким образом, подстановка $\sigma_2\sigma_1$ унифицирует первоначальные термы.

Пример 2.5. Рассмотрим термы

$$hgxygxz$$
$$hgxzgyz$$

той же сигнатуры и применим к ним алгоритм унификации.

На первом шаге критической является четвёртая позиция. Возникает подстановка $\sigma_1(y) = z$. После этой подстановки получаем термы

$$hgxzgxz$$
$$hgxzgzz$$

и переходим к следующему шагу.

На втором шаге критической является шестая позиция. Возникает подстановка $\sigma_2(z) = x$. После этой подстановки получаем термы

$$hgxxgxx$$
$$hgxxgxx$$

и заканчиваем работу, так как нет критических позиций.

В результате получаем одинаковые термы. Их унифицирует тривиальная подстановка. Таким образом, подстановка $\sigma_2\sigma_1$ унифицирует первоначальные термы.

Пример 2.6. Если мы попробуем унифицировать термы

$$hxy$$
$$hxhyfhuv,$$

то придём к ситуации, когда y надо заменять на терм, содержащий y, что делать нельзя. Так что рассматриваемые термы не могут быть унифицированы.

Теорема 2.3. *Если после некоторого шага алгоритма унификации термы оказались одинаковыми, то композиция ρ всех построенных в ходе работы алгоритма унификации подстановок даёт самую общую унификацию первоначально рассматриваемых термов.*

Доказательство. Будем использовать индукцию по числу шагов в ходе работы алгоритма унификации.

Подстановка ρ_1, полученная после первого шага, приводит к термам, которые унифицируются алгоритмом унификации. Любая унификация σ первоначальных термов должна присваивать переменной, стоящей в критической позиции, и переменным, входящим в терм, начинающийся в этой критической позиции какие-то значения. После этого значение переменной, стоящей в этой критической позиции, и значение терма, начинающегося в этой критической позиции, должны совпадать. Следовательно, рассматриваемая унификация первоначальных термов является композицией подстановки ρ_1, полученной после первого шага унификации, и некоторой подстановки σ_1, унифицирующей термы, полученные после первого шага алгоритма унификации.

Так как композиция ρ_2 всех подстановок, построенных в ходе работы алгоритма унификации после первого шага, даёт самую общую унификацию термов, полученных после первого шага, это доказывает теорему.

Действительно, $\rho = \rho_2\rho_1$ и существует такая подстановка δ, что $\sigma_1 = \delta\rho_2$ и, значит,

$$\sigma = \sigma_1\rho_1 = \delta\rho_2\rho_1 = \delta\rho.$$

∎

Теорема 2.4. *Если после некоторого шага алгоритма унификации выдаётся сообщение, что унификация невозможна, то первоначальные термы неунифицируемы.*

Доказательство. По предыдущей теореме 2.3, достаточно доказать теорему 2.4 только для одного шага работы алгоритма унификации. Если сообщение о невозможности унификации выдаётся из-за того, что в критической позиции стоят разные символы операций, термы неунифицируемы очевидным образом. Осталось заметить, что унификация невозможна, если в критической позиции стоит переменная и начинается отличный от этой переменной терм, содержащий эту переменную. Это следует из того, что никакое собственное начало терма не является термом (теорема 2.1б)). ∎

2.4 Унификация нескольких термов

Для унификации нескольких термов надо сначала унифицировать первые два, потом результат их унификации и третий, потом результат этой унификации третьего и четвёртый и так далее.

Определение 2.6 (унификации нескольких термов).
Подстановка τ называется унификацией термов $t_1,\ldots,\ t_m$, если $\tau(t_1)=\cdots=\tau(t_m)$. Если для термов $t_1,\ldots,\ t_m$ существует унификация, эти термы называются унифицируемыми. Унификация термов $t_1,\ldots,\ t_m$ называется самой общей, если эта унификация является более общей, чем любая другая унификация термов $t_1,\ldots,\ t_m$.

