Wykład 6.

6.1 Zupełność *SLD*-rezolucji

Niech P będzie programem. Przyjmijmy, że

$$s(P) = \{A : A \text{ jest stałym atomem i istnieje } SLD\text{-wyprowadzenie z } P \text{ klauzuli } \Box, \text{ rozpoczynające się celem } \leftarrow A\}.$$

Z twierdzenia 2.3 o poprawności rezolucji i wniosku 4.5 otrzymujemy następujący

Wniosek 6.1 *Zachodzi zawieranie* $s(P) \subseteq T_P^\omega$.

Dowód. W strukturze $M_P = \mathcal{H}_{T_P^\omega}$ są spełnione wszystkie klauzule z P. Niech $A \in s(P)$. Z twierdzenia 2.3 wynika, że $M_P \models A\theta$ dla pewnego podstawienia θ. Ponieważ A jest formułą bez zmiennych, więc $A\theta = A$. Stąd i z definicji M_P wynika, że $\mathcal{H}_{T_P^\omega} \models A$. Definicja modeli Herbranda implikuje, że $A \in T_P^\omega$. $\Box$

Lemat 6.2 *Zbiory* $s(P)$ *i* T_P^ω *są równe.*

Dowód. Wystarczy dowieść, że atomy należące do T_P^ω należą także do $s(P)$. Niech $A \in T_P^\omega$. Przyjmijmy, że $T_0 = \emptyset$ oraz $T_{n+1} = T_P(T_n)$. Z definicji T_P^ω wynika, że $T_P^\omega = \bigcup_{i=1}^\infty T_i$. Wobec tego $A \in T_k$ dla pewnego $k \in N$. Dalszy dowód będzie prowadzony przez indukcję ze względu na k.

Załóżmy więc, że $A \in T_{n+1} = T_P(T_n)$ i – na podstawie założenia indukcyjnego – że $T_n \subseteq s(P)$. Z definicji operatora T_P mamy, że istnieją atomy $B_1, \ldots, B_m \in T_n$ takie, że klauzula $A \leftarrow B_1, \ldots, B_m$ nie zawiera zmiennych i została otrzymana w wyniku podstawiania θ_1 w klauzuli $A' \leftarrow B'_1, \ldots, B'_m \in P$. Będziemy definiować *SLD*-wyprowadzenie z programu P klauzuli $\Box$, rozpoczynające się klauzulą $\leftarrow A$.

Najpierw zauważmy, że jeżeli mamy *SLD*-wyprowadzenie klauzuli $\Box$, rozpoczynające się klauzulą $\leftarrow B$, oraz stałe atomy $C_1, \ldots, C_l$, to po zastąpieniu w tym wyprowadzeniu klauzul $\leftarrow N$ klauzulami $\leftarrow (N, C_1, \ldots, C_l)$ otrzymujemy *SLD*-wyprowadzenie klauzuli $\leftarrow (C_1, \ldots, C_l)$, rozpoczynające się klauzulą $\leftarrow (B, C_1, \ldots, C_l)$. Jest to konsekwencją oczywistego faktu, że każdego stałego atomu C_i zachodzi równość $C_i\theta = C_i$.

Wiemy, że $B_1, \ldots, B_m \in T_n \subseteq s(P)$. Wobec tego istnieją *SLD*-wyprowadzenia z P klauzuli $\Box$ rozpoczynające się formułami $\leftarrow B_1, \ldots, \leftarrow B_m$. Z powyższej uwagi, przez indukcję ze względu na m otrzymujemy, że te wyprowadzenia można przekształcić w *SLD*-wyprowadzenie z P klauzuli $\Box$, rozpoczynające się formułą $\leftarrow (B_1, \ldots, B_m)$.

Konstruowane *SLD*-wyprowadzenie z programu P klauzuli $\Box$ oczywiście rozpoczyna się klauzulą $\leftarrow A$. W pierwszym kroku tego wyprowadzenia atom A unifikujemy z poprzednikiem klauzuli $A' \leftarrow B'_1, \ldots, B'_m$. Podstawienie θ_1 jest unifikatorem A i A'. Przyjmijmy, że $\theta_1 = \theta'_1\eta$ dla θ'_1 będącego najogólniejszym unifikatorem A i A'. Po zastosowaniu reguły rezolucji otrzymujemy rezolwentę $\leftarrow (B'_1, \ldots, B'_m)\theta'_1$. Zauważmy, że

$$\leftarrow (B'_1, \ldots, B'_m)\theta'_1\eta = \leftarrow (B'_1, \ldots, B'_m)\theta_1 = \leftarrow (B_1, \ldots, B_m).$$

