УДК 51:33(075.8)

Особенности логического вывода в ПРОЛОГ-программах

С. Ф. Тюрин

Пермский национальный исследовательский политехнический университет
Россия, 614990, г. Пермь, ул. Комсомольский пр., 29

tyurinsergfe@yandex.ru; +7-952-32-02-510

Пермский государственный национальный исследовательский университет
Россия, 614990, г. Пермь, ул. Букирева, 15

А. Ю. Городилов

Пермский государственный национальный исследовательский университет
Россия, 614990, г. Пермь, ул. Букирева, 15

gora830@yandex.ru; +7-902-83-07-328

Логический вывод может быть основан на методе резолюций, который предполагает доказательство от противного путем получения резольвент из исходного множества дизъюнктов. Если выводится пустой дизъюнкт (пустое множество), вывод считается правильным. Язык ПРОЛОГ, как язык логического программирования, базируется на первопорядковой логике предикатов. Однако имеются существенные отличия в процедурах вывода. В статье на конкретном примере рассматриваются особенности логического вывода, основанного на логике предикатов и логического вывода в программе на языке ПРОЛОГ.

Ключевые слова: логический вывод; резольвента; дизъюнкт; программа.

DOI: 10.17072/1993-0550-2019-3-91-97

Введение

Логический вывод, проверка правильности логических выводов является одной из основных процедур математической логики [1–4]. При этом вывод считается правильным, если из конъюнкции посылок следует заключение [1–4]. В логике высказываний, предикатов первого порядка используется метод резолюций. Доказательство ведется от противного до получения пустой резольвенты, обозначаемой пустым множеством, когда получают следствие из двух взаимно противоположных предложений (дизъюнктов) с использованием подстановок. В частном случае это могут быть атомарные предложения (формулы).

Необходимо предварительно получить формулы логики предикатов в стандартной бескванторной форме, кванторы существования исключаются с помощью введения функции Сколема. Получают конъюнктивную нормальную форму (КНФ) конъюнкции посылок и отрицания заключения.

Элементы КНФ это есть дизъюнкты или предложения, или клаузы (так и называют – *клаузная форма*). Вывод иногда представляют в форме дерева опровержения. Язык логического программирования ПРОЛОГ [5] использует метод резолюций, однако имеются особенности по сравнению с языком логики предикатов. Рассмотрим эти особенности на примере.

1. Вывод в логике предикатов

Пусть дано следующее: Тоня, Миша и Дима – члены альпинистского клуба. Каждый член этого клуба является либо горнолыжником, либо скалолазом или и тем, и другим. Все скалолазы не любят дождь. Все горнолыжники любят снег. Дима любит снег. Тоня любит снег и дождь. Миша любит все, что не любит Тоня, и не любит все, что любит Тоня. Требуется установить, есть ли альпинисты – скалолазы, не являющиеся горнолыжниками, и кто они (их имена).

Формализуем задачу в логике предикатов (рис. 1).

№	Формулировка	Формула логики предикатов	Примечание
1	спортсмен (миша)	$S(m)$	Факт-одноместный предикат (свойство)
2	спортсмен (тоня)	$S(t)$	Факт-одноместный предикат (свойство)
3	спортсмен (дима)	$S(d)$	Факт-одноместный предикат (свойство)
4	Каждый член этого клуба является либо горнолыжником, либо скалолазом или и тем и другим	$\forall x\{S(x) \rightarrow [P(x,g) \vee P(x,c) \vee P(x,g)P(x,c)]\} = \forall x\{S(x) \rightarrow [P(x,g) \vee P(x,c)]\}$	Правило $P(x,y)$ - быть (кто кем) г-горнолыжник с-скалолаз
5	Все скалолазы не любят дождь.	$\forall x[P(x,c) \rightarrow \bar{L}(x,r)]$	Правило $L(x,y)$ любить (кто что)
6	Все горнолыжники любят снег	$\forall x[P(x,g) \rightarrow L(x,c)]$	Правило
7	любит (дима, снег)	$L(d,s)$	Факт-двухместный предикат (бинарное отношение)
8	любит (тоня, снег)	$L(t,s)$	Факт-двухместный предикат (бинарное отношение)
9	любит (тоня, дождь)	$L(t,r)$	Факт-двухместный предикат (бинарное отношение)
10	любит (миша, X) :- любит (тоня, X)	$\bar{L}(t,x) \rightarrow L(m,x),$ $L(t,x) \rightarrow \bar{L}(m,x).$	Правило
11	есть (Z, скалолаз), но (есть (Z, горнолаз))?	$P(x,c)\bar{P}(x,g)$	Вопрос (цель)