Теорема 2.5. *Если термы $t_1,\ldots,\ t_m$ унифицируемы, то существует самая общая унификация этих термов.*

Доказательство. Индукцией по числу термов. Рассмотрим самую общую унификацию первых двух термов. Существует самая общая унификация полученного терма и оставшихся. Композиция этих самых общих унификаций и будет самой общей унификацией термов $t_1,\ldots,$ t_m. ∎

Определение 2.7 (унификации нескольких пар термов).
Подстановка τ называется унификацией пар термов $t_1,\ s_1,\ldots,$ $t_m,\ s_m$, если $\tau(t_i)=\tau(s_i)$ для $i=1,\ldots,m$. Если для пар термов $t_1,\ s_1,\ldots,\ t_m,\ s_m$ существует унификация, эти термы называются унифицируемыми. Унификация пар термов $t_1,\ s_1,\ldots,\ t_m,\ s_m$ называется самой общей, если эта унификация является более общей, чем любая другая унификация этих пар термов $t_1,\ s_1,\ldots,\ t_m,\ s_m$.

Теорема 2.6. *Если пары термы $t_1,\ s_1,\ldots,\ t_m,\ s_m$ унифицируемы, то существует самая общая унификация этих пар термов.*

Доказательство теоремы 2.6 аналогично доказательству теоремы 2.5 и предлагается читателю как упражнение.

3 Резолюция в логике предикатов

3.1 Эрбрановский универсум

Рассмотрим некоторую сигнатуру Ω и произвольную алгебраическую систему этой сигнатуры. Будем предполагать, что рассматриваемая сигнатура содержит нульместные символы операций. Напомним, что нульместные символы операций ещё называются константами или символами выделенных элементов. Понятно, что на множестве всех замкнутых термов сигнатуры Ω определены все основные операции сигнатуры Ω, если m-местному символу операции f поставлена в соответствие m-местная операция, которая для любых замкнутых термов $t_1,\ldots,\ t_m$ на наборе аргументов $t_1,\ldots,t_m$ принимает значение $ft_1\ldots t_m$. В частности, константе f в качестве значения этой константы сопоставлена сама эта константа f. Полученная алгебра называется **эрбрановским** универсумом сигнатуры Ω.

Эрбрановский универсум многими различными способами может быть расширен до алгебраической системы сигнатуры Ω. Для этого каждому символу отношения сигнатуры Ω надо поставить в соответствие некоторое отношение той же местности на множестве всех замкнутых термов сигнатуры Ω. Другими словами, для каждого m-местного символа отношения r и каждых замкнутых термов $t_1,\ldots,\ t_m$ выбрать $rt_1\ldots t_m$ или $\neg rt_1\ldots t_m$ в качестве истинной формулы.

Каждую алгебраическую систему, основное множество которой является эрбрановским универсумом, назовём **эрбрановской**.

Если в алгебраической системе сигнатуры Ω не рассматривать основные отношения, то полученная алгебра называется обеднением этой системы до алгебры.

Теорема 3.1. *Если замкнутая формула Φ сигнатуры Ω не содержит символа равенства и имеет модель $\mathfrak{A}$, порождённую значениями сигнатурных констант в $\mathfrak{A}$, то эта формула имеет эрбрановскую модель $\mathfrak{E}$.*

Доказательство. Можно предполагать, что рассматриваемая формула является предварённой. Сначала рассмотрим случай, когда формула не содержит кванторов, а потом используем индукцию по числу кванторов.

Рассмотрим гомоморфизм τ эрбрановского универсума сигнатуры Ω на обеднение модели $\mathfrak{A}$ до алгебры, при котором каждому замкнутому терму ставится в соответствие его значение в системе $\mathfrak{A}$.

Для каждого m-местного символа отношения r и каждых замкнутых термов $t_1,\ldots, t_m$ выбирем $rt_1\ldots t_m$ в качестве истинной формулы тогда и только тогда, когда $r\tau(t_1)\ldots\tau(t_m)$ истинна в системе $\mathfrak{A}$. Ясно, что при этом сохраняется истинность замкнутых бескванторных формул, не содержащих символа равенства. ∎

При проведении индукции надо использовать следующую лемму.