Tak więc z klauzuli $\leftarrow (B'_1, \ldots, B'_m)\theta'_1\eta$ można wyprowadzić klauzulę pustą. Z lematu 5.2 o podstawianiu wynika, że klauzulę pustą można wyprowadzić także z klauzuli $\leftarrow (B'_1, \ldots, B'_m)\theta'_1$. Dopisując wyprowadzenie z tej klauzuli do $\leftarrow A$ otrzymujemy szukane wyprowadzenie klauzuli pustej rozpoczynające się $\leftarrow A$. Z istnienia tego wyprowadzenia wynika, że $A \in s(P)$. $\Box$

Twierdzenie 6.3 (o zupełności *SLD*-rezolucji) *Przypuśćmy, że P jest programem i N jest celem. Jeżeli zbiór klauzul $P \cup \{N\}$ jest sprzeczny, to istnieje SLD-wyprowadzenie z programu P klauzuli $\Box$, rozpoczynające się celem N.*

Dowód. Niech $N = \leftarrow A_1, \ldots A_n$. Z założenia mamy, że także M_P nie jest modelem $P \cup \{N\}$. Z wniosku 4.5 wynika, że M_P jest modelem P. Wobec tego $M_P \not\models N$, a więc istnieje wartościowanie h takie, że $M_P \not\models N[h]$. Z definicji spełniania i z prawa de Morgana otrzymujemy, że $M_P \models (A_1, \ldots, A_n)[h]$.

Niech X będzie zbiorem zmiennych występujących w $A_1, \ldots, A_n$. Ponieważ M_P jest modelem Herbranda, więc wartościowanie h przypisuje zmiennym termy stałe. Niech θ będzie podstawieniem, które za zmienne ze zbioru X podstawia termy przypisywane tym zmiennym przez wartościowanie h. Nietrudno sprawdzić, że jeżeli w termie t występują tylko zmienne ze zbioru X, to jego wartość $t[h]$ w modelu Herbranda, przy wartościowaniu h jest termem stałym $t\theta$. Korzystając z faktu 3.2, dla symbolu relacyjnego p (i odpowiednich termów) mamy, że w dowolnym modelu Herbranda $\mathcal{H}_I$

$$\mathcal{H}_I \models p(t_1, \ldots, t_m)[h] \Leftrightarrow (t_1[h], \ldots, t_m[h]) \in p^{\mathcal{H}_I} \Leftrightarrow (t_1\theta, \ldots, t_m\theta) \in p^{\mathcal{H}_I}$$

$$\Leftrightarrow p(t_1\theta, \ldots, t_m\theta) \in I \Leftrightarrow \mathcal{H}_I \models p(t_1\theta, \ldots, t_m\theta) \Leftrightarrow \mathcal{H}_I \models p(t_1, \ldots, t_m)\theta$$

Stąd otrzymujemy, że w rozważanej sytuacji zachodzi własności $M_P \models (A_1, \ldots, A_n)\theta$, a więc (po skorzystaniu z definicji modelu M_P) także $A_1\theta, \ldots, A_n\theta \in T_P^\omega$.

Skorzystajmy teraz z lematu 6.2. Wynika z niego, że $A_1\theta, \ldots, A_n\theta \in s(P)$. Tak więc z negacji każdego z wymienionych atomów można wyprowadzić klauzulę pustą. Stosując metodę opisamą w dowodzie lematu 6.2, wyprowadzenia te przekształcamy w wyprowadzenie klauzuli pustej, rozpoczynające się klauzulą $\leftarrow (A_1, \ldots, A_n)\theta$. Teraz teza wynika bezpośrednio z lematu 5.2. $\Box$

Wniosek 6.4 *Przypuśćmy, że P jest programem i N jest celem. Zbiór klauzul $P \cup \{N\}$ jest sprzeczny wtedy i tylko wtedy, gdy istnieje SLD-wyprowadzenie z programu P klauzuli $\Box$, rozpoczynające się celem N.*

