Logika

Materiały pomocnicze do wykładu
dla pierwszego roku informatyki UW

Paweł Urzyczyn
urzy@mimuw.edu.pl

25 sierpnia 2005

Wnioskowanie o prawdziwości rozmaitych stwierdzeń jest powszednim zajęciem matematyków i nie tylko matematyków. Dlatego filozofowie i matematycy od dawna zajmowali się systematyzacją metod wnioskowania i kryteriów ich poprawności. Oczywiscie ostatecznym kryterium poprawności rozumowania pozostaje zawsze zdrowy rozsądek i przekonanie o słuszności wywodu. Logika, jako nauka o rozumowaniu, jest jednak ważnym i potrzebnym narzędziem, które to przekonanie ułatwia.

Szczególną rolę wśród rozmaitych działów logiki zajmuje logika matematyczna, poświęcona opisowi i analizie poprawności wnioskowań matematycznych. Jest to dyscyplina w pewnym sensie paradoksalna: będąc sama częścią matematyki, traktuje matematykę jako swój przedmiot zainteresowania. Dla uniknięcia „błędnego koła” musimy więc tutaj zauważyć, że logika formalna nie opisuje rzeczywistych wywodów matematyka, ale ich uproszczone modele, które bez zastrzeżeń można uważać za zwykłe obiekty matematyczne. Mimo tego ograniczenia, logika matematyczna dostarcza niezwykle ważnych wniosków o charakterze filozoficznym i metamatematycznym.

Logika formalna była kiedyś ezoteryczną nauką z pogranicza filozofii i matematyki. Później okazało się jednak, że metody logiki formalnej znalazły ważne zastosowania praktyczne, dostarczając technik weryfikacji poprawności programów.

Jak stwierdziliśmy wyżej, logika matematyczna zajmuje się badaniem rozmaitych systemów formalnych, modelujących rzeczywiste sposoby wnioskowania matematycznego. W ramach tego wykładu zajmować się będziemy trzema takimi systemami. Najszerszy z nich to *logika pierwszego rzędu*, zwana też *rachunkiem predykatów* lub *rachunkiem kwantyfikatorów*. Dwa ważne fragmenty logiki pierwszego rzędu, którym poświęcimy wiele uwagi, to *wnioskowanie równościowe* i *rachunek zdań*.

1 Struktury relacyjne

Definicja 1.1 Przez *sygnaturę* rozumiemy pewien (zwykle skończony) zbiór *symboli relacyjnych i funkcyjnych*, każdy z ustaloną liczbą argumentów. Sygnaturę Σ przedstawić można jako sumę:

$$\Sigma = \bigcup_{n\in\mathbb{N}} \Sigma_n \;\cup \bigcup_{n\in\mathbb{N}} \Sigma_n^R,$$

gdzie Σ_n jest zbiorem n-argumentowych *symboli funkcyjnych*, a Σ_n^R jest zbiorem n-argumentowych *symboli relacyjnych*. O symbolu $f \in \Sigma_n$ powiemy, że jego *arność* to liczba n i napiszemy $ar(f) = n$. Podobnie dla $r \in \Sigma_n^R$. Elementy zbioru Σ_0 to *symbole stałych*. Natomiast elementy zbioru Σ_0^R nazwiemy *symbolami zdaniowymi*.

Uwaga: Jeżeli sygnatura jest skończona, to prawie wszystkie ze zbiorów Σ_n i Σ_n^R są puste.

Oczywiście n-argumentowe symbole funkcyjne posłużą nam jako nazwy n-argumentowych funkcji a n-argumentowe symbole będą nazwami n-argumentowych relacji. Przypomnijmy, że n-argumentowa relacja w zbiorze A to dowolny podzbiór iloczynu kartezjańskiego A^n. Inaczej mówiąc, jest to pewien zbiór krotek postaci $\langle a_1, \ldots, a_n\rangle$, gdzie $a_1, \ldots, a_n \in A$. Krotkę $\langle a\rangle$ utożsamiamy z elementem a, tj. uważamy, że A^1 to to samo co A. Natomiast zbiór A^0 ma tylko jeden element, mianowicie pustą krotkę $\langle\ \rangle$. Są więc tylko dwie relacje zero-argumentowe: pusta i pełna. Możemy je oznaczać odpowiednio przez 0 i 1.

Skoro zbiór A^0 ma tylko jeden element, to funkcja zeroargumentowa $f : A^0 \to A$ przyjmuje tylko jedną wartość. Będziemy więc każdą taką funkcję nazywać *stałą* i utożsamiać z odpowiednim elementem zbioru A.

Definicja 1.2 *Struktura relacyjna* albo *model* sygnatury Σ to niepusty zbiór (zwany *dziedziną* lub *nośnikiem* struktury) wraz z interpretacją symboli sygnaturowych jako funkcji i relacji o odpowiedniej liczbie argumentów. Dokładniej, jeśli $\Sigma = \{f_1, \ldots, f_n, r_1, \ldots, r_m\}$, to *strukturą relacyjną* (*modelem*) nazywamy krotkę postaci:

$$\mathcal{A} = \langle A, f_1^{\mathcal{A}}, \ldots, f_n^{\mathcal{A}}, r_1^{\mathcal{A}}, \ldots, r_m^{\mathcal{A}}\rangle,$$

gdzie dla dowolnego i:

- $f_i^{\mathcal{A}} : A^k \to A$, jeśli $ar(f_i) = k$ (w szczególności $f_i^{\mathcal{A}} \in A$, gdy $k = 0$);
- $r_i^{\mathcal{A}} \subseteq A^k$, gdy $ar(r_i) = k$.

Definicja 1.3 Jeśli sygnatura Σ nie zawiera żadnych symboli relacyjnych, to modele tej sygnatury nazywamy *algebrami*.

Konwencje notacyjne: Nośnik struktury $\mathcal{A}$ oznaczamy przez $|\mathcal{A}|$. Często przyjmujemy domyślnie, że $|\mathcal{A}| = A$, $|\mathcal{B}| = B$ itd., lub po prostu strukturę i jej nośnik oznaczamy tym samym symbolem. Często też tak samo oznacza się symbol relacyjny (funkcyjny) i odpowiadającą mu relację (funkcję).

Przykład 1.4 Niech $\Sigma = \{+, \bullet, 0, 1, \leq\}$, gdzie $\Sigma_2 = \{+, \bullet\}$, $\Sigma_0 = \{0, 1\}$, oraz $\Sigma_2^R = \{\leq\}$. Modelem dla sygnatury Σ jest oczywiście zbiór liczb rzeczywistych ze zwykłymi działaniami i porządkiem:

$$\mathcal{R} = \langle \mathbb{R}, +^{\mathcal{R}}, \bullet^{\mathcal{R}}, 0^{\mathcal{R}}, 1^{\mathcal{R}}, \leq^{\mathcal{R}} \rangle.$$

Ten model zwykle zapiszemy po prostu tak:

$$\mathcal{R} = \langle \mathbb{R}, +, \cdot, 0, 1, \leq \rangle.$$

Inne modele dla tej sygnatury to np. zbiór liczb naturalnych ze zwykłymi działaniami i porządkiem oraz zbiór wszystkich podzbiorów $\mathbb{R}$ z działaniami mnogościowymi i inkluzją:

$$\mathcal{N} = \langle \mathbb{N}, +, \cdot, 0, 1, \leq \rangle;$$
$$\mathcal{P} = \langle \mathbf{P}(\mathbb{R}), \cup, \cap, \emptyset, \mathbb{R}, \subseteq \rangle.$$

Ale modelem jest też taka struktura:

$$\mathcal{A} = \langle \mathbb{R}, \cdot, f, \pi, 0, \emptyset \rangle,$$

gdzie $f(a, b) = 3$ dla dowolnych liczb a, b (symbol "+" jest interpretowany jako mnożenie!).