Лемма 3.2. *Если t — замкнутый терм сигнатуры Ω и Φ — предварённая формула сигнатуры Ω, то в алгебраической системе $\mathfrak{A}$ сигнатуры Ω на состоянии, которое x приписывает значение t, истинна формула Φ тогда и только тогда, когда в $\mathfrak{A}$ на этом состоянии истинна формула $(\Phi)^x_t$.*

Доказательство. Для бескванторных формул утверждение леммы очевидно. Используя индукцию по числу кванторов, можно предполагать, что Φ имеет вид $(\exists y)\Psi$. Остаётся заметить, что либо y отлична от x и $((\exists y)\Psi)^x_t$ есть $(\exists y)(\Psi)^x_t$, либо $((\exists y)\Psi)^x_t$ есть $(\exists y)\Psi$. ∎

Напомним, что формула называется **универсальной**, если эта формула является предварённой и её кванторная приставка не содержит кванторов существования. Далее мы рассматриваем только универсальные формулы.

Для универсальной формулы $(\forall y_1)\ldots(\forall y_m)\Psi$, где Ψ является бескванторной, **частным случаем** этой формулы называется $\rho(\Psi)$ для каждой подстановки ρ замкнутых термов $t_1,\ldots, t_m$, $s_1,\ldots, s_n$ вместо переменных $y_1,\ldots, y_m$, $z_1,\ldots, z_n$. Как обычно, $\rho(\Psi)$ есть

$$(\Psi)^{y_1,\ldots,y_m,\ z_1,\ldots,z_n}_{t_1,\ldots,t_m,\ s_1,\ldots,s_n}.$$

По предыдущему (лемма 3.2), истинность замкнутой универсальной формулы в эрбрановской алгебраической системе равносильна одновременной истинности в этой системе всех частных случаев этой формулы.

3.2 Клаузулы

Зафиксируем некоторую сигнатуру Ω. Будем предполагать, что рассматриваемая сигнатура содержит нульместные символы операций.

Литералом называется каждая атомная формула сигнатуры Ω, не содержащая символа равенства, и отрицание каждой такой формулы. Конечное множество литералов называется **клаузулой**. Клаузула называется замкнутой, если все входящие в неё литералы замкнуты. Для каждой подстановки ρ термов вместо переменных множество, составленное из $\rho(L)$ для всех входящих в клаузулу K литералов L, обозначается через $\rho(K)$ и называется **частным случаем** этой клаузулы. Частный случай клаузулы, состоящий из замкнутых литералов, называется замкнутым. Если M — множество клаузул и ρ — некоторая подстановка, то $\rho(M)$ состоит из $\rho(K)$ для всех клаузул K, входящих в M.

Определение 3.1 (совместности множества клаузул). *Множество клаузул истинно в некоторой эрбрановской алгебраической системе, если каждая клаузула из этого множества истинна в этой системе. Клаузула истинна в этой системе, если каждый замкнутый частный случай этой клаузулы истинен в этой системе. Частный случай клаузулы, состоящий из замкнутых литералов, истинен в этой системе, если хотя бы один из входящих в этот частный случай литералов истинен в этой системе. В частности, пустая клаузула ложна в каждой эрбрановской алгебраической системе.*

Множество клаузул совместно, если это множество истинно в некоторой эрбрановской алгебраической системе.

Тривиальным следствием этих определений является следующее замечание.

Замечание 3.1. *Если M — совместное множество клаузул и ρ — некоторая подстановка, то $M \cup \rho(M)$ является совместным множеством клаузул.*

Доказательство. Достаточно заметить, что частные случаи $\rho(K)$ являются частными случаями K. ∎

Рассмотрим теперь замкнутую универсальную формулу

$$(\forall y_1) \ldots (\forall y_m)\Psi,$$

где Ψ является бескванторной. Для формулы $\neg\Psi$ существует эквивалентная кнф. По этой причине формулу $\neg\Psi$ можно представлять как конечное множество клаузул.

Отрицание этой универсальной формулы эквивалентно экзистенциальной формуле, в которой кванторы существования можно удалить, заменяя в $\neg\Psi$ переменные $y_1, \ldots, y_m$ новыми константами. Получим после этого некоторое новое конечное множество замкнутых клаузул.

Если отрицание рассматриваемой универсальной формулы имеет модель, то, по теореме 3.1, это отрицание имеет эрбрановскую модель. Для доказательства отсутствия такой эрбрановской модели и, значит, тождественной истинности рассматриваемой универсальной формулы достаточно доказать несовместность полученного нового множества клаузул.

