ROCZNIKI FILOZOFICZNE
Tom XLVI-XLVII, zeszyt 1 – 1998-1999

PIOTR KULICKI
Lublin

LOGIKA PROGRAMOWANIA A SYLOGISTYKA ARYSTOTELESA

Przedmiotem pracy jest porównanie założeń metodologicznych programowania w logice i założeń metodologicznych sylogistyki Arystotelesa. W szczególności pokazujemy, że postulat wyrażenia teorii pierwszego rzędu w języku klauzul Horna, konieczny do wykorzystania tej teorii jako programu komputerowego, jest równoważny z warunkiem, który musi spełniać sylogistyka Arystotelesa, by mogła być adekwatnie zaksjomatyzowana.

Niech T będzie ustaloną teorią pierwszego rzędu. Wyrażenia o postaci

$$X \leftarrow Y,$$

gdzie $X, Y \subseteq T$ są skończonymi zbiorami wyrażeń atomowych, reprezentują klauzule, tj. implikacje, których warunek jest koniunkcją formuł ze zbioru Y, a konkluzja alternatywą wyrażeń ze zbioru X. Jeżeli zbiór X spełnia warunek $|X| \leq 1$, to mówimy o klauzuli, iż jest klauzulą Horna. Klauzulę $\varnothing \leftarrow \varnothing$ nazywamy klauzulą pustą i interpretujemy jako sprzeczność. Dla uproszczenia zapisu klauzul będziemy opuszczać teoriomnogościowe nawiasy, a zamiast symbolu sumy zbiorów pisać będziemy przecinek. Z klauzul możemy otrzymywać inne przy użyciu reguły rezolucji o schemacie:

$$\frac{X_1 \leftarrow Y_1, \alpha;\ X_2, \beta \leftarrow Y_2}{X_1^{\#}, X_2^{\#} \leftarrow Y_1^{\#}, Y_2^{\#}},$$

gdzie $X_1, X_2, Y_1, Y_2 \subseteq T$ oraz $\alpha, \beta \in T$ są odpowiednio skończonymi zbiorami formuł atomowych i formułami atomowymi, formuły α i β dają się zunifikować, tj. istnieją podstawienia s_1 i s_2 takie, że $s_1\alpha = s_2\beta$, a zbiory formuł opatrzone symbolem # są otrzymane przez zastosowanie podstawień użytych do unifikacji formuł α i β. Przesłanki reguły nazywane są rodzicami, wniosek – rezolwentą.

Będziemy mówili, że klauzula $C = \alpha_1, ..., \alpha_i \leftarrow \beta_1, ..., \beta_j$ (i, j $\geq$ 0) jest wyprowadzalna ze zbioru klauzul Γ, jeżeli ze zbioru Γ i klauzul $\emptyset \leftarrow \alpha_1^*$, ..., $\emptyset \leftarrow \alpha_i^*$, $\beta_1^* \leftarrow \emptyset$, .., $\beta_j^* \leftarrow \emptyset$ (wyrażenia opatrzone gwiazdkami powstają przez zastąpienie wszystkich zmiennych różnymi stałymi nie występującymi w $\Gamma \cup \{C\}$) można przy użyciu rezolucji otrzymać klauzulę pustą. Klauzula jest ważna w teorii T, jeżeli odpowiadająca jej implikacja jest w niej zawsze prawdziwa. Teorię pierwszego rzędu T nazywamy t e o r i ą H o r n a, jeżeli istnieje adekwatna aksjomatyzacja tej teorii w języku klauzul Horna, tj. jeżeli istnieje zbiór klauzul Horna Γ, z którego są wyprowadzalne wszystkie klauzule ważne w teorii T i tylko one. Znaczenie teorii Horna przejawia się w fakcie, iż mogą one być użyte jako programy w języku logiki.

