Алгоритм поиска модели произвольного конечного множества дизъюнкций

С.Б. Прешич

В работе показано, как PL-алгоритм [1], [2], [3] можно применить к проблеме поиска модели данного конечного множества дизъюнкций.

Пусть F - произвольное множество пропозициональных дизъюнкций $A_1 \vee ... \vee A_n$, где каждое A_i - пропозициональная буква, ее отрицание, символ $\top$ ("истина") или символ $\perp$ ("ложь"). Наша задача следующая:
(1) Найти хотя бы одну модель (если она существует) данного конечного множества дизъюнкций F.

Пусть p - некоторая пропозициональная буква. Множество F будем обозначать через $F(p)$. В PL-алгоритме основную роль играют следующие логические эквиваленции[1]

$$(2) \quad (i) \quad F(p) \vdash p \Longleftrightarrow F(\perp) \vdash \perp, \qquad (ii) \quad F(p) \vdash \neg p \Longleftrightarrow F(\top) \vdash \perp$$

$$(3)\ \mathcal{F}, \quad A_1 \vee A_2 \vee ... \vee A_k \vdash \perp \quad \Longleftrightarrow \quad \mathcal{F} \vdash \neg A_1 \textbf{ и } \mathcal{F} \vdash \neg A_2 \textbf{ и}...\textbf{и } \mathcal{F} \vdash A_k;$$

$\mathcal{F}$ - множество формул, A_i - формулы.

$$(4) \qquad \mathcal{F}, \ \perp \vdash \perp \quad \Longleftrightarrow \quad \vdash \top,$$

то есть справедлива любая секвенция вида $\mathcal{F}, \ \perp \vdash \perp$.

Через PLdis обозначим алгоритм, с помощью которого будем решать вопрос вида (1). Основная идея этого алгоритма состоит в применении следующего факта:

[1] Доказательства этих эквиваленций см. в [1, Леммы 1, 2]

Множество F имеет модель тогда и только тогда, когда секвенция $F \vdash \bot$ ложна.

PLdis-алгоритм включает в себя PL-алгоритм, а также следующее правило

(Δ) При применении (2) к секвенции вида $F \vdash p$ или $F \vdash \neg p$, в случае,

когда не получаем секвенцию вида G, $\bot\ \vdash \bot$, где G – некоторое множество дизъюнкций, сделаем следующее:

Запомним значение буквы p. В случае (i) – это $\bot$, а в случае (ii) – это $\top$. В случае, если p уже получило некоторое значение, старое значение заменим новым.

Теперь коротко опишем PLdis-алгоритм:

На начальном шаге рассмотрим секвенцию $F \vdash \bot$, возьмем одну дизъюнкцию из F и применим (3). После этого воспользуемся PL-алгоритмом, т.е. эквиваленциями вида (2), (3), (4), и всегда при применении эквиваленции вида (2) применяем также правило (Δ).

В результате мы можем получить одну из секвенций: 1) $\vdash \top$, 2) $\vdash \bot$. В первом случае заключаем, что начальное множество F не имеет модели, а во втором (что мы докажем позже):

(5) *Множество F имеет модель, определенную значениями пропозициональных букв, полученными в течение работы алгоритма.*

Теперь приведем несколько примеров.

Пример 1. *Найти хотя бы одну модель множества дизъюнкций* $a, \neg a$.

Решение. Имеем следующую цепь эквиваленций

$a, \neg a\ \vdash\ \bot$

$\Longleftrightarrow\ \neg a\ \vdash\ \neg a$ (Взяли дизъюнкцию a и применили (3))

$\Longleftrightarrow\ \bot\ \vdash\ \bot$ (По (2))

$\Longleftrightarrow \vdash\ \top$ (По (4))

Заключение: множество $\{a, \neg a\}$ не имеет модели.