Przykład 1.5 Modelami dla sygnatury $\Sigma = \{\bullet, 1\}$, gdzie $\Sigma_2 = \{\bullet\}$ i $\Sigma_0 = \{1\}$, są na przykład struktury $\langle \mathbb{N}, +, 0 \rangle$ i $\langle \mathbb{N}, *, 1 \rangle$, oraz algebra słów z konkatenacją i słowem pustym: $\langle \{a, b\}^*, \cdot, \varepsilon \rangle$.

Przykład 1.6 Graf zorientowany $G = \langle V, E \rangle$, gdzie V jest zbiorem wierzchołków, natomiast $E \subseteq V \times V$ jest zbiorem krawędzi, jest modelem jednoelementowej sygnatury $\Sigma = \Sigma_2^R = \{r\}$.

Definicja 1.7 *Podstruktura* struktury $\mathcal{A}$ sygnatury Σ to taki niepusty podzbiór $B \subseteq A$, który jest zamknięty ze względu na wszystkie operacje, tj. dla dowolnego n i dowolnego $f \in \Sigma_n$ spełnia warunek:

- Jeśli $a_1, \ldots a_n \in B$ to $f^{\mathcal{A}}(a_1, \ldots, a_n) \in B$.

Taki podzbiór B jest oczywiście nośnikiem pewnej struktury

$$\mathcal{B} = \langle B, f_1^{\mathcal{B}}, \ldots, f_n^{\mathcal{B}}, r_1^{\mathcal{B}}, \ldots, r_m^{\mathcal{B}} \rangle,$$

której operacje i relacje są odpowiednimi obcięciami operacji i relacji w $\mathcal{A}$, tj. $f_i^{\mathcal{B}} = f^{\mathcal{A}}|_{B^{k_i}}$ oraz $r_j^{\mathcal{B}} = r_j^{\mathcal{A}} \cap B^{l_j}$, dla wszystkich i, j. Taką strukturę też nazywamy podstrukturą $\mathcal{A}$, i zwykle po prostu utożsamiamy ze zbiorem B. Podstruktury algebr nazywamy oczywiście *podalgebrami*.

Przykład 1.8

- Zbiór liczb naturalnych jest podalgebrą algebry liczb rzeczywistych z dodawaniem i mnożeniem, ale nie jest podalgebrą algebry liczb rzeczywistych z dodawaniem, mnożeniem i odejmowaniem.
- Rodzina złożona za wszystkich skończonych podzbiorów zbioru $\mathbb{N}$ i ich dopełnień (nazywanych zbiorami *koskończonymi*) jest podalgebrą algebry $\langle \mathbf{P}(\mathbb{N}), \cup, \cap, \emptyset, \mathbb{N}, \rangle$.

Fakt 1.9

- *Jeśli iloczyn jakiejś rodziny podstruktur struktury $\mathcal{A}$ jest niepusty, to też jest podstrukturą struktury $\mathcal{A}$.*
- *Jeśli w sygnaturze jest choć jedna stała, lub zbiór $B \subseteq |\mathcal{A}|$ jest niepusty, to istnieje najmniejsza podstruktura struktury $\mathcal{A}$ zawierająca B. Jest to iloczyn wszystkich podstruktur zawierających B.*

Dowód: Łatwe ćwiczenie. ∎

Definicja 1.10 Najmniejszą podstrukturę struktury $\mathcal{A}$ zawierającą zbiór B nazywamy podstrukturą *generowaną* przez B. Jeśli jest to cała struktura $\mathcal{A}$, to mówimy, że B jest *zbiorem generatorów* struktury $\mathcal{A}$. Zwrot *„Struktura $\mathcal{A}$ ma k generatorów”* oznacza, że istnieje zbiór generatorów mocy k. Struktura (algebra) generowana przez zbiór pusty nazywana jest strukturą (algebrą) *Herbranda*.

Przykład 1.11

- Podstruktura generowana w $\langle \mathbb{R}, +, \cdot, 0, 1, \leq \rangle$ przez $\{-1\}$ składa się ze wszystkich liczb całkowitych, a generowana przez zbiór pusty — ze wszystkich liczb naturalnych. Nie istnieje skończony zbiór generatorów dla $\langle \mathbb{R}, +, \cdot, 0, 1, \leq \rangle$.
- Struktura $\langle \mathbb{N}, 0, \mathbf{s} \rangle$, gdzie $\mathbf{s}(n) = n + 1$, dla dowolnego $n \in \mathbb{N}$, jest algebrą Herbranda.

Fakt 1.12 *Niech $\mathcal{A} = \langle A, f_1^{\mathcal{A}}, \ldots, f_n^{\mathcal{A}}, r_1^{\mathcal{A}}, \ldots, r_m^{\mathcal{A}} \rangle$, gdzie symbole $f_1, \ldots, f_n$ są odpowiednio $k_1, \ldots, k_n$-argumentowe, i niech B będzie podzbiorem A. Podstruktura generowana przez B jest sumą wstępującego ciągu zbiorów B_n, gdzie $B_0 = B$ oraz $B_{n+1} = B_n \cup \bigcup_{i=1}^{n} \vec{f}_i(B_n^{k_i})$.*

Dowód: Ćwiczenie. ∎

Homomorfizmy

Definicja 1.13 *Homomorfizmem* ze struktury $\mathcal{A}$ w stukturę $\mathcal{B}$ (tej samej sygnatury Σ) jest każde przekształcenie $h : A \to B$, które zachowuje funkcje i relacje struktury $\mathcal{A}$, tj.

- dla dowolnego $f \in \Sigma_n$ i dowolnych $a_1, \ldots, a_n \in |\mathcal{A}|$, zachodzi $h(f^{\mathcal{A}}(a_1, \ldots, a_n)) = f^{\mathcal{B}}(h(a_1), \ldots, h(a_n))$;
- dla dowolnego $r \in \Sigma_n^R$ i dowolnych $a_1, \ldots, a_n \in |\mathcal{A}|$, jeśli $\langle a_1, \ldots, a_n \rangle \in r^{\mathcal{A}}$ to także $\langle h(a_1), \ldots, h(a_n) \rangle \in r^{\mathcal{B}}$.

Homomorfizm jest *silny*, wtedy i tylko wtedy, gdy dodatkowo, dla dowolnego $r \in \Sigma_n^R$, spełnia taki warunek:

- jeśli $\langle h(a_1), \ldots, h(a_n) \rangle \in r^{\mathcal{B}}$ dla $a_1, \ldots, a_n \in |\mathcal{A}|$, to istnieją takie $a'_1, \ldots, a'_n \in |\mathcal{A}|$, że $h(a'_1) = h(a_1), \ldots, h(a'_n) = h(a_n)$, oraz $\langle a'_1, \ldots, a'_n \rangle \in r^{\mathcal{A}}$.

Na szczęście najczęściej interesują nas homomorfizmy algebr, a oczywiście:

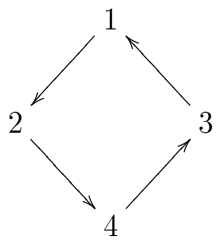
Fakt 1.14 *Każdy homomorfizm algebr jest silny.* ∎

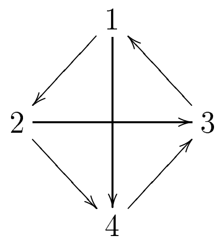
Definicja 1.15 Homomorfizm $h : \mathcal{A} \xrightarrow[\text{na}]{1-1} \mathcal{B}$ nazywamy izomorfizmem, jeżeli przekształcenie odwrotne $h^{-1} : \mathcal{B} \xrightarrow[\text{na}]{1-1} \mathcal{A}$ też jest homomorfizmem. Piszemy wtedy $\mathcal{A} \approx \mathcal{B}$.

Przykład 1.16

- Przekształcenie $h : \{0,1\}^* \to \mathbb{N}$, dane przez $h(w) = |w|$, jest homomorfizmem z algebry $\langle \{0,1\}^*, \cdot, \varepsilon \rangle$ w algebrę $\langle \mathbb{N}, +, 0 \rangle$.
- Izomorfizm porządkowy zbiorów częściowo uporządkowanych A i B to to samo co izomorfizm struktur $\langle A, \leq \rangle$ i $\langle B, \leq \rangle$.
- Reszta z dzielenia przez 3 jest silnym homomorfizmem grafów na rysunku:

- Funkcja identycznościowa jest homomorfizmem grafów na drugim rysunku, ale nie jest to homomorfizm silny ani izomorfizm.