Dowód. Niech $N = \leftarrow A_1, \ldots, A_n$. Jedna z implikacji wynika z twierdzenia o zupełności . Implikacja odwrotna jest konsekwencją twierdzenia 2.3 o poprawności rezolucji. Dla dowodu nie wprost załóżmy, że w strukturze $\mathcal{A}$ spełnione są wszystkie klauzule ze zbioru $P \cup \{N\}$. Z twierdzenia 2.3 otrzymujemy, że $\mathcal{A} \models (A_1 \wedge \ldots \wedge A_n)\theta$ dla pewnego podstawienia θ. Spełnianie w tej strukturze celu N implikuje, że $\mathcal{A} \models (\neg A_1 \vee \ldots \vee \neg A_n)\theta$, a to nie jest możliwe. $\Box$

6.2 Podstawienia poprawne i obliczone

Przypuśćmy, że mamy dane program P i cel $N = \leftarrow A_1, \ldots, A_n$. Dla takich danych definiujemy podstawienia poprawne i obliczone. Podstawienia te nie zmieniają zmiennych nie występujących w N, a ponadto mają niżej podane własności. Podstawieniem poprawnym nazywamy takie podstawienie θ, że formuła $(A_1 \wedge \ldots \wedge A_n)\theta$ (czyli $\neg N\theta$) jest spełniona w dowolnej strukturze, w której są spełnione wszystkie klauzule z programu P. Podstawienie θ nazywamy obliczonym, jeżeli istnieją ciągi celów $N_0, N_1, \ldots, N_k$, klauzul $C_1, \ldots, C_k$ z programu P i podstawień $\theta_1, \ldots, \theta_k$ tworzących *SLD*-wyprowadzenie z programu P klauzuli pustej, rozpoczynające się klauzulą $N_0 = N$, oraz $x\theta = x\theta_1 \ldots \theta_k$ dla wszystkich zmiennych występujących w N. Z twierdzenia 2.3 wynika, że podstawienia obliczone są poprawne.

Udowodnimy także

Twierdzenie 6.5 *Jeżeli P jest programem, N jest celem, a θ – podstawieniem poprawnym (dla programu P i celu N), to istnieje podstawienie obliczone bardziej ogólne niż θ.*

Dowód. Najpierw udowodnimy pewien fakt pomocniczy. Rozważmy program P i cel N. Przypuśćmy, że mamy zbiór $\{a_1, \ldots, a_n\}$ (nowych) stałych nie występujących w programie P i równoliczny z nim zbiór zmiennych $\{x_1 \ldots, x_n\}$ (występujących w klauzuli N). Niech γ będzie podstawieniem $\{a_1/x_1, \ldots, a_n/x_n\}$ podstawiającym stałe a_i za zmienne x_i, a δ – podstawieniem odwrotnym $\{x_1/a_1, \ldots, x_n/a_n\}$. Wobec tego $\delta\gamma$ i $\gamma\delta$ są podstawieniami identycznościowymi (pustymi, niezmieniającymi żadnych termów). Rozważmy program P i cel N. Niech $N_0, N_1, N_2, \ldots, N_k$, $C_1, C_2, \ldots, C_k$ i $\theta_1, \theta_2, \ldots, \theta_k$ będą *SLD*-wyprowadzeniem z programu P klauzuli pustej, rozpoczynającym się klauzulą $N_0 = N\gamma$, wraz z ciągiem wariantów klauzul programowych i najogólniejszych unifikatorów wykorzystywanych w tym wyprowadzeniu. Rozważmy ciągi $N_0\delta, N_1\delta, N_2\delta, \ldots, N_k\delta$, $C_1\delta, C_2\delta, \ldots, C_k\delta$ i $\gamma\theta_1\delta, \gamma\theta_2\delta, \ldots, \gamma\theta_k\delta$. Zauważmy, że ciągi te tworzą *SLD*-wyprowadzenie z programu P klauzuli pustej, rozpoczynające się klauzulą $N_0\gamma\delta = N$. Oczywiście, $C_i\delta$ jest równe C_i (gdyż stałe a_j nie występują w programie P), a więc $C_i\delta$ jest wariantem klauzuli z programu P. Jest też jasne, że $\gamma\theta_i\delta$ jest unifikatorem klauzul $N_{i-1}\delta$ oraz $C_i\delta$, ponieważ θ_i jest unifikatorem N_{i-1} oraz C_i. Jest najogólniejszym unifikatorem tych klauzul, gdyż dowolny unifikator θ, a więc podstawienie takie, że $N_{i-1}\delta\theta = C_i\delta\theta$, można przedstawić tak, aby $\theta = \theta_i\eta$ dla pewnego podstawienia η. Ponieważ $\delta\theta$ unifikuje N_{i-1} oraz C_i, więc $\delta\theta = \theta_i\eta'$ dla pewnego η'. Stąd otrzymujemy, że $\theta = \gamma\delta\theta = (\gamma\theta_i\delta)\gamma\eta'$. Teraz już można się przekonać, że jeżeli N_i jest rezolwentą klauzul N_{i-1} oraz C_i, to $N_i\delta$ jest rezolwentą klauzul $N_{i-1}\delta$ oraz $C_i\delta$. Tak więc zostało skonstruowane *SLD*-wyprowadzenie z programu P klauzuli pustej, rozpoczynające się klauzulą N, i dodatkowo podstawienia w tym wyprowadzeniu są takie, że ich złożenie $\gamma\theta_1\delta\gamma\theta_2\delta \ldots \gamma\theta_k\delta = \gamma\theta_1\theta_2 \ldots \theta_k\delta$ nie podstawiają za zmienne x_i występujące w klauzuli N. Wynika to stąd, że $a_i\theta_j = a_i$ dla dowolnych i oraz j, a więc $x_i\gamma\theta_1\theta_2 \ldots \theta_k\delta = a_i\theta_1\theta_2 \ldots \theta_k\delta = a_i\delta = x_i$.