Пример 2. *Найти хотя бы одну модель данного множества F:*

1) a, 2) $a \vee b$, 3) $a \vee \neg b,\ \neg b \vee c,\ \neg c,\ \neg a \vee b$

Решение. 1). Имеем следующую цепь эквиваленций

$a \vdash \bot \iff \vdash \neg a$ (по (3))

$\iff \vdash \bot$ (по (2) (ii), a получило значение $\top$)

Заключение: в силу (5) множество F имеет модель, в которой a имеет значение $\top$, что будем записывать также следующим образом: $\tau(a) = \top$. Вообще, впредь через τ будем обозначать искомую модель.

2) Имеем следующую цепь эквиваленций

$a \vee b \vdash \bot \iff \vdash \neg a$ и $\vdash \neg b$ (по (3))

Получили конъюнкцию A_1 **и** A_2 двух секвенций $A_1 : \vdash \neg a$, $A_2 : \vdash \neg b$. Теперь будем "вычислять" A_1. Имеем следующую цепь эквиваленций

$A_1 \iff \vdash \bot$ (по (2), a получило значение $\top$, т.е. $\tau(a) = \top$.)

Значит, секвенция A_1 ложна. Следовательно, конъюнкция A_1 **и** A_2 также ложна. Следовательно, и начальная секвенция ложна.[2]

Заключение: секвенция $a \vee b \vdash \bot$ ложна при $\tau(a) = \top$. Иными словами, одна модель τ определена: $\tau(a) = \top$, $\tau(b)$ – произвольное. В самом деле:

В силу эквиваленции $F \vdash \bot \iff A_1 \iff \vdash \bot$ (при $\tau(a) = \top$) получаем эквиваленцию $F \vdash \bot \iff \vdash \bot$ при $\tau(a) = \top$, что означает, что условием $\tau(a) = \top$ определена одна модель для F.

3) Имеем следующую цепь эквиваленций

$a \vee \neg b,\ \neg b \vee c,\ \neg c,\ \neg a \vee b \vdash \bot$

$\iff \neg b \vee c,\ \neg c,\ \neg a \vee b \vdash \neg a$ **и** $\neg b \vee c,\ \neg c,\ \neg a \vee b \vdash b$

$\iff b,\ \neg b \vee c,\ \neg c \vdash \bot$ (по (2), $\tau(a) = \top$) **и** $\neg a, \neg c \vdash \bot$ (по (2), $\tau(b) = \bot$)

$\iff A_1 \vdash \bot$ **и** $A_2 \vdash \bot$,

где A_1 есть $b,\ \neg b \vee c,\ \neg c$ и A_2 есть $\neg a, \neg c$

Теперь "вычисляем" $A_1 \vdash \bot$:

$A_1 \vdash \bot \iff \neg b \vee c,\ \neg c \vdash \neg b$ (по (3))

$\iff c,\ \neg c \vdash \bot$ (по (2), $\tau(b) = \top$)

$\iff \neg c \vdash \neg c$ (по (3))

$\iff \bot \vdash \bot$ (по (2))

$\iff \vdash \top$

[2] Вообще, если начальная секвенция $F \vdash \bot$ эквивалентна некоторой конъюнкции A_1 **и** A_2 **и** ... **и** A_k, где A_1 – ложная секвенция, то секвенция $F \vdash \bot$ тоже ложна, и алгоритм завершает работу.

Следовательно, секвенция $A_1 \vdash \perp$ истинна. По этой причине мы должны "вычислить" $A_2 \vdash \perp$ тоже. Имеем следующую цепь эквиваленций

$A_2 \vdash \perp \iff \neg c \vdash a$ (по (3))

$\iff \neg c \vdash \perp$ (по (2), $\tau(a) = \perp$. Это новое значение для a)

$\iff \vdash c$ (по (3))

$\iff \vdash \perp$ (по (2), $\tau(c) = \perp$)

Для a, b, c получили значения: $\tau(a) = \perp, \tau(b) = \perp, \tau(c) = \perp$. Легко проверить, что тем самым мы получили одну модель для F. Это утверждение согласуется с (5), но (5) мы еще не доказали. Приведем теперь это доказательство:

Доказательство утверждения (5). Предположим, что для множества F посредством PLdis-алгоритма доказано, что секвенция $F \vdash \perp$ ложна, т.е. что верна эквиваленция $F \vdash \perp \iff \vdash \perp$. Обозначим через n число всех пропозициональных букв, входящих в элементы множества F. Доказательство будет индукцией по n. Если $n = 1$, то секвенция $F \vdash \perp$ должна быть вида $p \vdash \perp$ или $\neg p \vdash \perp$, и тогда (5), очевидно, верно.