Рис. 1. Формализация задачи о спортсменах

Построим аргумент [6]:

$$\frac{S(m), S(t), S(d), \forall x\{S(x) \rightarrow [P(x,g) \vee P(x,c)],$$

$$\forall x[P(x,c) \rightarrow \bar{L}(x,r)], \forall x[P(x,g) \rightarrow L(x,c)],$$

$$\forall x[\bar{L}(t,x) \leftrightarrow L(m,x)], L(d,s), L(t,s), L(t,r)}{\exists x[P(x,c)\bar{P}(x,g)]}. \quad (1)$$

Получим множество дизъюнктов:

$$S(m), S(t), S(d), \bar{S}(x) \vee P(x,g) \vee P(x,c),$$

$$\bar{P}(z,c) \vee \bar{L}(z,r), \bar{P}(y,g) \vee L(y,s),$$

$$L(t,w) \vee L(m,w), \bar{L}(t,w) \vee \bar{L}(m,w), \quad (2)$$

$$L(d,s), L(t,s), L(t,r),$$

$$\bar{P}(v,c) \vee P(v,g).$$

Построим дерево опровержения [6], используя метод резолюций (рис. 2):

Рис. 2. Дерево опровержения задачи о спортсменах

То есть, вывод правильный:

$$\exists x[P(x,c)\bar{P}(x,g)]. \quad (3)$$

Строим модифицированное дерево, добавляем к отрицанию заключения само заключение, чтобы определить этот x , который существует:

$$\bar{P}(v,c) \vee P(v,g) \vee P(v,c)\bar{P}(v,g). \quad (4)$$

и получаем модифицированное дерево (рис. 3):

Рис. 3. Модифицированное дерево опровержения задачи о спортсменах

В корне модифицированного дерева получили ответ

$$P(m,c)\bar{P}(m,g). \quad (5)$$

То есть ответ – Миша, (m), он скалолаз (c), но не горнолыжник (g). Так и есть. Миша не любит снег, Миша не любит дождь, поскольку согласно заданию "Миша...не любит все, что любит Тоня". Все отлично, все логично!

2. Вывод в языке ПРОЛОГ

Напишем программу, решающую задачу из приведенного выше примера, на языке ПРОЛОГ-Д [5]. Модель1 (1) не может быть использована полностью, поскольку выражение

$$\forall x\{S(x) \rightarrow [P(x,g) \vee P(x,c)] \quad (6)$$

не является хорновским дизъюнктом, а только таким, каким он (имеющие не более одного неинверсного литерала), ПРОЛОГ, допускает. Здесь два литерала, или говоря по-простому, два предиката без инверсии:

$$\bar{S}(x) \vee P(x,g) \vee P(x,c). \quad (7)$$

Рассмотрим другую модель, модель2, приведенную в [7] (рис. 4).

Рис. 4. Модель 2 работает

В этой модели использована только одна импликация, характеризующая пристрастия Миши (Миша любит все, что не любит Тоня, и не любит все, что любит Тоня):

(любит(миша,х): НЕ(любит(тоня,х))).

Кроме того, вместо "каждый член этого клуба является либо горнолыжником, либо скалолазом или и тем, и другим. Все скалолазы не любят дождь. Все горнолыжники любят снег" приведено два правила:

есть(х,горнолыжник):
-спортсмен(х),любит(х,снег).
есть(х,скалолаз):
-спортсмен(х),НЕ(любит(х,дождь)).

В результате получаем ответ: "Миша". Проверим эту модель (рис. 5) в логике предикатов.

3. Вывод в логике предикатов по модели, работающей в языке ПРОЛОГ

Построим новый аргумент по программе (рис. 4):

$$\begin{aligned} &S(m), S(t), S(d), L(d, s), \\ &L(t, s), L(t, r), \\ &\forall x [\bar{L}(t, x) \rightarrow L(m, x)], \\ &\forall y [S(y) L(y, s) \rightarrow P(y, g)], \\ &\forall z [S(z) \bar{L}(z, r) \rightarrow P(z, c)]. \\ &\hline \exists v [P(v, c) \bar{P}(v, g)] \end{aligned} \quad (8)$$

Получим множество дизъюнктов (стандартную бескванторную форму в КНФ), отрывающая заключение:

$$\begin{aligned} &S(m), S(t), S(d), L(d, s), \\ &L(t, s), L(t, r), \\ &L(t, x) \vee L(m, x), \\ &\bar{S}(y) \vee \bar{L}(y, s) \vee P(y, g), \\ &\bar{S}(z) \vee L(z, r) \vee P(z, c), \\ &\bar{P}(v, c) \vee P(v, g) \end{aligned} \quad (9)$$

Однако для (9) все попытки получения опровержения не приводят к успеху.