Uwaga: Pojęcie izomorfizmu, jak zawsze w matematyce, służy do utożsamienia ze sobą tych obiektów, które są nierozróżnialne ze względu na interesujące nas własności. W naszym przypadku, stwierdzenie, że dwie struktury (algebry) są izomorficzne, oznacza, że operacje i relacje są w nich określone dokładnie tak samo. Struktury izomorficzne mogą się tylko różnić od siebie nośnikami. W algebrze (i logice) takie struktury uważa się za identyczne, co pozwala np. na takie stwierdzenia jak *„istnieje dokładnie jedna taka algebra, że ...”*

Fakt 1.17

- *Złożenie homomorfizmów jest homomorfizmem.*
- *Obraz podalgebry przy homomorfizmie jest podalgebrą przeciwdziedziny.*
- *Homomorfizm jest izomorfizmem wtedy i tylko wtedy, gdy jest silnym homomorfizmem i bijekcją.*[1]

Dowód: Łatwe ćwiczenie. ∎

Lemat 1.18 *Jeśli zbiór C generuje strukturę $\mathcal{A}$ oraz $h_1, h_2 : \mathcal{A} \to \mathcal{B}$ są homomorfizmami spełniającymi warunek $h_1|_C = h_2|_C$, to $h_1 = h_2$.*

Dowód: Łatwo sprawdzić, że zbiór $D = \{a \in A \mid h_1(a) = h_2(a)\}$ jest podstrukturą struktury $\mathcal{A}$. Z założenia $C \subseteq D$, a więc $A = D$, bo A jest najmniejszą podstrukturą zawierającą C. ∎

Morał 1.19 *Homomorfizm jest jednoznacznie wyznaczony przez wartości jakie przyjmuje na generatorach. Wystarczy znać te wartości i już znamy homomorfizm. Uwaga: nie zawsze istnieje homomorfizm przypisujący generatorom żądane wartości.*

[1]A zatem homomorfizm algebr jest izomorfizmem wtedy i tylko wtedy, gdy jest bijekcją.

Kongruencje i ilorazy

Definicja 1.20 Relacja równoważności $\sim$ w zbiorze $|\mathcal{A}|$ jest *kongruencją* w $\mathcal{A}$ wtedy i tylko wtedy, gdy dla dowolnego n i dowolnego $f \in \Sigma_n$ zachodzi warunek:

$$\text{Jeśli } a_1 \sim a'_1,\ \ldots,\ a_n \sim a'_n \text{ to } f^{\mathcal{A}}(a_1, \ldots, a_n) \sim f^{\mathcal{A}}(a'_1, \ldots, a'_n).$$

Przykład 1.21

- Relacja przystawania modulo 7 jest kongruencją w algebrze liczb całkowitych z dodawaniem i mnożeniem.
- Relacja *„różnica symetryczna zbiorów x i y jest skończona”* jest kongruencją w algebrze $\langle \mathbf{P}(\mathbb{N}), \cup, \cap, \emptyset, \mathbb{N}, \rangle$.

Fakt 1.22 *Iloczyn dowolnej niepustej rodziny kongruencji jest kongruencją.*

Dowód: Łatwy. ∎

Przypomnijmy, że *jądrem* przekształcenia $f : A \to B$ nazywamy relację równoważności $\ker(f) = \{\langle a, b\rangle \in A \times A \mid f(a) = f(b)\}$.

Fakt 1.23 *Jądro dowolnego homomorfizmu jest kongruencją.*

Dowód: Niech $h : \mathcal{A} \to \mathcal{B}$ będzie homomorfizmem i przypuśćmy, że $h(a_i) = h(a'_i)$ dla $i = 1, \ldots, n$. Jesli teraz $f \in \Sigma_n$ to: $h(f^{\mathcal{A}}(a_1, \ldots, a_n)) = f^{\mathcal{B}}(h(a_1), \ldots, h(a_n)) = f^{\mathcal{B}}(h(a'_1), \ldots, h(a'_n)) = h(f^{\mathcal{A}}(a'_1, \ldots, a'_n))$. ∎

Definicja 1.24 Niech $\sim$ będzie kongruencją w strukturze $\mathcal{A} = \langle A, f_1^{\mathcal{A}}, \ldots, f_n^{\mathcal{A}}, r_1^{\mathcal{A}}, \ldots, r_m^{\mathcal{A}}\rangle$. *Struktura ilorazowa* struktury $\mathcal{A}$ wyznaczona przez relację $\sim$ to struktura

$$\mathcal{A}/_{\sim} = \langle A/_{\sim}, f_1^{\mathcal{A}/\sim}, \ldots, f_n^{\mathcal{A}/\sim}, r_1^{\mathcal{A}/\sim}, \ldots, r_m^{\mathcal{A}/\sim}\rangle,$$

gdzie operacje i relacje są zdefiniowane tak:

- Jeśli f_i ma k argumentów, to $f_i^{\mathcal{A}/\sim}([a_1]_\sim, \ldots, [a_k]_\sim) = [f_i^{\mathcal{A}}(a_1, \ldots, a_k)]_\sim$.
- Jeśli r_i ma k argumentów, to warunek $\langle [a_1]_\sim, \ldots, [a_k]_\sim\rangle \in r^{\mathcal{A}/\sim}$ zachodzi wtedy i tylko wtedy, gdy istnieją takie $a'_1, \ldots, a'_k$, że $a_1 \sim a'_1, \ldots, a_k \sim a'_k$ oraz $\langle a'_1, \ldots, a'_k\rangle \in r^{\mathcal{A}}$.

Uwaga: Poprawność (jednoznaczność) powyższej definicji wynika natychmiast stąd, że $\sim$ jest kongruencją.

Fakt 1.25 *Niech $\sim$ będzie kongruencją w $\mathcal{A}$. Wówczas przekształcenie $\kappa : \mathcal{A} \xrightarrow{\text{na}} \mathcal{A}/_\sim$, określone przez $\kappa(a) = [a]_\sim$, jest silnym homomorfizmem.*

Dowód: Łatwe ćwiczenie. ∎

Homomorfizm κ nazywany bywa homomorfizmem *kanonicznym*. Relacja $\sim$ jest oczywiście jądrem tego homomorfizmu, skąd otrzymujemy:

Morał 1.26 *Kongruencje i jądra homomorfizmów to to samo.*

Nastepujące twierdzenie, zwane (pierwszym) twierdzeniem o homomorfizmie, to dalszy krok w tym samym kierunku. Wraz z Faktem 1.25 stwierdza ono, że ilorazy danej struktury to to samo, co jej obrazy przy silnych homomorfizmach.

Twierdzenie 1.27 *Niech $h : \mathcal{A} \xrightarrow{\text{na}} \mathcal{B}$ będzie silnym homomorfizmem, którego jądrem jest relacja $\sim$. Wtedy $\mathcal{B}$ jest izomorficzne z $\mathcal{A}/_\sim$.*

Dowód: Izomorfizm $i : \mathcal{A}/_\sim \to \mathcal{B}$ jest określony równaniem $i([a]_\sim) = h(a)$. Ponieważ warunki $a \sim a'$ i $h(a) = h(a')$ są równoważne, więc i jest dobrze określoną funkcją różnowartościową. Ta funkcja jest też „na", bo h jest „na". Jest to homomorfizm, bo:

- Jeśli $f \in \Sigma_n$, to $i(f^{\mathcal{A}/\sim}([a_1]_\sim, \ldots, [a_n]_\sim)) = i([f^{\mathcal{A}}(a_1, \ldots, a_n)]_\sim) = h(f^{\mathcal{A}}(a_1, \ldots, a_n)) = f^{\mathcal{B}}(h(a_1), \ldots, h(a_n)) = f^{\mathcal{B}}(i([a_1]_\sim), \ldots, i([a_n]_\sim))$.
- Jeśli $ar(r) = n$ oraz $\langle [a_1]_\sim, \ldots, [a_n]_\sim \rangle \in r^{\mathcal{A}/\sim}$ to $\langle a_1', \ldots, a_n' \rangle \in r^{\mathcal{A}}$, dla pewnych $a_1', \ldots, a_m'$ spełniających warunek $a_1 \sim a_1', \ldots, a_m \sim a_m'$. Wtedy
$$\langle i([a_1]_\sim), \ldots, i([a_n]_\sim) \rangle = \langle h(a_1), \ldots, h(a_n) \rangle = \langle h(a_1'), \ldots, h(a_n') \rangle \in r^{\mathcal{B}},$$
bo h jest homomorfizmem.