TWIERDZENIE: Teoria T jest teorią Horna wtedy i tylko wtedy, gdy ma następującą własność dysjunkcji:

(WD) $\quad X, Y \leftarrow Z \in \Gamma$ wtw $X \leftarrow Z \in \Gamma$ lub $Y \leftarrow Z \in \Gamma$,

gdzie X, Y, Z $\subseteq$ T są skończonymi zbiorami atomów oraz Γ jest zbiorem wszystkich ważnych klauzul w teorii T.

Dowód: ($\rightarrow$). Niech Γ_1 będzie zbiorem wszystkich klauzul Horna ważnych w teorii T, a klauzula $X, Y \leftarrow Z$ będzie postaci: $\alpha_1, ..., \alpha_i \leftarrow \beta_1, ..., \beta_j$. Zgodnie z założeniem klauzula ta jest wyprowadzalna z Γ_1. Stąd ze zbioru klauzul $\Gamma_1 \cup \{\emptyset \leftarrow \alpha_1^*, ..., \emptyset \leftarrow \alpha_i^*, \beta_1^* \leftarrow \emptyset, ..., \beta_j^* \leftarrow \emptyset\}$ (wyrażenia z gwiazdkami rozumiemy tak jak poprzednio) można otrzymać klauzulę pustą. Zauważmy, że wszystkie klauzule z tego zbioru są klauzulami Horna. Analizując możliwość stosowania w tym przypadku reguły rezolucji, widzimy, że konkluzja rezolwenty pochodzi od „lewego rodzica", a więc:

1) każda rezolwenta jest klauzulą Horna,

2) konkluzja „prawego rodzica" zawsze będzie pojedynczą formułą i nie może być pusta (w przeciwnym wypadku zastosowanie reguły rezolucji nie byłoby możliwe),

3) jedynie konkluzja „lewego rodzica" może być pusta i w tym przypadku konkluzja rezolwenty jest także pusta i rezolwenta ta może być dalej użyta jedynie jako „lewy rodzic".

Z obserwacji tych wynika, że pusta konkluzja pochodzi dokładnie od jednej klauzuli z pustą konkluzją. Zatem klauzulę pustą można otrzymać ze zbioru klauzul $\Gamma_1 \cup \{\varnothing \leftarrow \alpha_k^*, \beta_l^* \leftarrow \varnothing, ..., \beta_j^* \leftarrow \varnothing\}$ dla pewnego $k \leq i$. Stąd klauzula $\alpha_k \leftarrow \beta_l, ..., \beta_j$; jest wyprowadzalna z $\Gamma_1 \subseteq \Gamma$. Ponieważ $\alpha_k \in X \cup Y$, zatem $X \leftarrow Z \in \Gamma$ lub $Y \leftarrow Z \in \Gamma$.

($\leftarrow$). Z równoważności (WD) wynika, że każda ważna klauzula może być rozłożona równoważnie na klauzule Horna. Stąd zbiór wszystkich ważnych klauzul Horna stanowi adekwatną aksjomatykę dla teorii T. **Q.E.D.**

J. Łukasiewicz (por. [1]) sformalizował sylogistykę zdań asertorycznych jako teorię nabudowaną na klasycznym rachunku zdań. Posługując się metodą aksjomatycznego odrzucania, podał dla tej teorii procedurę rozstrzygania, która daje się zastosować do wszystkich sylogizmów sformułowanych przez Arystotelesa. Nie jest ona jednak wystarczająco ogólna, ponieważ umieszczenie sylogistyki w kontekście klasycznego rachunku zdań pozwala na sformułowanie nieskończonej liczby wyrażeń, które w tak zbudowanej teorii nie dają się rozstrzygnąć. J. Słupecki (por. [2]) sformułował rekursywny schemat reguł odrzucania, który dołączony do systemu Łukasiewicza pozwala na odrzucenie wszystkich wyrażeń uprzednio nie rozstrzygniętych, czyniąc z niego rozstrzygalny system, pokrywający się z fragmentem algebry zbiorów, utożsamianym powszechnie z sylogistyką. Reguła Słupeckiego ma schemat:

