PUBLICATIONS DE L'INSTITUT MATHÉMATIQUE
Nouvelle série, tome 33 (47), 1983, pp. 89–95

СПОСОБ ВСТРОЕНИЯ ПРАВИЛА ИНДУКЦИИ В ПРОЦЕДУРЫ АВТОМАТИЧЕСКОГО ДОКАЗАТЕЛЬСТВА ТЕОРЕМ С РЕЗОЛЮЦИЕЙ

Петар З. Хотомски

В настоящей работе осуществляется включение правила бинарной индукции, описанного в [**1**], в автоматические процедуры опровержения в теориях первого порядка. Приводится модифицированный алгоритм унификации для отыскания подстановки которая потребуется для примењенија правила бинарной индукции. Рассматривается алгоритм поиска опровержения с резолюцией и правилом бинарной индукции в теориях первого порядка с математической индукцией. Приводится сведения о программной системе и результатах отладки на эвм.

1. Правило бинарной индукции и алгоритм отыскания подстановки

Правило бинарной индукции, предложенное в [**1**], можно сформулировать следующим образом:

Из дизьюнктов $P_1 \vee C_1$ и $\neg P_2 \vee C_2$ (где P_1 и P_2 литеры, C_1 и C_2 дизьюнкты) не имеющих общих переменых и удовлетворяющих условию (i) существует подстановка σ дающая σ-примеры $P_{1\sigma}$ и $P_{2\sigma}$ вида $L_x(0)$ и $L_x(t)$, выводится дизьюнкты:

$$L_x\left(g(z_1, \dots, z_s)\right) \vee C_{1\sigma} \vee C_{2\sigma} \text{ и } \neg L_x\left(Sg(z_1, \dots, z_s)\right) \vee C_{1\sigma} \vee C_{2\sigma}$$

где g функция Сколема s аргументов; $z_1, \dots, z_s$ все различные переменные в литере $L_x(0)$; S – непосредственно следующий[1]

Если литера $L_x(0)$ не содержит переменных, то g являетая новой константой Сколема и правило можно записать в форме:

$$(P) \qquad P_1 \vee C_1,\ \neg P_2 \vee C_2 \underset{\text{при условии } (i)}{\vdash} L_x(g) \vee C_{1\sigma} \vee C_{2\sigma},\ \neg L_x(Sg) \vee C_{1\sigma} \vee C_{2\sigma}.$$

Подстановка σ определяется по особому алгоритму, который является расширением алгоритма унификации (об алгоритме унификации см. [**2**] или [**4**]).

[1] $A_x(t)$ формула полученная из формулы $A(x)$ замещением каждого свободного вхождения переменной x на терм t, который свободен для x в $A(x)$.

Блок схема алгоритма для отыскания подстановки σ представлена на фигуре 1. На виходе получается НОУ (найболе общий унификатор) если НОУ существует, а в противном на выходе получается подстановка σ, либо ответ что она не существует. Перерывистая линия на фигуре 1 приведена лишь для сравнения с алгоритмом унификации и не принадлежит алгоритму для отыскания подстановки σ. По приведенному алгоритму отыскивается найболе общая подстановка (НОП) когда подстановка существует (это следствие аналогичного свойства алгоритма унификации) и получается сведение о несуществовании НОП, когда подстановка не существует.

В приведенной схеме:

n — счетчик рассогласований которые не могут быть устранены унификацией. Когда на выходе $n = 0$, то σ является подстановкой обеспечивающей применение правила бинарной индукции.

k — счетчик всех рассогласований.

e — пустая подстановка.

$L_i\sigma_k$ — литера полученная применением подстановки σ_k к литере L_i.

$A_n\sigma_k$ — множество полученное из множества A_n применением подстановки σ_k к элементам (литерам) множества A_n.

$\sigma_{k+1}\theta$ — композиция подстановок σ_{k+1} и θ.

X — сохраняет первий по очереди терм U_k из литеры $L_i\sigma_k$, $i \in \{1, 2\}$, который не возможно унифицировать с отвечающим термом V_k литеры $L_j\sigma_k$, $j \in \{1, 2\}$ и $j \neq i$. Этот терм является кандидатом для применения правила бинарной индукции.