Ponadto i jest silnym homomorfizmem, co wynika natychmiast stąd, że h jest silnym homomorfizmem, i że $\langle i([a_1]_\sim), \ldots, i([a_n]_\sim) \rangle$ to to samo co $\langle h(a_1), \ldots, h(a_n) \rangle$. Z Faktu 1.17 wnioskujemy, że i jest izomorfizmem. ∎

Ponieważ każdy homomorfizm algebr jest silny, więc z Faktu 1.25 i Twierdzenia 1.27 wynika:

Morał 1.28 *Ilorazy i obrazy homomorficzne algebr to to samo.*

Produkty

Definicja 1.29 Niech $\mathcal{A}_t = \langle A_t, f_1^t, \ldots, f_n^t, r_1^t, \ldots, r_m^t \rangle$, dla $t \in T$. *Produktem* rodziny indeksowanej $\{\mathcal{A}_t\}_{t \in T}$ nazywamy strukturę

$$\prod_{t \in T} \mathcal{A}_t = \langle \prod_{t \in T} A_t, f_1^\Pi, \ldots, f_n^\Pi, r_1^\Pi, \ldots, r_m^\Pi \rangle,$$

w której operacje i relacje są zdefiniowane tak:

- Jeśli f_i ma k argumentów, to $f_i^\Pi(\xi_1, \ldots, \xi_k)(t) = f_i^t(\xi_1(t), \ldots, \xi_k(t))$;
- Jeśli r_i ma k argumentów, to $\langle \xi_1, \ldots, \xi_k \rangle \in r_i^\Pi$ zachodzi wtedy i tylko wtedy, gdy dla wszystkich $t \in T$ ma miejsce $\langle \xi_1(t), \ldots, \xi_k(t) \rangle \in r_i^t$.

Oczywiście, jeśli zbiór T jest dwuelementowy, na przykład $T = \{0, 1\}$, to zamiast $\prod_{t \in T} \mathcal{A}_t$ piszemy po prostu $\mathcal{A}_0 \times \mathcal{A}_1$, a należące do produktu funkcje $\xi : \{0, 1\} \to A_0 \cup A_1$ utożsamiamy z parami uporządkowanymi $\langle \xi(0), \xi(1) \rangle$.

Jeżeli natomiast wszystkie $\mathcal{A}_t$ są identyczne z $\mathcal{A}$, to zamiast $\prod_{t \in T} \mathcal{A}$ piszemy $\mathcal{A}^T$ i mówimy o strukturze *potęgowej*.

Przykład 1.30 Struktura $\langle \mathbf{P}(\mathbb{N}), \cup, \cap, \emptyset, \mathbb{N}, \rangle$ jest izomorficzna z potęgą $\langle \{0, 1\}, \vee, \wedge, 0, 1 \rangle^\mathbb{N}$, gdzie $i \vee j = \max\{i, j\}$ oraz $i \wedge j = \min\{i, j\}$.

Definicja 1.31 (Rzuty) Homomorfizm $\pi_s : \prod_{t \in T} \mathcal{A}_t \xrightarrow{\text{na}} \mathcal{A}_s$, określony przez warunek $\pi_s(\xi) = \xi(s)$, nazywamy *rzutowaniem na współrzędną s*.

2 Termy

Jak powiedzieliśmy na początku, symbole należące do sygnatury mają nam służyć jako *nazwy* pewnych funkcji i relacji. *Znaczenie* tych nazw zależy oczywiście od wybranego modelu. W szczególności symbole stałych są nazwami ustalonych elementów modelu. Inne elementy modelu też mogą być nazwane, na przykład jeśli $f \in \Sigma_2$ i $c \in \Sigma_0$ to możemy *napisać* coś takiego:

$$„f(f(c, c), f(c, f(c, c)))",$$

co w modelu $\mathcal{A} = \langle A, \mathrm{f}, \mathrm{c} \rangle$ będzie oczywiście nazwą elementu $\mathrm{f}(\mathrm{f}(\mathrm{c}, \mathrm{c}), \mathrm{f}(\mathrm{c}, \mathrm{f}(\mathrm{c}, \mathrm{c})))$.

W ten sposób można jednak nazywać tylko elementy podstruktury generowanej przez stałe. Aby nazywać dowolne elementy modelu potrzebujemy *zmiennych*. Ustalmy więc pewien zbiór symboli V, którego elementy (oznaczane $x, y, \ldots$) będziemy nazywali *zmiennymi indywiduowymi*, lub po prostu *zmiennymi*. Zwykle przyjmuje się, że V jest nieskończonym zbiorem przeliczalnym, wygodnie jest też zakładać, że jego elementy są ponumerowane, tj. $V = \{x_i : i \in \mathbb{N}\}$. Okazjonalnie będziemy dopuszczać zbiory zmiennych dowolnej mocy.

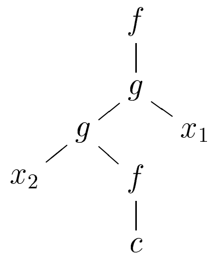
Definicja 2.1 *Przez termy sygnatury Σ rozumiemy elementy najmniejszego zbioru napisów T_Σ (ozn. też po prostu przez T) spełniającego warunki:*

- $V \subseteq T_\Sigma$;
- *jeśli* $f \in \Sigma_n$ *oraz* $t_1, \ldots, t_n \in T_\Sigma$ *to* „$f(t_1, \ldots, t_n)$” $\in T_\Sigma$.

Konwencje notacyjne: Niektóre dwuargumentowe symbole funkcyjne, jak np. „+”, „$\cup$” są tradycyjnie pisane pomiędzy argumentami. Dlatego i my zamiast formalnie poprawnego „+(2, 2)” zwykle napiszemy „2+2”.

Przykład 2.2 Wyrażenie „$f(g(g(x_2, f(c)), x_1))$” jest termem sygnatury Σ, gdzie $g \in \Sigma_2$, $f \in \Sigma_1$ oraz $c \in \Sigma_0$. Wyrażenie „$(0 + x_1) \bullet 1$” jest termem sygnatury Σ, w której $+, \bullet \in \Sigma_2$ i $0, 1 \in \Sigma_0$.

Często wygodnie jest reprezentować termy za pomocą drzew skończonych, w których liście są etykietowane zmiennymi i stałymi, a wierzchołki wewnętrzne symbolami funkcyjnymi. Oczywiście stopień wyjściowy wierzchołka (liczba dzieci) musi się zgadzać z liczbą argumentów użytego symbolu funkcyjnego. Na przykład term $f(g(g(x_2, f(c)), x_1))$ przedstawiamy jako:



Definicja 2.3 Dla dowolnego termu t, zbiór *zmiennych wolnych* termu t, oznaczany przez $FV(t)$, jest określony przez indukcję:

- $FV(x) = \{x\}$, gdy x jest zmienną;
- $FV(f(t_1, \ldots, t_n)) = FV(t_1) \cup \cdots \cup FV(t_n)$.