$$\frac{\dashv \neg\alpha \rightarrow (\gamma_1 \wedge ... \wedge \gamma_k \rightarrow \gamma),\ \dashv \neg\beta \rightarrow (\gamma_1 \wedge ... \wedge \gamma_k \rightarrow \gamma)}{\dashv (\neg\alpha \wedge \neg\beta) \rightarrow (\gamma_1 \wedge ... \wedge \gamma_k \rightarrow \gamma)},$$

gdzie formuły $\gamma, \gamma_1, ... \gamma_k$ są atomami[1] bądź ich negacjami, formuły α, β są atomami, a symbol $\dashv$ reprezentuje odrzucanie w systemie sylogistyki.

W zapisie klauzulowym reguła otrzymuje równoważną postać:

$$\frac{\dashv X, \alpha \leftarrow Y;\ \dashv X, \beta \leftarrow Y}{\dashv X, \alpha, \beta \leftarrow Y},$$

[1] Dokładniej – atomami są pozytywne zdania kategoryczne.

gdzie X i Y oraz α i β są odpowiednio skończonymi zbiorami atomów oraz atomami.

Oczywiście jest ona równoważna z regułą:

$$(\mathrm{WD}^*) \qquad \frac{\dashv X \leftarrow Z; \quad \dashv Y \leftarrow Z}{\dashv X \cup Y \leftarrow Z},$$

gdzie X, Y oraz Z są skończonymi zbiorami atomów.

Można ją także wzmocnić i wyrazić w postaci następującej równoważności:

$$\dashv X, Y \leftarrow Z \text{ wtw } \dashv X \leftarrow Z \text{ oraz } \dashv Y \leftarrow Z.$$

Korespondencja tej reguły z warunkiem (WD) z twierdzenia narzuca się sama. Zachodzenie reguły WD* pozwala na ograniczenie problemu odrzucania w sylogistyce do klauzul Horna. Zaskakującym faktem jest to, że postulat rozkładu formuł na klauzule Horna, wyrażony właśnie przez tę regułę, wraz z aksjomatami odrzucania podanymi przez Łukasiewicza oraz używaną przez niego regułą transpozycji wystarcza do kompletnego zaksjomatyzowania zbioru formuł odrzuconych sylogistyki. Ponieważ sylogistyka w ujęciu Łukasiewicza jest teorią Horna, zachodzi dla niej pozytywny odpowiednik warunku (WD) z twierdzenia, tj.

$$\vdash X, Y \leftarrow Z \text{ wtw } \vdash X \leftarrow Z \text{ lub } \vdash Y \leftarrow Z.$$

Dlatego też zarówno uznawanie, jak i odrzucanie dowolnych formuł w sylogistyce opiera się na uznaniu i odrzucaniu klauzul Horna. Widać stąd, że można stosować narzędzia programowania w logice do obu tych aspektów sylogistyki.

BIBLIOGRAFIA

1. Łukasiewicz J., O sylogistyce Arystotelesa, (Sprawozdania Polskiej Akademii Umiejętności, 44, nr 6), Kraków 1939, s. 220-227.
2. Słupecki J., Z badań nad sylogistyką Arystotelesa, Wrocław 1948.

LOGIC OF PROGRAMMING AND ARISTOTLE'S SYLLOGISTIC

Summary

In the paper we compare methodological assumptions underlying Aristotle's syllogistic and logic programming. To be used as a logic program a theory has to be expressed in the language of Horn clauses. This is possible if a certain form of disjunction property holds for that theory. Aristotle's syllogistic requires the same form of disjunction property for complete axiomatisation of its theorems and non-theorems. Such analogy makes it possible to use the tools of logic programming in syllogistic, and on the other hand to use techniques developed for syllogistic in programming.

Summarized by Piotr Kulicki