Работу алгоритма проследим на множестве:

$$A = \{P(0, h(x), x),\ \ P(f(y, z), w, f(0, w))\}.$$

$A_0 = A$. Множество рассогласования $B_0 = \{0, f(y, z)\}$, следственно НОУ для A не существует. Так как $V_0 = 0$ и $U_0 = f(y, z)$, то $\sigma_1 = \sigma_0 = e$, $X = f(y, z)$ и после устранения обноруженной разницы, для $n = 1$, будет $A_1 = \{P(0, h(x), x),\ P(0, w, f(0, w))\}$. Дальше, $B_1 = \{w, h(x)\}$. Это различие устранимо унификацией, поэтому $\sigma_2 = \{h(x)/w\}$ и для $k = 2$.

$$A_1\sigma_2 = \{P(0, h(x), x),\ \ P(0, h(x), f(0, h(x)))\}.$$

Теперь $B_2 = \{x, f(0, h(x))\}$, различие не устранимо унификацией и так как $V_2 = x$ входит в $U_2 = f(0, h(x))$ будет

$$\sigma_3 = \sigma_2\{0/x\} = \{h(0)/w, 0/x\}.$$

Так как $n = 1$ и оба терма $U_2 = f(0, h(x))$ и $X = f(y, z)$ происходят из одной и тойже литеры $P(f(y, z), w, f(0, w))$ ищется НОУ θ для $U_2\sigma_3 = f(0, h(0))$ и $X\sigma_3 = f(y, z)$.

По алгоритму унификации определяется $\theta = \{0/y, h(0)/z\}$ и новая подстановка σ_3 получается из $\sigma_3\theta$:

$$\sigma_3 = \{h(0)/w, 0/x, 0/y, h(0)/z\}.$$

Для $n = 2$, после устранения различия получается

$$A_2 = \{P(0, h(x), x), P(0, h(x), 0)\} \text{ и } A_2\sigma_3 = \{P(0, h(0), 0), P(0, h(0), 0)\} = \\ = \{P(0, h(0), 0)\}.$$

Множество $A_2\sigma_3$ одночленно, поэтому подстановка $\sigma = \sigma_3$ найдена. Действительно, применением σ к множеству A получается

$$A\sigma = \{P(0, h(0), 0), P(f(0, h(0)), h(0), f(0, h(0)))\}.$$

Так как

$$P(0, h(0), 0) = [P(X, h(0), X)]^{\cdot}_X(0) = L^{\cdot}_X(0)$$

и

$$P(f(0, h(0)), h(0), f(0, h(0))) = [P(X, h(0), X]^{\cdot}_X(f(0, h(0))) = L^{\cdot}_X(t)$$

для $t = f(0, h(0))$, к дизьюнктам

$$D_1 : P(0, h(x), x) \vee C_1 \text{ и } D_2 : \neg P(f(y, z), w, f(0, w)) \vee C_2$$

где C_1 и C_2 дизьюнкты, можно применить правилао бинарной индукции и вывести

$$P(g, h(0), g) \vee C_{1\sigma} \vee C_{2\sigma} \text{ и } \neg P(Sg, h(0), Sg) \vee C_{1\sigma} \vee C_{2\sigma}$$

где G – новая константа Сколема; $s = 0$ так как $L^{\cdot}_X(0)$ не содержит переменных.