Например, первый дизъюнкт может дать резольвенту только с восьмым:

$$\frac{\bar{S}(y) \vee \bar{L}(y, s) \vee P(y, g), S(m)}{\bar{L}(m, s) \vee P(m, g)} \quad (10)$$

или с девятым:

$$\frac{\bar{S}(z) \vee L(z, r) \vee P(z, c), S(m)}{L(m, r) \vee P(m, c)} \quad (11)$$

Получить резольвенту из результатов (10), (11) невозможно – эти предложения не унифицируются – константы разные, в одной – дождь (г), в другой скалолаз (с). Аналогично можно получить и некоторые другие резольвенты, но опровержение вывода не достигается. Тем более что при ближайшем рассмотрении дизъюнкт

$$\bar{S}(z) \vee L(z, r) \vee P(z, c) \quad (12)$$

так же, как и (6), (7) оказывается не хорновским.

Тогда, вместо формулировки № 4 на рис. 1, попробуем, используя контрапозицию для изменения направления импликации, получить такой же, как (12) дизъюнкт с двумя положительными литералами:

$$\forall x \{ \bar{S}(x) \leftarrow [\bar{P}(x, g) \bar{P}(x, c)] \}. \quad (13)$$

Однако для такой модели1 ПРОЛОГ выдает результат "НЕТ" (рис. 5).

Рис. 5. Модель 1 с измененной формулировкой № 4 не работает

То есть, таких спортсменов нет.

4. Вывод в логике предикатов с использованием эквиваленции

Вернемся к модели 2. В статье [7] приведен, например, такой вывод для программы, изображенной на рис. 4:

есть(з, скалолаз), НЕ (есть(з, горнолыжник)),
есть(х, скалолаз):-спортсмен(х),
НЕ (любит(х, дождь))

спортсмен(х), НЕ (любит(х, дождь)),
НЕ (есть(х, горнолыжник)),

То есть заменяется "есть(з, скалолаз)" на "спортсмен(х), НЕ (любит(х, дождь))" с заменой переменной. Другими словами, фиксируется эквивалентность обеих частей импликации! Значит, вместо импликации выполняется эквиваленция. Таким образом, вместо (8) имеем:

$$\begin{array}{l} S(m), S(t), S(d), \\ L(d, s), L(t, s), L(t, r), \\ \forall x[\bar{L}(t, x) \leftrightarrow L(m, x)], \\ \forall y[S(y)L(y, s) \leftrightarrow P(y, g)], \\ \forall z[S(z)\bar{L}(z, r) \leftrightarrow P(z, c)] \\ \hline \exists v[P(v, c)\bar{P}(v, g)] \end{array} \quad (14)$$

Раскрываем импликацию, отрицая заключение и получая бескванторную форму:

$$\begin{array}{l} S(m), S(t), S(d), \\ L(d, s), L(t, s), L(t, r), \\ L(t, x) \vee L(m, x), \bar{L}(t, x) \vee \bar{L}(m, x), \\ \bar{S}(y) \vee \bar{L}(y, s) \vee P(y, g), \\ S(y)L(y, s) \vee \bar{P}(y, g), \\ \bar{S}(z) \vee L(z, r) \vee P(z, c), \\ S(z)\bar{L}(z, r) \vee \bar{P}(z, c), \\ \bar{P}(v, c) \vee P(v, g). \end{array} \quad (15)$$

Получим множество дизъюнктов и по нему построим дерево опровержения (рис. 6):

Рис. 6. Вывод в логике предикатов по модели 2 с эквиваленцией

Таким образом, с эквиваленцией все работает!

Аналогично можно построить и модифицированное дерево опровержения. То есть такая модель удовлетворяет требованиям и логике предикатов и ПРОЛОГу.

Но сопоставлять правила с эквиваленцией в общем случае неверно. Так, спортсмен, не любящий дождь, может быть, например, серфингистом, а не скалолазом.