Более общий случай представляет импликация двух замкнутых универсальных формул. Если эта импликация не является тождественно истинной, то её отрицание имеет эрбрановскую модель. В этой модели посылка импликации истинна, а заключение ложно. Заменяя отрицание этого заключения на бескванторную формулу расширенной сигнатуры, получим множество клаузул, эквивалентное отрицанию рассматриваемой импликации. Для доказательства тождественной истинности рассматриваемой импликации достаточно доказать несовместность полученного множества клаузул.

Для использования этих идей на практике полезно следующее замечание о равенстве.

Замечание 3.2. *Равенство является отношением конгруэнтности.*

Другими словами, если конъюнкция формулы, полученной из заданной формулы заменой символа равенства на символ бинарного отношения e, не входящего в сигнатуру Ω, и формулы, утверждающей, что e является отношением конгруэнтности, не имеет модели, то не имеет модели и заданная формула.

Пример 3.1. Докажем, что в каждом группоиде каждый левый нуль равен каждому правому нулю и, значит, имеется только один двусторонний нуль. Это утверждение записывается формулой

$$(\forall x)(\forall y)((\forall z)(= fxzx \& = fzyy) \to = xy).$$

Если это утверждение ложно, то существует группоид и в нём такие элементы c и d, для которых выполняется

$$((\forall z)(= fczc \& = fzdd) \& \neg = cd).$$

Заменим теперь символ равенства на символ бинарного отношения e и добавим формулу, утверждающую, что e является отношением конгруэнтности. Полученную конъюнкцию преобразуем во множество клаузул. Получим

$$\{\{exx\},\ \{\neg exy,\ eyx\},\ \{\neg exy,\ \neg eyz,\ exz\},\ \{\neg eux,\ \neg evy,\ efuvfxy\},\\ \{efczc\},\ \{efzdd\},\ \{\neg ecd\}\}.$$

Осталось доказать несовместность полученного множества клаузул.

Пример 3.2. Докажем, что каждый группоид левых нулей является полугруппой. Это утверждение записывается формулой

$$((\forall x)(\forall y) = fxyx \to (\forall x)(\forall y)(\forall z) = ffxyzfxfyz).$$

Если это утверждение ложно, то существует группоид и в нём такие элементы b, c и d, для которых выполняется

$$((\forall x)(\forall y) = fxyx \& \neg = ffbcdfbfcd).$$

Заменим теперь символ равенства на символ бинарного отношения e и добавим формулу, утверждающую, что e является отношением конгруэнтности. Полученную конъюнкцию преобразуем во множество клаузул. Получим

$$\{\{exx\},\ \{\neg exy,\ eyx\},\ \{\neg exy,\ \neg eyz,\ exz\},\ \{\neg eux,\ \neg evy,\ efuvfxy\},\\ \{efxyx\},\ \{\neg effbcdfbfcd\}\}.$$

Осталось доказать несовместность полученного множества клаузул.

Итак, некоторые вопросы о тождественной истинности формул сводятся к доказательству несовместности некоторого множества клаузул.

Первый способ получения такого доказательства состоит в рассмотрении множества всех замкнутых частных случаев клаузул рассматриваемого множества и доказательстве несовместности полученного множества клаузул. Для этого доказательства достаточно заменить различные атомные формулы различными пропозициональными переменными и доказать несовместность полученного множества пропозициональных клаузул.

Действительно, если бы первоначальное множество частных случаев было истинным в некоторой эрбрановской системе, то присваивая каждой пропозициональной переменной то значение, которое имеет заменённая атомная формула, получим, что полученное множество пропозициональных клаузул тоже совместно.

По теореме компактности (теорема 1.3), если множество пропозициональных клаузул несовместно, то оно имеет конечную несовместную часть. Это означает, что нужно угадать некоторое конечное несовместное подмножество множества всех частных случаев клаузул рассматриваемого множества.

Второй способ состоит в использовании резолюции.