Jeśli $X \subseteq V$, to stosujemy takie oznaczenia:

- $T_\Sigma(X) = \{t \in T_\Sigma \mid FV(t) \subseteq X\}$
- $T_\Sigma(n) = T_\Sigma(\{x_1, \ldots, x_n\})$

Zauważmy, że $FV(c) = \emptyset$, gdy c jest symbolem stałej ($c \in \Sigma_0$).

Definicja 2.4 *Struktura termów* sygnatury Σ, to dowolna taka struktura $\mathcal{A}$, że:

- $|\mathcal{A}| = T_\Sigma(X)$, dla pewnego X;
- $f^{\mathcal{A}}(t_1, \ldots, t_n) =$ „$f(t_1, \ldots, t_n)$”, dla dowolnego $f \in \Sigma_n$;

Symbole relacyjne w strukturze termów mogą być interpretowane dowolnie, istnieje więc wiele takich struktur.

W przypadku sygnatur bez symboli relacyjnych, mówimy oczywiście o *algebrach termów*. Oczywiście istnieje wtedy tylko jedna algebra wszystkich termów, oznaczana przez $\mathcal{T}_\Sigma$. Dla dowolnego k mamy też jednoznacznie wyznaczoną algebrę termów k-argumentowych $\mathcal{T}_\Sigma(k)$.

Wygodnie jest rozszerzyć pojecie algebry termów na dowolne sygnatury. Zrobimy to tak: *algebra termów* $\mathcal{T}_\Sigma(X)$ to taka struktura termów o nośniku $T_\Sigma(X)$, w której wszystkie relacje są... puste.

Fakt 2.5 *Zbiór $X \subseteq V$ jest zbiorem generatorów algebry $\mathcal{T}_\Sigma(X)$.*

Dowód: Ćwiczenie. ∎

Wartościowania i podstawienia

Oczywiście chcemy używać termów jako nazw obiektów indywiduowych (elementów jakiegoś modelu $\mathcal{A}$). To znaczy, że chcemy każdemu termowi przypisać jego *wartość* w modelu $\mathcal{A}$.

Definicja 2.6 *Wartościowaniem* w danej strukturze $\mathcal{A}$ nazywamy dowolny homomorfizm $v : \mathcal{T}_\Sigma \to \mathcal{A}$.

Łatwo widzieć, że każde wartościowanie v jest jednoznacznie wyznaczone przez swoje obcięcie do zbioru zmiennych, tj. funkcję $v|_V : V \to |\mathcal{A}|$. Wynika to z Morału 1.19. Co więcej:

Fakt 2.7 *Każda funkcja $u : V \to |\mathcal{A}|$ jednoznacznie rozszerza się do pewnego wartościowania $v : \mathcal{T}_\Sigma \to \mathcal{A}$.*

Dowód: Dla dowolnego $t \in T_\Sigma$ zdefiniujemy $v(t)$ przez indukcję ze względu na długość t:

a) $v(x) = u(x)$, gdy x jest zmienną;

b) $v(f(t_1, \ldots, t_n)) = f^{\mathcal{A}}(v(t_1), \ldots, v(t_n))$, gdy $ar(f) = n$ i $t_1, \ldots, t_n \in T_\Sigma$.

Nietrudno sprawdzić, że tak określone przekształcenie jest homomorfizmem. (Zwłaszcza warunek dotyczący relacji jest łatwy do sprawdzenia.) ∎

Wartościowanie w $\mathcal{A}$ często definiuje się właśnie jako funkcję $u : V \to \mathcal{A}$, a następnie rozszerza się do $v : \mathcal{T}_\Sigma \to \mathcal{A}$ za pomocą warunków (a) i (b) występujących w dowodzie Faktu 2.7. Piszemy wtedy $v(t) = [\![t]\!]_u$. Jak widać, wychodzi na to samo, więc zwykle utożsamia się u i v.

Jeśli znamy wartości zmiennych, to znamy wartość dowolnego termu. W istocie ważne są tylko te zmienne, które w danym termie występują.

Fakt 2.8 *Jeśli v_1, v_2 są takimi wartościowaniami, że $v_1|_{FV(t)} = v_2|_{FV(t)}$, to $v_1(t) = v_2(t)$.*

Dowód: Jeśli $FV(t) = X$ to term t należy do algebry $\mathcal{T}(X) = \mathcal{T}_\Sigma(X)$ generowanej przez X. Jest to oczywiście podalgebra algebry $\mathcal{T}_\Sigma$. Funkcje $v_1|_{\mathcal{T}(X)}$ i $v_2|_{\mathcal{T}(X)}$ są homomorfizmami z $\mathcal{T}_\Sigma(X)$ do $\mathcal{A}$, które pokrywają się na zbiorze generatorów X. Z Lematu 1.18 wynika, że pokrywają się na całym $\mathcal{T}_\Sigma(X)$, w szczególności $v_1(t) = v_2(t)$. ∎

Morał 2.9 *Wartość termu zależy tylko od wartości jego zmiennych wolnych.*

Zauważmy, że jeśli term nie ma zmiennych wolnych, to jego wartość nie zależy od wartościowania i zamiast $[\![t]\!]_u$ można pisać po prostu $[\![t]\!]$.

Oznaczenie: Przez v_x^a oznaczamy wartościowanie, które dla dowolnej zmiennej y spełnia takie warunki:

$$v_x^a(y) = \begin{cases} a & \text{gdy } y = x; \\ v(y) & \text{w przeciwnym przypadku.} \end{cases}$$

Oczywiście takie wartościowanie jest tylko jedno.

Podstawienia

Definicja 2.10 Wartościowanie w algebrze termów nazywamy *podstawieniem*. Jeśli podstawienie S jest takie, że $S(x_i) = t_i$, dla $i = 1, \ldots, n$ oraz $S(y) = y$ dla wszystkich zmiennych $y \notin \{x_1, \ldots, x_n\}$, to często zamiast $S(t)$ piszemy $t[x_1 := t_1, \ldots, x_n := t_n]$ lub $t[t_1/x_1, \ldots, t_n/x_n]$.

Uwaga: $t[x := s, y := u]$ to zwykle nie to samo, co $t[x := s][y := u]$.

O podstawieniu należy myśleć jak o operacji syntaktycznej. Term $S(t)$ powstaje z termu t przez wstawienie termu $S(x)$ na miejsce każdej zmiennej x.

Przykład: $f(x, y)[x := f(y, x)] = f(f(y, x), y)$.

Lemat 2.11 *Niech S będzie podstawieniem i niech v bedzie wartościowaniem w pewnej strukturze $\mathcal{A}$. Przez v_S oznaczmy takie wartościowanie, że $v_S(x) = v(S(x))$. Wówczas, dla dowolnego termu t, zachodzi równość $v(S(t)) = v_S(t)$. W szczególności:*

$$v(t[x := s]) = v_x^{v(s)}(t).$$

Dowód: Wartościowania $v \circ S$ i v_S są homomorfizmami, które pokrywają się na generatorach. Zatem teza wynika z Lematu 1.18. ∎

Wniosek 2.12 *Niech $FV(t) = \{x_1, \ldots, x_k\}$ i niech $y_1, \ldots, y_k$ będą różnymi zmiennymi, nie należącymi do $FV(t)$. Przyjmijmy $t' = t[x_1 := y_1] \ldots [x_k := y_k]$. Dla dowolnego v istnieje takie w, że $v(t) = w(t')$.*

Dowód: Niech $a_i = v(x_i)$, dla $i = 1, \ldots, n$. Jeśli $w(y_i) = a_i$ to z Lematu 2.11 wynika równość $w(t') = w_{x_1}^{a_1} \cdots {}_{x_k}^{a_k}(t) = v(t)$, bo wartościowania v i $w_{x_1}^{a_1} \cdots {}_{x_k}^{a_k}$ pokrywają się dla argumentów z $FV(t)$. ∎

Morał na zakończenie tej części jest taki: Jeśli wartości zmiennych przebiegają jakiś zbiór, to wartości termów tworzą podalgebrę generowaną przez ten zbiór.