Udowodnioną własność można w skrócie wyrazić w następujący sposób: jeżeli P, N i γ mają wyżej podane własności i istnieje *SLD*-wyprowadzenie z programu P klauzuli pustej, rozpoczynającym się klauzulą $N\gamma$, to podstawienie identycznościowe (puste) jest jednym z podstawień obliczonych.

Niech teraz P, N i θ spełniają założenia twierdzenia. Będziemy korzystać z podanej uwagi dla programu P i celu $N\theta$. Poprawność podstawienia θ oznacza, że formuła $\neg N\theta$ jest spełniona w każdej strukturze, w której są spełnione wszystkie klauzule z P. Weźmy dowolną taką strukturę $\mathcal{A}$ i rozszerzmy ją o stałe $a_1, \ldots, a_n$ (stałych tych jest tyle, ile zmiennych występuje w N). Formuła $\neg N\theta$ jest spełniona w strukturze $\mathcal{A}$. Stąd $\mathcal{A} \models \forall \vec{x}\ \neg N\theta$. Niech $\gamma = \{a_1/x_1, \ldots, a_n/x_n\}$. Mamy także $\mathcal{A} \models \neg N\theta\gamma$. Oznacza to, że zbiór klauzul $P \cup \{N\theta\gamma\}$ jest sprzeczny. Korzystając z twierdzenia 6.3 o zupełności otrzymujemy, że istnieje *SLD*-wyprowadzenie z programu P klauzuli pustej, rozpoczynające się celem $N\theta\gamma$. Z uwagi z początku dowodu istnieje także *SLD*-wyprowadzenie z programu P klauzuli pustej, rozpoczynające się celem $N\theta$, z podstawieniem pustym jako podstawieniem obliczonym. Niech $\theta_1, \ldots, \theta_k$ będzie ciągiem najogólniejszych unifikatorów wykorzystywanych w tym wyprowadzeniu (tak więc $x\theta_1 \ldots \theta_k = x$ dla zmiennych x występujących w $N\theta$).

Teraz skorzystamy z lematu 5.2 o podstawieniu. Wynika z niego, że istnieje *SLD*-wyprowadzenie z programu P klauzuli pustej, rozpoczynające się celem N, z takim ciągiem $\theta'_1, \ldots, \theta'_k$ najogólniejszych unifikatorów, że złożenie $\theta'_1 \ldots \theta'_k$ jest bardziej ogólne niż $\theta\theta_1 \ldots \theta_k$. Tak więc $\theta\theta_1 \ldots \theta_k = \theta'_1 \ldots \theta'_k\eta$ dla pewnego η.

Niech $(\theta'_1 \ldots \theta'_k)_N$ oznacza podstawienie $\theta'_1 \ldots \theta'_k$ obcięte do zmiennych występujących w N, a więc pewne podstawienie obliczone. Zauważmy, że podstawienia θ i $(\theta'_1 \ldots \theta'_k)_N$ podstawiają tylko za zmienne występujących w N i za te zmienne podstawiają to samo. Dla x występujących w N mamy bowiem, że $\theta(x) = x\theta\theta_1 \ldots \theta_k = x\theta'_1 \ldots \theta'_k\eta = x\theta'_1 \ldots \theta'_k\eta = (\theta'_1 \ldots \theta'_k)_N(x)\eta$.
$\square$