2. Алгоритм поиска опровержения с резолюцией и правилом бинарной индукции

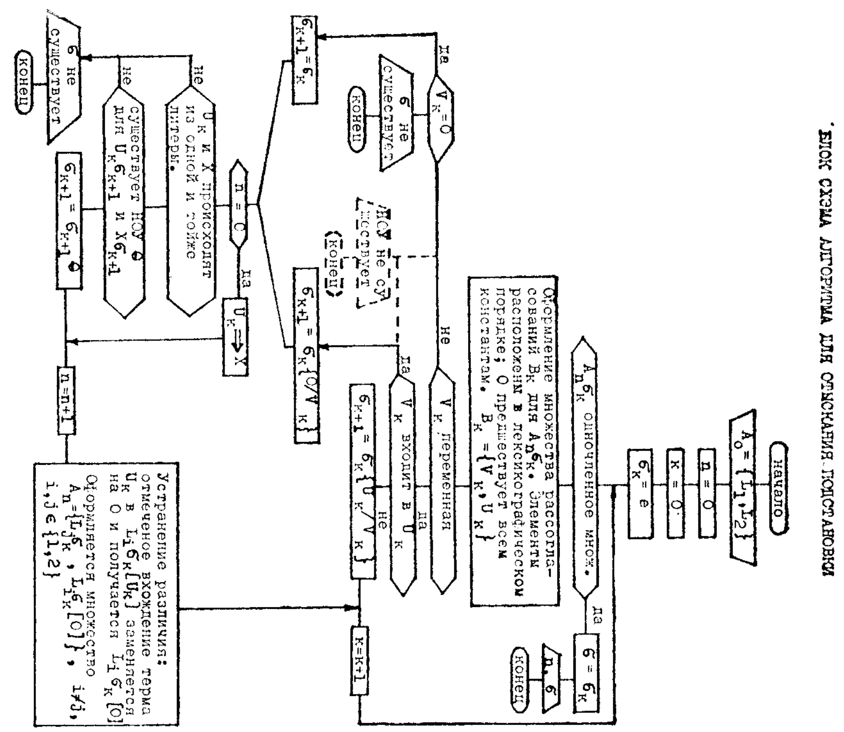
Встроение правила бинарной индукции в процедуры опровержения с резолюцией можно осуществить по следующему принципу: *Правило бинарной индукции применяется только если к данным дизьюнктным не применимо правило резолюции, а все условия для применения правила бинарной индукции выполнены.*

Этот принцип не зависит от конкретной формы резолюции (бинарная, хиперрезолюция, линейная резолюция и т. п.) и от конкретной стратегии поиска, поэтому применим в каждой из них.

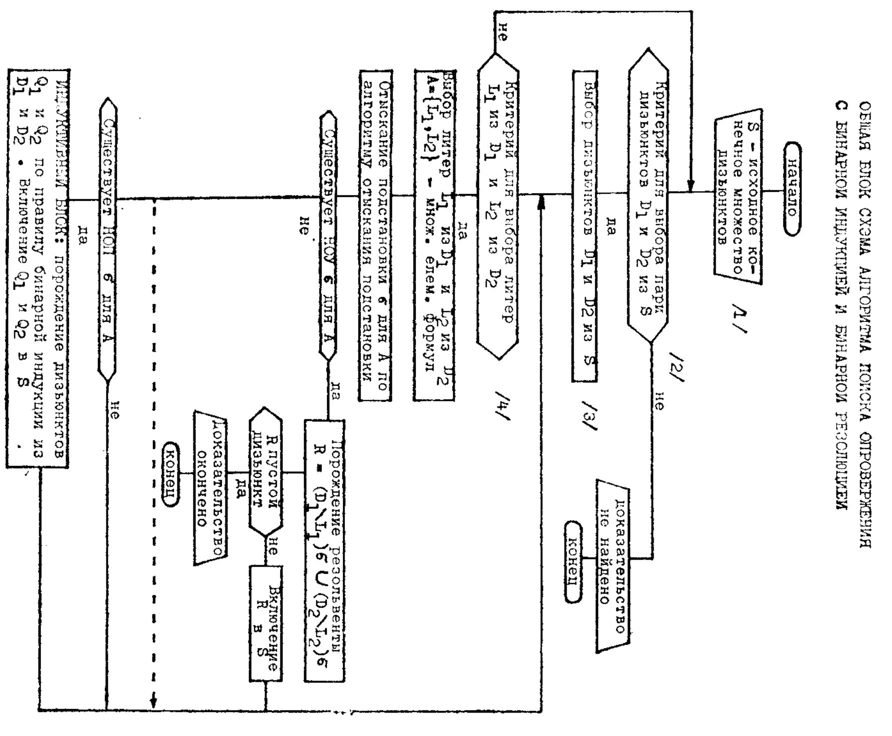
Общая блок схема алгоритма поиска опровержения обоснованного на резолюции трансформируется без нарушения основной структуры схемы с помощью следующих дополнений:

— прибавляется индуктивный блок котором по правилу бинарной индукции порождаются новые дизьюнкты, когда для обнаруженных литер не существует НОУ но существует НОП σ,

— прибавляется алгоритм отыскания НОП (фигура 1).



