Metoda rezolucji

Metoda rezolucji (Robinson, 1965) jest pełną (i stosunkowo efektywną) procedurą dowodzenia twierdzeń w logice pierwszego rzędu.

Problem dowodzenia twierdzeń: Dla danej bazy wiedzy KB i danej formuły α stwierdzić, czy $KB \models \alpha$. (Problem półrozstrzygalny.)

Problem dowodzenia twierdzeń jest równoważny **problemowi spełnialności**: Dla danej bazy wiedzy KB i danej formuły α stwierdzić, czy zbiór $KB \cup \{\neg\alpha\}$ jest niespełnialny.

Ponieważ bazy wiedzy są skończone, $KB \cup \{\neg\alpha\}$ można zawsze traktować jako pojedynczą formułę.

Metoda rezolucji dla danej formuły β próbuje stwierdzić, czy β jest niespełnialna.

Metodę rezolucji stosuje się dla formuł w postaci **klauzulowej**.

Klauzula – formuła postaci

$$l_1 \vee \cdots \vee l_n \quad (n \geq 0) \tag{1}$$

gdzie $l_1, \ldots, l_n$ są literałami. Jeśli $n = 0$, klauzulę (1) nazywamy **klauzulą pustą** i oznaczamy symbolem $\bot$.

Postać klauzulowa:

$$\forall\, \overline{x_1}\, C_1 \wedge \cdots \wedge \forall\, \overline{x_m}\, C_m \quad (m \geq 1)$$

gdzie $C_1, \ldots, C_n$ są klauzulami, natomiast $\overline{x_i}$ jest listą wszystkich zmiennych występujących w C_i.

Twierdzenie

Istnieje algorytm, który dla dowolnej formuły A znajduje formułę B taką, że:

(1) B jest w postaci klauzulowej.

(2) A jest spełnialna wtw gdy B jest spełnialna. (A i B nie muszą być równoważne.)

Sprowadzanie formuły A do postaci klauzulowej

1. Niech $\overline{x}$ będzie listą wszystkich zmiennych wolnych występujących w A. Zastępujemy A formułą $\exists\,\overline{x}\;A$.

2. Eliminujemy wszystkie redundantne kwantyfikatory, tzn. każdy kwantyfikator postaci $\forall x$ lub $\exists x$ taki, że zmienna x nie występuje w obrębie działania tego kwantyfikatora.

3. Przemianowujemy zmienne tak, aby wszystkie kwantyfikowane zmienne były różne.

4. Eliminujemy "$\Rightarrow$" i "$\Leftrightarrow$", zastępując $B \Rightarrow C$ przez $\neg B \vee C$ i $B \Leftrightarrow C$ przez $((\neg B \vee C) \wedge (\neg C \vee B))$.

5. Przesuwamy "$\neg$" w prawo, zastępując $\neg\forall x B$ przez $\exists x\ \neg B$, $\neg\exists x\ B$ przez $\forall x\ \neg B$, $\neg(B \vee C)$ przez $\neg B \wedge \neg C$, $\neg(B \wedge C)$ przez $\neg B \vee \neg C$, $\neg\neg B$ przez B, tak długo aż wszystkie symbole negacji występują bezpośrednio przed formułami atomowymi.

6. (Krok opcjonalny. Poniżej Q oznacza $\forall$ lub $\exists$; $\diamondsuit$ oznacza $\wedge$ lub $\vee$.)
   Przesuwamy kwantyfikatory w prawo, zastępując $Qx(B \diamondsuit C)$ przez $B \diamondsuit (QxC)$, jeśli x nie jest wolna w B, i przez $(QxB) \diamondsuit C$, jeśli x nie jest wolna w C.

7. Skolemizacja (Usunięcie kwantyfikatorów egzystencjalnych). Powtarzamy poniższy proces, aż wszystkie kwantyfikatory $\exists$ zostaną usunięte.

   Bierzemy dowolną podformułę postaci $\exists y\ B(y)$. Niech $x_1, \ldots, x_n$ ($n \geq 0$) będzie listą wszystkich zmiennych wolnych występujących w $\exists y\ B(y)$, które są uniwersalnie kwantyfikowane w pewnej nadformule formuły $\exists y\ B(y)$. Jeśli $n = 0$, zastępujemy $\exists y\ B(y)$ przez $B(A)$, gdzie A jest **nową** stałą indywiduową. Jeśli $n > 0$, zastępujemy $\exists y\ B(y)$ przez $B(F(x_1, \ldots, x_n))$, gdzie F jest **nowym** symbolem funkcji n-argumentowej.