Fakt 2.13 *Niech B będzie podalgebrą generowana w $\mathcal{A}$ przez C. Wtedy następujące warunki są równoważne:*

a) $a \in B$;

b) $a = [\![t]\!]_u$ dla pewnego termu t i pewnego $u : V \to |\mathcal{A}|$, takiego, że $\vec{u}\,(FV(t)) \subseteq C$.

W szczególności, najmniejszą podalgebrą algebry $\mathcal{A}$ (generowaną przez zbiór pusty) jest zbiór $\{[\![t]\!] \mid t \in \mathcal{T}(0)\}$, złożony z wartości termów stałych.

Dowód: Implikację z (b) do (a) można udowodnić przez łatwą indukcję ze względu na długość termu t. Aby udowodnić implikację z (a) do (b) musimy pokazać, że zbiór

$$D = \{[\![t]\!]_u \mid t \in \mathcal{T}_\Sigma \text{ oraz } \vec{u}\,(FV(t)) \subseteq C\}$$

jest podalgebrą algebry $\mathcal{A}$. Przypuśćmy więc, że f jest n-argumentowym symbolem funkcyjnym i że $a_1, \ldots, a_n \in D$. Zatem dla dowolnego $i = 1, \ldots, n$ mamy $a_i = [\![t_i]\!]_{u_i}$ dla pewnych termów t_i i pewnych wartościowań u_i, spełniających $\vec{u}_i(FV(t_i)) \subseteq C$. Na mocy Wniosku 2.12 możemy, bez straty ogólności założyć, że zbiory $FV(t_i)$ są parami rozłączne. Więc istnieje takie wartościowanie u, że $u|_{FV(t_i)} = u_i|_{FV(t_i)}$ dla wszystkich i. Z Faktu 2.8 wynika, że $[\![t_i]\!]_u = a_i$, dla wszystkich i. A zatem $f^{\mathcal{A}}(a_1, \ldots, a_n) = f^{\mathcal{A}}([\![t_1]\!]_u, \ldots, [\![t_n]\!]_u) = [\![f(t_1, \ldots, t_n)]\!]_u \in D$ i dobrze. ∎

3 Klasy algebr

O *klasach* mówimy wtedy, gdy nie możemy mówić o zbiorach, tj. gdy odpowiednie zbiory nie istnieją. Na przykład formułując ogólne stwierdzenia o ciałach, wygodnie jest użyć skrótu myślowego i powiedzieć „klasa wszystkich ciał". Chętnie używamy też wtedy notacji „$\mathfrak{F} = \{\mathcal{A} \mid \mathcal{A}$ jest ciałem$\}$" i piszemy „$\mathcal{A} \in \mathfrak{F}$" gdy algebra $\mathcal{A}$ jest ciałem. Pamiętajmy jednak, że stosujemy tę terminologię i takie oznaczenia tylko nieformalnie. W szczególności klasy nie są już elementami innych klas.

Umowa: Kiedy mówimy o klasie algebr $\mathfrak{K}$, to zawsze zakładamy, że należą do niej algebry takiej samej sygnatury i że $\mathfrak{K}$ jest zamknięta ze względu na izomorfizmy: jeśli $\mathcal{A} \in \mathfrak{K}$ oraz $\mathcal{A} \approx \mathcal{B}$ to $\mathcal{B} \in \mathfrak{K}$. Algebry izomorficzne uważamy przecież za nieodróżnialne.

Definicja 3.1 Gdy $\mathfrak{K}$ jest klasą algebr to $H(\mathfrak{K})$, $S(\mathfrak{K})$ i $P(\mathfrak{K})$ oznacza odpowiednio:

- klasę wszystkich obrazów homomorficznych algebr z klasy $\mathfrak{K}$;
- klasę wszystkich podalgebr algebr z klasy $\mathfrak{K}$;
- klasę wszystkich produktów algebr z klasy $\mathfrak{K}$.

Piszemy $HSP(\mathfrak{K})$ zamiast $H(S(P(\mathfrak{K})))$.

Lemat 3.2 *Dla dowolnej klasy algebr $\mathfrak{K}$ zachodzą następujące inkluzje:*

$$S(H(\mathfrak{K})) \subseteq H(S(\mathfrak{K})) \qquad P(H(\mathfrak{K})) \subseteq H(P(\mathfrak{K})) \qquad P(S(\mathfrak{K})) \subseteq S(P(\mathfrak{K}))$$

Dowód: Ćwiczenie. ∎

Wniosek 3.3 *Dla dowolnych klas algebr $\mathfrak{K}$ i $\mathfrak{L}$:*

- $\mathfrak{K} \subseteq HSP(\mathfrak{K})$;

- *$HSP(HSP(\mathfrak{K})) = HSP(\mathfrak{K})$;*
- *Jeśli $\mathfrak{K} \subseteq \mathfrak{L}$ to $HSP(\mathfrak{K}) \subseteq HSP(\mathfrak{L})$.*

Dowód: Ćwiczenie. ∎

Trzy charakterystyczne własności wymienione we Wniosku 3.3 pozwalają nam powiedzieć, że *HSP* jest *operacją domknięcia*.

Równości

Napis postaci „$t_1 = t_2$" (gdzie $t_1, t_2 \in \mathcal{T}$) nazywamy *równaniem*. Równanie jest *spełnione* przez wartościowanie v w strukturze $\mathcal{A}$ gdy $v(t_1) = v(t_2)$. Piszemy wtedy $\mathcal{A}, v \models t_1 = t_2$. Jeśli każde wartościowanie w $\mathcal{A}$ spełnia równanie, to mówimy, że jest ono *prawdziwe* w $\mathcal{A}$ i piszemy $\mathcal{A} \models t_1 = t_2$. A jeśli tak jest dla każdej algebry z klasy $\mathfrak{K}$, to piszemy $\mathfrak{K} \models t_1 = t_2$. Jeśli E jest zbiorem równań, to notacja $\mathcal{A} \models E$ i $\mathfrak{K} \models E$ oznacza, że warunek $\mathcal{A} \models e$ (odpowiednio $\mathfrak{K} \models e$) zachodzi dla wszystkich równań $e \in E$.

Uwaga: Czasem mówi się, że algebra (klasa) "spełnia" równanie, zamiast powiedzieć, że to równanie jest prawdziwe w tej algebrze (klasie). Grozi to jednak nieporozumieniem, bo przez "spełnialność" zwykle rozumiemy spełnienie warunku przez *pewne* wartościowanie, a nie przez *każde*.

Definicja 3.4 Mówimy że klasa algebr $\mathfrak{K}$ jest *definiowalna równościowo* (jest *rozmaitością*), gdy istnieje taki zbiór równań E, że dla dowolnej algebry $\mathcal{A}$:

$$\mathcal{A} \in \mathfrak{K} \qquad \text{wtedy i tylko wtedy, gdy} \qquad \mathcal{A} \models E.$$

Stosujemy wtedy notację $\mathfrak{K} = \mathrm{Mod}(E)$.

Uwaga: Niech $\mathrm{Eq}(\mathfrak{K})$ oznacza zbiór wszystkich równań prawdziwych we wszystkich algebrach klasy $\mathfrak{K}$. Oczywiście zachodzą inkluzje $\mathfrak{K} \subseteq \mathrm{Mod}(\mathrm{Eq}(\mathfrak{K}))$ oraz $E \subseteq \mathrm{Eq}(\mathrm{Mod}(E))$.

Przykłady klas definiowalnych równościowo

W przykładach poniżej symbole $+$, $\cdot$ są zawsze dwuargumentowe, symbol $-$ jest jednoargumentowy a 0 i 1 oraz K i S to symbole stałych.