Фигура 1.



Фигура 2.

Общая блок схема алгоритма поиска опровержения с бинарной резолюцией и бинарной индукцией приведена на фигуре 2. Перерывистая линия на фигуре 2 приведена лишь для сравнения с алгоритмом который обоснован только на резолюции и не принадлежит алгоритму.

Коментарий к фигуре 2:

(1) Множество S содержит дизьюнкты происходящие из отрицания формулы которая подлежит доказательству, а если доказательство проводится в теории первого порядка, то в S входят и дизьюнкты происходящие из собственных аксиом (либо известных теорем) теории, кроме аксиом индукции.

(2), (3) и (4). Критерий зависит от ограничений на применение правила резолюции и от выбранной стратегии порождения дизьюнктов. Критерий включает и решение о перерыве поиска когда вычерпаны ресурсы времени или памяти машины, а также и другие условия для перерыва поиска когда доказательство не найдено.

Реализация индуктивного блока очевидна. Нужно только иметь в виду что при каждом новом применении правила бинарной индукции вводится новый символ функции (либо константы) Сколема. Этого можно добиться введением нумерации этих символов.

3. Сведения о программной системе и откладке на ЭВМ

Программная система доказательства теорем в теориях первого порядка с математической индукцией, обоснована на упорядоченной линейной резолюции (OL-резолюция, см. в [**2**]) и на правиле бинарной индукции. Упорядоченная линейная резолюция, включающая стратегию множества подержки и устранения тавтологий, выбрана благодаря преимуществу состоящем в необходимости запоминания только дизьюнктов порожденых на промежуточных — соседных уровнях поиска, а не всех порожденых дизьюнктов. Кроме того, один из дизьюнктов на каждом шагу эфиксирован в качестве "центрального" дизьюнкта, а резолюция (или индукция), без нарушения полноты, выполняется только для его последней литеры.

Так как использование правила бинарной индукции, когда резолюция не применима, приводит к порождению двух дизьюнктов, то один из них задерживается кандидатом на центральный дизьюнкт для следующего уровня, а другой записывается в ишодное множество "боковых" дизьюнктов.

Автором написана программная система состоящая из основной программы и последовательности 36 подпрограм. Эта система инкорпорирована в интерактивную систему "Graph" для классификации и развития знаний по теории графов, которая разрабатывается под руководительством проф. др. Драгоша Цветковича на Электротехническом факультете Белградского универститета. Об этой системе общие сведения можно найти в [**3**]. Программы написанны на ФОРТРАН-е и отлаженны на ЭВМ PDP 11/34.

Программная система доказательств теорем работает в двух режимах, использующих резолюцию и бинарную индукцию, либо только резолюцию. Входными данными являются дизьюнкты полученнте с помощью сколемизации из отрицания замкнутойформулы, подлежащей доказательству, и дизьюнкты полученные тем же способом из собстевенных аксиом теории первого порядка (с исключением схемы-аксиом математическиой индукции). В результате получается доказательство о невыполнимости ишодного множества дизьюнктов в виде отпечатанного опровержения, либо информация о невозможности отыскать опровержение в предназначенных размерах машинной памяти.

Признаком перерыва поиска является достижение предъявленного максимального числа всех порожденных дизьюнктов. В поиске используются и ограничения на длину литер, длину дизьюнктов и количество дизьюнктов порожденых на каждом уровне. Дизьюнкты превосходящие эти ограничения не порождаются.