5. Дальнейшие модификации программы на языке ПРОЛОГ

Попробуем модифицировать программу. Удалим "Миша любит все, что не любит Тоня, и не любит все, что любит Тоня" (рис. 7). Получаем Мишу! Как говорится, скрипач (в нашем случае – скрипачка) не нужен!

Рис.7. Модель 3 работает!

Таким образом, программа считает, что, если Миша не упомянут в "любящих", значит он "не любящий", что конечно, в общем случае опять не есть верно. Безусловно, такая модель № 3 тем более не корректна в логике предикатов (рис. 8).

Рис. 8. Попытка вывода в модели 3

Добавим факты о Мише, которые "домысливает" ПРОЛОГ (рис. 9).

Рис. 9. Попытка вывода в модифицированной модели 3 с дополнительными фактами, выделенными цветом

Получаем резольвенту $P(m,c)$, действительно, Миша-то скалолаз... Это следует из посылок. Но как вывести тот факт, что Миша не горнолыжник? Ясно, что из посылок нельзя вывести $P(m,g)$ аналогично, поскольку другое правило. Поэтому программа формирует НЕТ, то есть: $\bar{P}(m,g)$. Тогда и получаем (рис. 10).

Рис. 10. Вывод в модифицированной модели 3 с дополнительными фактами, выделенными цветом

Аналогично строится и модифицированное дерево, в корне которого будет ожидаемое выражение (5). Но для корректности рассуждений о Мише необходимо оставить в посылках дизъюнкт

$$\bar{L}(t,w) \vee \bar{L}(m,w), \quad (16)$$

при этом дизъюнкт

$$L(t,w) \vee L(m,w) \quad (17)$$

точно не нужен, поскольку Тоня любит и дождь, и снег.

Тогда корректно получаются Мишины факты.

6. Программа на языке ПРОЛОГ с учетом корректной базы знаний

Попробуем скорректировать базу знаний. Добавим правила, указывающие на то, кто является "не горнолыжником" и "не скалолазом". Информация о том, что любит и что не любит Миша, имеется в постановке задачи ("Миша любит все, что не любит Тоня, и не любит все, что любит Тоня"), но, поскольку есть только два варианта (любит–не любит), а Тоня только любит, оставим, как и раньше, только "не любит все, что любит Тоня".

Из вышеописанного мы знаем, что этого достаточно, чтобы определить отношение Миши к снегу и дождю. Полной информации о Диме у нас нет, то есть мы не знаем, любит ли он дождь, поэтому оставляем, как есть.

Получаем следующий аргумент:

$$\begin{aligned} & S(m), S(t), S(d), \\ & L(d,s), L(t,s), L(t,r), \\ & \forall x[\bar{L}(t,x) \leftarrow L(m,x)], \\ & \forall y[S(y)L(y,s) \rightarrow P(y,g)], \\ & \forall y[S(y)\bar{L}(y,s) \rightarrow \bar{P}(y,g)] \\ & \forall z[S(z)\bar{L}(z,r) \rightarrow P(z,c)] \\ & \frac{\forall z[S(z)L(z,r) \rightarrow \bar{P}(z,c)]}{\exists v[P(v,c)\bar{P}(v,g)]}. \end{aligned} \quad (18)$$

Из выражения (18) получаем множество дизъюнктов:

$$\begin{aligned} & S(m), S(t), S(d), \\ & L(d,s), L(t,s), L(t,r), \\ & \bar{L}(t,x) \vee \bar{L}(m,x), \\ & \bar{S}(y) \vee \bar{L}(y,s) \vee P(y,g), \\ & \bar{S}(y) \vee L(y,s) \vee \bar{P}(y,g), \\ & \bar{S}(z) \vee L(z,r) \vee P(z,c), \\ & \bar{S}(z) \vee \bar{L}(z,r) \vee \bar{P}(z,c), \\ & \bar{P}(v,c) \vee P(v,g). \end{aligned} \quad (19)$$

Видим, что (19) так же, как и (2) содержит как предикаты "есть", так и предикаты "не есть". Получим по (19) дерево опровержения (рис. 11):

Рис. 11. Дерево опровержения по модели 3

Опровержение достигается. Легко видеть, что модифицированное дерево опровержения, подобное приведенному на рис. 3, даст ответ "Миша". Таким образом, новая программа, дающая верный результат, будет выглядеть следующим образом (рис. 12):

Рис. 12. Программа по корректной модели 3

В этом случае и модель в логике предикатов и программа на языке ПРОЛОГ наконец-то дают одинаковый результат.