Przykład 3.5 Algebrę postaci $\langle A, \cdot, 1\rangle$, w której prawdziwe są równania

$$„x \cdot 1 = x", \quad „1 \cdot x = x"\,, \quad „x \cdot (y \cdot z) = (x \cdot y) \cdot z"$$

nazywamy *półgrupą z jednością* albo *monoidem*. Często zamiast „półgrupa z jednością" mówimy po prostu „półgrupa". Przykładem półgrupy jest struktura słów $\langle A^*, \cdot, \varepsilon\rangle$. Półgrupa jest *przemienna* (inaczej *abelowa*) gdy jest w niej dodatkowo prawdziwe równanie „$x \cdot y = y \cdot x$"

Przykład 3.6 Jeśli $\langle A, \cdot, 1\rangle$ jest półgrupą, w której określono dodatkową jednoargumentową operację $^{-1}$ o własnościach

$$x \cdot x^{-1} = 1, \qquad x^{-1} \cdot x = 1,$$

to strukturę $\langle A, \cdot, ^{-1}, 1\rangle$ nazywamy *grupą*. Przykładem grupy jest struktura $\langle F, \circ, ^{-1}, \mathrm{id}_A\rangle$, gdzie F jest zbiorem wszystkich bijekcyj z A do A.

Przykład 3.7 *Pierścień* (przemienny z jednością) to struktura $\mathcal{A} = \langle A, \cdot, +, -, 1, 0\rangle$, o takich własnościach:

- $\langle A, +, -, 0\rangle$ jest grupą abelową;
- $\langle A, \cdot, 1\rangle$ jest półgrupą abelową;
- $\mathcal{A} \models x \cdot (y + z) = (x \cdot y) + (x \cdot z)$.

Przykładem pierścienia jest zbiór $\mathbb{R}[x]$ wszystkich wielomianów o współczynnikach rzeczywistych.

Przykład 3.8 Pierścień $\mathcal{A}$ nazywamy *ciałem*, gdy ma co najmniej dwa elementy i dla dowolnego niezerowego $a \in |\mathcal{A}|$ istnieje takie $b \in |\mathcal{A}|$, że $a \cdot b = 1$. Przykładem ciała jest zbiór $\mathbb{Z}(x)$ wszystkich funkcji wymiernych o współczynnikach całkowitych. Albo zbiór $\{0, 1, 2, 3, 4, 5, 6\}$ z działaniami modulo 7.

Uwaga: W przeciwieństwie do poprzednich przykładów, klasa wszystkich ciał *nie jest* definiowalna równościowo.

Przykład 3.9 *Algebra kombinatoryczna* to algebra $\langle A, \cdot, \mathsf{K}, \mathsf{S}\rangle$, w której prawdziwe są równania:

- $(\mathsf{K} \cdot x) \cdot y = x$;
- $(((\mathsf{S} \cdot x) \cdot y) \cdot z) = (x \cdot z) \cdot (y \cdot z)$.

Operacja $\cdot$ nazywana jest operacją *aplikacji*.

Ćwiczenie*: W każdej algebrze kombinatorycznej prawdziwe jest równanie

$$((\mathsf{S}\cdot\mathsf{K})\cdot\mathsf{K})\cdot x = x.$$

Dlatego *kombinator*[2] $(\mathsf{S}\cdot\mathsf{K})\cdot\mathsf{K}$ oznaczany jest przez I. Jak zdefiniować kombinator B i kombinator 2, żeby w każdej takiej algebrze prawdziwe były równania

$$((\mathsf{B}\cdot x)\cdot y)\cdot z = x\cdot(y\cdot z) \qquad \text{i} \qquad (2\cdot x)\cdot y = x\cdot(x\cdot y)?$$

4 Kraty i algebry Boole'a

Poniżej, symbole $\sqcap$, $\sqcup$ są zawsze dwuargumentowe, symbol $-$ jest jednoargumentowy a 0 i 1 to symbole stałych.

Definicja 4.1 *Kratą* nazywamy algebrę $\langle B,\sqcap,\sqcup\rangle$, w której są prawdziwe równania:

$$\begin{array}{rl} x\sqcup y = y\sqcup x & \qquad x\sqcap y = y\sqcap x \\ (x\sqcup y)\sqcup z = x\sqcup(y\sqcup z) & \qquad (x\sqcap y)\sqcap z = x\sqcap(y\sqcap z) \\ x\sqcap(x\sqcup y) = x & \qquad x\sqcup(x\sqcap y) = x. \end{array}$$

Lemat 4.2 *Jeśli $\mathcal{B}=\langle B,\sqcap,\sqcup\rangle$ jest kratą, oraz $a,b\in B$, to*

$$a\sqcup b = b \textit{ wtedy i tylko wtedy, gdy } a\sqcap b = a.$$

Dowód: Jeśli $a\sqcup b = b$ to $a\sqcap b = a\sqcap(a\sqcup b) = a$. Na odwrót podobnie. ∎

Fakt 4.3

1) *Jeśli $\mathcal{B}=\langle B,\sqcap,\sqcup\rangle$ jest kratą, to relacja $\leq$ określona warunkiem*

$$a\leq b \quad \textit{wtedy i tylko wtedy, gdy} \quad a\sqcap b = a,$$

jest częściowym porządkiem i to takim, że $a\sqcup b=\sup\{a,b\}$ i $a\sqcap b=\inf\{a,b\}$, dla dowolnych a,b.

2) *Jeśli $\langle B,\leq\rangle$ jest zbiorem częściowo uporządkowanym, w którym każdy zbiór dwuelementowy ma kres górny i kres dolny, to B jest kratą z operacjami $a\sqcup b=\sup\{a,b\}$ i $a\sqcap b=\inf\{a,b\}$.*

*Dla chętnych

[2]Tak nazywamy termy nad tą sygnaturą.

Dowód: (1) Relacja $\leq$ jest zwrotna, bo $a = a \sqcap (a \sqcup (a \sqcap a)) = a \sqcap a$, dla dowolnego elementu $a \in B$. Jest przechodnia, bo jeśli $a \leq b$ i $b \leq c$, to $a \sqcap c = (a \sqcap b) \sqcap c = a \sqcap (b \sqcap c) = a \sqcap b = a$. Jest też antysymetryczna, bo jeśli $a \leq b$ i $b \leq a$, to $a = a \sqcap b = b \sqcap a = b$.

A więc jest to częściowy porządek. Ponieważ $a \sqcap b = (a \sqcap a) \sqcap b = a \sqcap (a \sqcap b)$, więc $a \sqcap b \leq a$. Podobnie $a \sqcap b \leq b$, zatem iloczyn jest ograniczeniem dolnym dla a i b. Jeśli c jest jakimś innym ograniczeniem dolnym, tj. $a \sqcap c = c$ i $b \sqcap c = c$, to $(a \sqcap b) \sqcap c = a \sqcap (b \sqcap c) = a \sqcap c = c$, czyli $c \leq a \sqcap b$. Wnioskujemy, że iloczyn jest kresem dolnym.

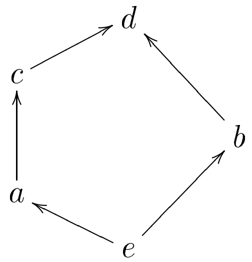
To, że suma jest kresem górnym wynika w analogiczny sposób z Lematu 4.2.

(2) Łatwe ćwiczenie. ∎

Definicja 4.4 Krata jest *dystrybutywna* wtedy i tylko wtedy, gdy są w niej prawdziwe równania:

$$a \sqcap (b \sqcup c) = (a \sqcap b) \sqcup (a \sqcap c); \qquad a \sqcup (b \sqcap c) = (a \sqcup b) \sqcap (a \sqcup c).$$

Przykład 4.5 Krata wypukłych podzbiorów płaszczyzny nie jest dystrybutywna.[3] Inny przykład kraty, która nie jest dystrybutywna, widać na obrazku:



Istotnie: $a \sqcup (b \sqcap c) = a \sqcup e = a$, tymczasem $(a \sqcup b) \sqcap (a \sqcup c) = d \sqcap c = c$.