8. Usuwamy wszystkie kwantyfikatory uniwersalne.

9. Rozdzielamy $\wedge$ względem $\vee$, zastępując $(B \wedge C) \vee D$ przez $(B \vee D) \wedge (C \vee D)$ oraz $B \vee (C \wedge D)$ przez $(B \vee C) \wedge (B \vee D)$.

10. Eliminujemy stałe $True$ i $False$. Aktualna formuła jest koniunkcją klauzul. Usuwamy każdą klauzulę zawierającą literał $True$. Z pozostałych klauzul usuwamy każdy literał $False$.

11. Dodajemy kwantyfikatory uniwersalne. Aktualna formuła jest postaci $C_1 \wedge \cdots \wedge C_k$, gdzie $C_1, \ldots, C_k$ są klauzulami. Zastępujemy ją formułą

$$\forall\, \overline{x_1}\, C_1 \wedge \cdots \wedge \forall\, \overline{x_k}\, C_k$$

gdzie $\overline{x_i}$ jest listą wszystkich zmiennych występujących w C_i.

Uwaga: Kroki 1 i 7 nie zachowują równoważności. (Zachowują spełnialność.)

Formułę w postaci klauzulowej

$$\forall\, \overline{x_1}\; C_1 \wedge \cdots \wedge \forall\, \overline{x_m}\; C_m$$

możemy zapisać jako zbiór klauzul, tzn. w postaci $\{C_1, \ldots, C_m\}$. Możemy też zawsze opuścić kwantyfikatory.

Literał dodatni – formuła atomowa.

Literał ujemny – negacja formuły atomowej.

$| \, l \, |$ oznacza formułę atomową literału l.

Piszemy $l_1, \ldots, l_n \in C$, aby wyrazić fakt, że literały $l_1, \ldots, l_n$ występują w klauzuli C. Jeśli $l_1, \ldots, l_n \in C$, to $C - \{l_1, \ldots, l_n\}$ oznacza klauzulę otrzymaną z C w wyniku usunięcia $l_1, \ldots, l_n$.

Podstawienie: Zbiór postaci

$$\{x_1/t_1, \ldots x_n/t_n\} \qquad (1)$$

gdzie $x_1, \ldots, x_n$ są różnymi zmiennymi, natomiast $t_1, \ldots t_n$ są termami. Dopuszczamy podstawienie **puste**, oznaczane ϵ.

Niech θ będzie podstawieniem postaci (1) i niech α będzie wyrażeniem (tzn. formułą lub termem). Piszemy $\alpha\theta$ na oznaczenie wyrażenia powstałego z α w wyniku jednoczesnego zastąpienia wszystkich wolnych wystąpień zmiennych $x_1, \ldots, x_n$ termami $t_1, \ldots t_n$.

Na przykład, jeśli $\theta = \{x/Sam, y/Pam\}$, to

$$(Likes(x, y))\theta = Likes(Sam, Pam).$$

Podstawienie θ jest **unifikatorem** wyrażeń $E_1, \ldots, E_n$ jeśli $E_1\theta = E_2\theta = \cdots = E_n\theta$.

Podstawienie θ jest **najogólniejszym** unifikatorem wyrażeń $E_1, \ldots, E_n$, jeśli dla każdego unifikatora tych wyrażeń, γ, istnieje podstawienie λ takie, że $\gamma = \theta\lambda$.

Reguła rezolucji

Definicja 1

Niech C_1 i C_2 będą klauzulami bez wspólnych zmiennych. Niech $l_1, \ldots, l_n \in C_1$ $(n > 0)$ i $l'_1, \ldots, l'_k \in C_2$ $(k > 0)$. Załóżmy ponadto, że wszystkie literały $l_1, \ldots, l_n$ są dodatnie i wszystkie literały $l'_1, \ldots, l'_k$ są ujemne, lub odwrotnie. Jeśli θ jest najogólniejszym unifikatorem zbioru $\{|\, l_1 \,|, \ldots, |\, l_n \,|, |\, l'_1 \,|, \ldots, |\, l'_k \,|\}$, to klauzulę

$$(C_1\theta - \{l_1\theta, \ldots, l_n\theta\}) \vee (C_2\theta - \{l'_1\theta, \ldots, l'_k\theta\})$$

nazywamy **binarną rezolwentą** klauzul C_1 i C_2.