Заметим, что следствием дизъюнктов первой модели (2)

$$\begin{aligned} \bar{S}(x) \vee P(x, g) \vee P(x, c), \\ \bar{P}(z, c) \vee \bar{L}(z, r), \bar{P}(y, g) \vee L(y, s) \end{aligned} \quad (21)$$

являются дизъюнкты

$$\begin{aligned} \bar{S}(x) \vee P(x, g) \vee P(x, c), \\ \frac{\bar{P}(z, c) \vee \bar{L}(z, r)}{\bar{S}(x) \vee P(x, g) \vee \bar{L}(z, r)} \end{aligned} \quad (22)$$

$$\begin{aligned} \bar{S}(x) \vee P(x, g) \vee P(x, c), \\ \frac{\bar{P}(y, g) \vee L(y, s)}{\bar{S}(x) \vee P(x, c) \vee L(y, s)} \end{aligned} \quad (23)$$

из второй модели (9).

Однако в ней не хватает важных дизъюнктов

$$\bar{P}(z, c) \vee \bar{L}(z, r), \quad (23)$$

$$\bar{P}(y, g) \vee L(y, s), \quad (24)$$

фиксирующих "не есть". В то же время в третьей модели (19) дизъюнкты

$$\bar{S}(y) \vee L(y, s) \vee \bar{P}(y, g), \quad (25)$$

$$\bar{S}(z) \vee \bar{L}(z, r) \vee \bar{P}(z, c) \quad (26)$$

следуют из дизъюнктов (24), (23).

Выводы

Модель, задаваемая на языке ПРОЛОГ, должна быть корректной, иначе получаются неверные следствия, выводы. При наличии встроенного предиката НЕ, если вывод не завершается, ПРОЛОГ ответ НЕТ инвертирует и формируется ответ ДА. Нехорновский дизъюнкт (13) с тремя НЕ не работает в ПРОЛОГЕ, хотя сообщения об ошибке не формируется. Однако нехорновский дизъюнкт "есть(х, скалолаз):спортсмен(х), НЕ(любит(х, дождь))" работает в ПРОЛОГЕ.

Список литературы

1. Судоплатов С.В., Овчинникова Е.В. Математическая логика и теория алгоритмов: учебник. М.: ИНФРА-М4Новосибирск: Изд-во НГТУ, 2008. 224 с.
2. Морозенко В.В. Дискретная математика: учеб. пособие. Пермь, 2006. ISBN 5-7944-0608-9. 226. Библиогр. С. 223–224.
3. Лавров И.А. Математическая логика: учеб. пособие для вузов / под ред. Л.Л. Максимовой. М.: Академия, 2006. ISBN 5-7695-2735-8. 240. Библиогр.: 230 с.
4. Алябьева В.Г. Математическая логика. Пермь, 2017. ISBN 978-5-7944-2904-66. Библиогр.: с. 19–21, 110.
5. Основы программирования на языке Пролог. Курс лекций: учеб. пособие / П.А. Шрайнер. URL: <http://www.stepanoff.info/lisp/materials/prolog.pdf> (дата обращения: 13.04.2019).
6. Аляев Ю.А., Тюрин С.Ф. Дискретная математика и математическая логика. М.: Финансы и статистика, 2006. 357 с.
7. Залогова Л.А. Формальное описание логического вывода в ПРОЛОГЕ // Вестник Пермского университета. Математика. Механика. Информатика. 2014. № 4(27). С. 84–91.

Features of logical inference in PROLOG-programs

S. F. Tyurin

Perm National Research Polytechnic University; 29, Komsomolsky Av., Perm, 614990, Russia
Perm State University; 15, Bukireva st., Perm, 614990, Russia
tyurinsergfeo@yandex.ru; +7 952-320-02-510

A. Yu. Gorodilov

Perm State University; 15, Bukireva st., Perm, 614990, Russia
gora830@yandex.ru; +7-902-83-07-328

The logical conclusion is based on the resolution method, which assumes proof by contradiction by obtaining resolvents from the original set of clauses. If an empty clause is deduced (empty set), the conclusion is considered correct. The PROLOG language, as a logic programming language, is based on the first-order logic of predicates. However, there are significant differences in the withdrawal procedures. The article on a specific example discusses the features of logical inference, based on the logic of predicates and logical inference in a program in the language of PROLOG.

Keywords: *inference; resolvent; disjunct; program.*