Пусть $n > 1$. Будем использовать следующее обозначение: p^α, где $\alpha \in \{\top, \perp\}$ означает p, если $\alpha = \top$ и $\neg p$, если $\alpha = \perp$ соответственно. Следуя PLdis-алгоритму, на первом шаге в множестве F выбираем некоторую дизъюнкцию. В связи с этим предположим, что F имеет следующий вид: $G(p_1, ..., p_k),\ p_1^{\alpha_1} \vee ... \vee p_k^{\alpha_k}$ (это будем обозначать также $F(p_1, ..., p_k)$). Тогда, применяя (3), получим эквиваленцию

$F(p_1, ..., p_k) \vdash \perp \iff G(p_1, ..., p_k) \vdash p_1^{\neg\alpha_1}$ **и** ... **и** $G(p_1, ..., p_k) \vdash p_k^{\neg\alpha_k}$.

Поскольку $F \vdash \perp$ – ложная секвенция, то по PLdis-алгоритму получим (первую) секвенцию вида $G(p_1, ..., p_k) \vdash p_i^{\neg\alpha_i}$, которая ложна. Следовательно, имеем эквиваленцию

$$F(p_1, ..., p_k) \vdash \perp \iff G(p_1, ..., p_k) \vdash p_i^{\neg\alpha_i}.$$

На следующем шаге применим (2) и получим

$$F(p_1, ..., p_k) \vdash \perp \iff G(p_1, ..., p_{i-1}, \alpha_i, p_{i+1}, ..., p_k) \vdash \perp,$$

и тогда p_i получит значение α_i, которое в дальнейшем по ходу выполнения алгоритма не может быть изменено (ибо p_i не входит в $G(p_1, ..., p_{i-1}, \alpha_i, p_{i+1}, ..., p_k)$).

Теперь применим индукционное предположение к правой части этой эквиваленции. Через $\bar{F}$, $\bar{G}$ обозначим конъюнкцию всех формул в F и G соответственно. Тогда существует τ - значение всех пропозициональных букв из F, кроме p_i, так что равенство

$$(*) \qquad \tau(\bar{G}(\tau(p_1), ..., \tau(p_{i-1}), \alpha, \tau(p_{i+1}), ..., \tau(p_k)) = \top$$

выполнено. Но, с другой стороны поскольку $F(p_1, ..., p_k)$ на самом деле есть $G(p_1, ..., p_k)$, $p_1^{\alpha_1} \vee ... \vee p_k^{\alpha_k}$, то формулы $\bar{F}(p_1, ..., p_{i-1}, \alpha_i, p_{i+1}, ..., p_k))$, $\bar{G}(p_1, ..., p_{i-1}, \alpha_i, p_{i+1}, ..., p_k))$ логически эквивалентны. Следовательно, по $(*)$ получим равенство

$$\tau(\bar{F}(\tau(p_1), ..., \tau(p_{i-1}), \alpha, \tau(p_{i+1}), ..., \tau(p_k)) = \top,$$

что означает, что τ, дополненное соотношением $\tau(p_i) = \top$, дает искомую для множества F. Доказательство закончено.

Список литературы

[1] S. B. Prešić *How to generalize logic programming to arbitrary set of clauses*, Publ. de l' Inst. Math. NS 61(75), Belgrade, 1997, p 137-152

[2] S. B. Prešić *Generalizing logic programming to arbitrary set of clauses*, Sci. Rev. Belgrade, 19/29, 1996, p 75-81

[3] С. Б . Прешич *Об обобщении логического программирования на произвольное множество дизъюнкций.* [см. наст. том, стр. 277]