Przykład

Niech

$$C_1 = P(x) \vee P(y) \vee Q(B).$$
$$C_2 = R(A) \vee \neg P(F(A)).$$

Klauzula $Q(B) \vee R(A)$ jest binarną rezolwentą klauzul C_1 i C_2.
($l_1 = P(x), l_2 = P(y), l_1' = \neg P(F(A));\ \ \theta = \{x/F(A), y/F(A)\}$.)

Definicja 2
Rezolwentą klauzul C_1 i C_2, oznaczaną $RES(C_1, C_2)$, nazywamy dowolną binarną rezolwentę wariantu klauzuli C_1 i wariantu klauzuli C_2.

Definicja 3
Regułą rezolucji nazywamy regułę wnioskowania

$$\frac{C_1, C_2}{RES(C_1, C_2)}.$$

Dowód rezolucyjny ze zbioru klauzul CL:
Ciąg klauzul $C_1, \ldots, C_n$ taki, że $C_n = \perp$ i dla każdego $1 \leq i \leq n$, $C_i \in CL$ lub C_i jest rezolwentą C_j i C_k, gdzie $j, k < i$.

Twierdzenie
Niech CL będzie zbiorem klauzul nie zawierających symbolu "=". CL jest niespełnialny wtw gdy istnieje dowód rezolucyjny z CL.

Aksjomaty równości

CL – zbiór klauzul. Niech $EQ(CL)$ – **zbiór aksjomatów równości dla** CL. $EQ(CL)$ zawiera następujące klauzule:

$x = x$.

$x \neq y \vee y = x$.

$x \neq y \vee y \neq z \vee x = z$.

$x_1 \neq y_1 \vee \cdots \vee x_n \neq y_n \vee \neg P(x_1, \ldots, x_n) \vee P(y_1, \ldots, y_n)$,
dla każdego n-argumentowego ($n > 0$) symbolu predykatowego P występującego w CL.

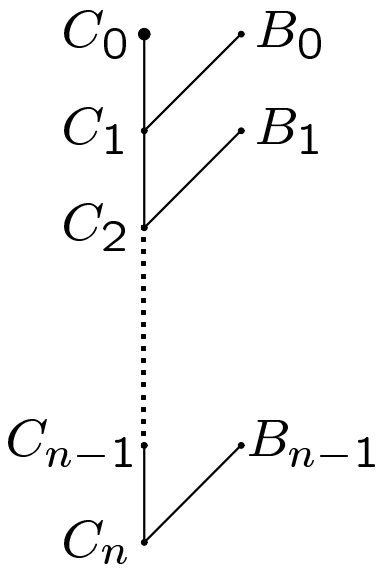
$x_1 \neq y_1 \vee \cdots \vee x_n \neq y_n \vee (F(x_1, \ldots, x_n) = F(y_1, \ldots, y_n))$,
dla każdego n-argumentowego ($n > 0$) symbolu funkcyjnego F występującego w CL.

Twierdzenie

Niech CL będzie zbiorem klauzul z równością. CL jest niespełnialny wtw gdy istnieje dowód rezolucyjny z $CL \cup EQ(CL)$.

Rezolucja liniowa

Liniowym dowodem rezolucyjnym ze zbioru klauzul CL nazywamy dowód postaci



gdzie:

1. $C_0 \in CL$
2. Dla $0 \leq i < n$, $B_i \in CL$ lub $B_i = C_J$, dla pewnego $j < i$.
3. Dla $1 \leq i \leq n$, C_i jest rezolwentą C_{i-1} i B_{i-1}.
4. $C_n = \bot$.

Twierdzenie Niech CL będzie zbiorem klauzul nie zawierających symbolu "=". CL jest niespełnialny wtw gdy istnieje liniowy dowód rezolucyjny z CL.

Twierdzenie Niech CL będzie zbiorem klauzul z równością. CL jest niespełnialny wtw gdy istnieje liniowy dowód rezolucyjny ze zbioru $CL \cup EQ(CL)$.

Klauzulę C_0 w liniowym dowodzie rezolucyjnym nazywamy **klauzulą główną**.

Twierdzenie Niech CL będzie niespełnialnym zbiorem klauzul i niech $CL' \subseteq CL$. Jeśli zbiór $CL - CL'$ jest spełnialny, to istnieje liniowy dowód rezolucyjny z CL (ewentualnie z $CL \cup EQ(CL)$) z klauzulą główną $C \in CL'$.