3.3 Резолюция

Определение 3.2 (резолюции). *Пусть M — некоторое множество клаузул. Пусть $K_1 = C_1 \cup \{L_1\}$, $K_2 = C_2 \cup \{L_2\}$ — клаузулы из M. Пусть, наконец, ρ — такая подстановка, что $\neg\rho(L_1) = \rho(L_2)$. Клаузула $\rho(C_1) \cup \rho(C_2)$ называется резольвентой клаузул K_1 и K_2. Говорят, что резольвента получается из исходных клаузул резолюцией.*

Теорема 3.3. *Пусть M — некоторое совместное множество клаузул. Пусть $K_1 = C_1 \cup \{L_1\}$, $K_2 = C_2 \cup \{L_2\}$ — клаузулы из M. Пусть ρ — такая подстановка, что $\neg\rho(L_1) = \rho(L_2)$. Если к M добавить резольвенту $\rho(C_1) \cup \rho(C_2)$ клаузул K_1 и K_2, то пополненное множество клаузул тоже совместно.*

Доказательство. В той эрбрановской системе, в которой истинны все клаузулы из M, истинна и клаузула $\rho(C_1) \cup \rho(C_2)$. Действительно, в этой системе истинны клаузулы $\rho(K_1)$ и $\rho(K_2)$ (доказательство замечания 3.1). Так как один из литералов $\rho(L_1)$, $\rho(L_2)$ ложен, то клаузула $\rho(C_1) \cup \rho(C_2)$ тоже истинна. ∎

Таким образом, для доказательства несовместности некоторого множества клаузул достаточно пополнять это множество резольвентами до тех пор, пока не получим пустую резольвенту.

3.4 Примеры

Пример 3.3. Докажем несовместность следующего множества клаузул:

$$\{\{exx\},\ \{\neg exy,\ eyx\},\ \{\neg exy,\ \neg eyz,\ exz\},\ \{\neg eux,\ \neg evy,\ efuvfxy\},\\ \{efczc\},\ \{efzdd\},\ \{\neg ecd\}\}.$$

Рассмотрим следующие частные случаи клаузул этого множества:

(1) $\{exx\}$,
(2) $\{\neg exy,\ eyx\}$,
(3) $\{\neg exy,\ \neg eyz,\ exz\}$,
(4) $\{\neg eux,\ \neg evy,\ efuvfxy\}$,
(5) $\{efcdc\}$,
(6) $\{efcdd\}$,
(7) $\{\neg ecd\}$.

Применяя подстановку $\rho(x) = fcd,\ \rho(y) = c$, получим $\{\neg efcdc,\ ecfcd\}$ как частный случай клаузулы (2). Применяя резолюцию к полученной клаузуле и (5), получим
(8) $\{ecfcd\}$.
Применяя подстановку $\rho(x) = c,\ \rho(y) = fcd,\ \rho(z) = d$, получим
(9) $\{\neg ecfcd,\ \neg efcdd,\ ecd\}$
как частный случай (3). Применяя резолюцию к (8) и (9), получим
(10) $\{\neg efcdd,\ ecd\}$.
Применяя резолюцию к (10) и (6), получим
(11) $\{ecd\}$.
Применяя резолюцию к (11) и (7), получим пустую клаузулу.

Пример 3.4. Докажем несовместность того же множества клаузул:

$$\{\{exx\},\ \{\neg exy,\ eyx\},\ \{\neg exy,\ \neg eyz,\ exz\},\ \{\neg eux,\ \neg evy,\ efuvfxy\},\\ \{efczc\},\ \{efzdd\},\ \{\neg ecd\}\},$$

но другим способом.

Рассмотрим следующее подмножество частных случаев клаузул этого множества:

{
(12) $\{\neg efcdc,\ ecfcd\}$,
(13) $\{\neg ecfcd,\ \neg efcdd,\ ecd\}$,
(5) $\{efcdc\}$,
(6) $\{efcdd\}$,
(7) $\{\neg ecd\}$
}.

Обозначая различные атомные формулы различными пропозициональными переменными, получим следующее множество пропозициональных клаузул:

$$\{\neg A,\ B\},\ \{\neg B,\ \neg C,\ E\},\ \{A\},\ \{C\},\ \{\neg E\}\}.$$

Доказать несовместность этого множества пропозициональных клаузул легко.