Для илюстрации приводим один из простых примеров реализованых на ЭВМ PDP 11/34 в процессе откладки.

Пример: Доказательство $\forall x P(x)$ опровержением из $P(O)$ и $\forall x(P(x) \Rightarrow P(Sx))P(x)$ литера в которойx единственная переменная). Входными данными являются следующие дизьюнкты:

$\neg P(C)$ отрицание и скол. формулы $\forall x P(x)$, C-конст. Скол. $P(O)$
$\neg P(X) \vee P(S(X))$.

На ЭВМ PDP 11/34 получено следующее опровержение (напечатанный документ приводим в переводе на русский язык):

ДОКАЗАТЕЬСТВО НАЙДЕНО

ОПРОВЕРЖЕНИЕ СОСТОИТ ИЗ СЛЕДУЮЩЕЙ ПОСЛЕДОВАТЕЛЬНОСТИ:

ЦЕНТРАЛЬНЫЙ ДИЗЬЮНКТ:

$\neg P(C)^*$

БОКОВОЙ ДИЗЬЮНКТ:

$P(0)^*$

ПОДСТАНОВКА ДЛЯ ИНДУКЦИИ:

* (пустая)

ЦЕНТРАЛЬНЫЙ ДИЗЬЮНКТ:

$\neg P(S(G1))^*$

БОКОВОЙ ДИЗЬЮНКТ:

$\neg P(X)P(S(X))^*$

ПОДСТАНОВКА ДЛЯ РЕЗОЛЮЦИИ (НОУ):

$G1/X^*$

ЦЕНТРАЛЬНЫЙ ДИЗЬЮНКТ:

$/\neg P(S(G1))\neg P(G1)^*$

БОКОВОЙ ДИЗЬЮНКТ:

$P(G1)^*$

ПОДСТАНОВКА ДЛЯ РЕЗОЛЮЦИИ (НОУ):

* (пустая)

ЦЕНТРАЛЬНЫЙ ДИЗЬЮНКТ:

ПУСТОЙ ДИЗЬЮНКТ:

ДОКАЗАТЕЬСТВО ОТПЕЧАТАНО

ДОКАЗАНА НЕВЫПОЛНИМОСТЬ ИСХОДНОГО МНОЖЕСТВА

Пояснения: Каждый следующий центральный дизьюнкт является результатом применения правила индукции или резулюции к предшествующим центральному и боковому дизьюнктам. Символ $*$ означает конец последовательности, определяющей дизьюнкт или подстановку. В процессе опреверження символ $\vee$ не используется, поэтому дизьюнкт представлен последовательностью литер. Символ / перед литерой указывает что литера стоящая за ним маркированна.

Простота отлаженных примеров обусловлена малым объемом оперативной памяти используемой ЭВМ.

В рамках системы "Graph" на ЭВМ большей мощности можно ожидать доказательства более сложных теорем.

Автор выражает глубокую благодарность проф. др Драгошу Цветковичу за помощ и подержку указанную при оформлении и отладке программной системы.

ЛИТЕРАТУРА

[1] П. Хотомски, *Правило индукции в доказательствах опроверженнем с применением к автоматическому доказательству теорем*, Publ. Inst. Math. (Beograd) (N.S.) **31 (45)** (1982).

[2] Э. В. Попов, Г. Р. Фирдман, *Алгоритмические основы интеллектуальных роботов и искусственного интеллекта*, Наука, Москва, 1076.

[3] D. Cvetkoviffl, *A projekt for using Computers in further development of Graph Theory*, Proc. of the 4-th Internat. Conf. on the Theory and Appl. of Graphs, Kalamazoo 1980, Wiley, 1981.

[4] J. A. Robinson, *A Machine-Oriented Logic Based on the Resolution Principle*, J. ACM, **12** (1965) 23–41.

[5] Н. Нильсон, *Искуственный интеллект*, Мир, Москва, 1973.

[6] Э. Мендельсон, *Введение в математическую логику*, Наука, Москва, 1976.

Педагошка академиjа
22000 Сремска Митровица
Jугославиjа

(Поступила 10 05 1982)