Definicja 4.6 *Algebrą Boole'a* nazywamy algebrę postaci $\langle B, \sqcap, \sqcup, -, 0, 1\rangle$, gdy $\langle B, \sqcap, \sqcup\rangle$ jest kratą dystrybutywną oraz prawdziwe są równania:

$$\begin{array}{rl} 0 \sqcap x = 0 & \qquad 0 \sqcup x = x \\ 1 \sqcap x = x & \qquad 1 \sqcup x = 1 \\ x \sqcup -x = 1 & \qquad x \sqcap -x = 0. \end{array}$$

Dowolną algebrę Boole'a, jak każdą kratę, można uporządkować (Fakt 4.3), przyjmując, że $a \leq b$ zachodzi wtedy i tylko wtedy, gdy $a \sqcap b = a$. Z Lematu 4.2 mamy oczywiście:

$$a \leq b \quad \text{wtedy i tylko wtedy, gdy} \quad a \sqcup b = b.$$

Stąd natychmiast wynika, że w każdej algebrze Boole'a:

[3]Ćwiczenie: Dlaczego? Wskazówka: jak jest określona suma w tej kracie?

- 1 jest elementem największym;
- 0 jest elementem najmniejszym,

ze względu na uporządkowanie $\leq$.

Przykład 4.7 Oczywiście każdy zbiór postaci $\mathbf{P}(A)$ jest algebrą Boole'a ze zwykłymi operacjami teoriomnogościowymi. Podobnie każda podalgebra takiej algebry (inaczej *ciało zbiorów*). W szczególności rodzina wszystkich skończonych podzbiorów $\mathbb{N}$ i ich dopełnień (tj. zbiorów *koskończonych*) tworzy algebrę Boole'a. Ale najważniejszym przykładem algebry Boole'a jest algebra dwuelementowa

$$\mathcal{B}_0 = \langle \{0,1\}, \wedge, \vee, \neg, 0, 1 \rangle,$$

w której $i \wedge j = \min(i,j)$, $i \vee j = \max(i,j)$ oraz $\neg i = 1 - i$.

Przykładem kraty, która nie jest algebrą Boole'a (chociaż niewiele jej brakuje) jest algebra $\mathcal{H} = \langle H, \cap, \cup, \neg, \emptyset, \mathbb{R}^2 \rangle$, której nośnik H to zbiór wszystkich otwartych podzbiorów płaszczyzny $\mathbb{R}^2$, a $\neg A$ to wnętrze dopełnienia zbioru A. W tej algebrze suma $\neg A \cup A$ zwykle nie pokrywa całego $\mathbb{R}^2$, ale jej dopełnienie jest brzegowe (ma puste wnętrze).

Zanim powrócimy do zagadnienia równościowego definiowania klas algebr, zbierzemy teraz kilka przydatnych własności algebr Boole'a. Niech $\mathcal{B} = \langle B, \sqcap, \sqcup, -, 1, 0 \rangle$ będzie algebrą Boole'a.

Lemat 4.8 *Dla dowolnych $a, b \in B$:*

- *$a \sqcup b = 1$ wtedy i tylko wtedy, gdy $-a \leq b$;*
- *$a \sqcap b = 0$ wtedy i tylko wtedy, gdy $-a \geq b$.*

Dowód: Załóżmy, że $a \sqcup b = 1$. Wtedy $b = b \sqcup 0 = b \sqcup (a \sqcap -a) = (b \sqcup a) \sqcap (b \sqcup -a) = 1 \sqcap (b \sqcup -a) = b \sqcup -a$, a więc $-a \leq b$. Na odwrót, nierówność $-a \leq b$ oznacza, że $b = b \sqcup -a$, a zatem $a \sqcup b = a \sqcup b \sqcup -a = a \sqcup -a \sqcup b = 1 \sqcup b = 1$. W ten sposób pokazaliśmy pierwszą część tezy. Dowód drugiej jest podobny. ∎

Wniosek 4.9 *Dla dowolnego $a \in B$ istnieje dokładnie jedno $b \in B$, spełniające warunki $a \sqcup b = 1$ i $a \sqcap b = 0$. Jest to oczywiście $-a$.*

Wniosek 4.10 *Dla dowolnych $a, b \in B$:*

- *$-(-a) = a$;*

- $-(a \sqcup b) = -a \sqcap -b$;
- $-(a \sqcap b) = -a \sqcup -b$.

Dowód: Wszystkie trzy warunki łatwo wynikają z Wniosku 4.9, bo na przykład $(a \sqcap b) \sqcup (-a \sqcup -b) = ((a \sqcup -a) \sqcap (b \sqcup -a)) \sqcup -b = (1 \sqcap (b \sqcup -a)) \sqcup -b = b \sqcup -a \sqcup -b = 1 \sqcup -a = 1$ i podobnie $(a \sqcap b) \sqcap (-a \sqcup -b) = 0$. Stąd $(-a \sqcup -b)$ musi być dopełnieniem $(a \sqcap b)$. ∎

Czasem wygodnie jest zdefiniować w algebrze Boole'a jeszcze jedną operację:

$$a \Rightarrow b = -a \sqcup b$$

Lemat 4.11 *Dla dowolnych $a, b \in B$:*

- $a \Rightarrow b = 1$ *wtedy i tylko wtedy, gdy* $a \leq b$.

Dowód: Łatwy. ∎

Przechodzimy do definicji bardzo ważnego pojęcia:

Definicja 4.12 *Filtrem* w algebrze $\mathcal{B}$ nazywamy niepusty podzbiór F zbioru B, spełniający dla dowolnych a, b takie warunki:

- Jeśli $a, b \in F$ to $a \sqcap b \in F$;
- Jeśli $a \in F$ i $a \leq b$ to $b \in F$.

Filtr F jest *właściwy* gdy $F \neq B$ (lub równoważnie, gdy $0 \notin F$). Mówimy, że F jest filtrem *maksymalnym*, gdy jest on maksymalny (ze względu na inkluzję) w rodzinie wszystkich filtrów właściwych. Inaczej mówiąc, filtr maksymalny to taki właściwy filtr F, że dla dowolnego właściwego filtru G, inkluzja $F \subseteq G$ oznacza w istocie $F = G$.

O filtrze należy myśleć jak o rodzinie elementów „dużych", tj. takich, które ze względu na jakieś kryterium mogą być uważane za bliskie elementowi największemu 1. Na przykład rodzina koskończonych podzbiorów zbioru $\mathbb{N}$ jest filtrem w algebrze wszystkich podzbiorów tego zbioru. Ale w tej algebrze filtrem (i to maksymalnym) jest także rodzina wszystkich tych zbiorów, do których należy liczba 13. Innym przykładem filtru (w algebrze $\mathbf{P}(\mathbb{R})$) jest rodzina wszystkich tych zbiorów, których dopełnienia są miary zero.

Nietrudno zauważyć, że do każdego filtru musi należeć jedynka, i że zbiór $\{1\}$ jest najmniejszym filtrem. Następny lemat mówi o tym, skąd się biorą filtry, a zwłaszcza maksymalne.

Lemat 4.13

1) *Jeśli $X \subseteq B$ ma taką własność, że żaden skończony iloczyn elementów X nie jest zerem*[4], *to istnieje filtr właściwy zawierający X.*

2) *Jeśli F jest filtrem i $a \notin F$, ale $a \sqcap f \neq 0$ dla dowolnego $f \in F$, to istnieje filtr właściwy zawierający $F \cup \{a\}$.*

3) *Jeśli F jest filtrem właściwym, to istnieje filtr maksymalny zawierający F.*

Dowód: (1) Rozpatrzmy zbiór

$$G = \{b \in B \mid b \geq d_1 \sqcap \ldots \sqcap d_n \text{ dla pewnego } n \text{ i pewnych } d_1, \ldots, d_n \in X\}.$$

Oczywiście $X \subseteq G$, nietrudno też sprawdzić, że G jest filtrem.

(2) Wystarczy zauważyć, że zbiór $F \cup \{a\}$ jest scentrowany. Dowolny skończony iloczyn elementów F jest bowiem elementem F i jest różny od zera, bo skoro $a \notin F$, to F jest właściwy. A więc skończone iloczyny elementów zbioru $F \cup \{a\}$ należą do F lub mają postać $f \sqcap a$.

(3) Rutynowe zastosowanie lematu Kuratowskiego-Zorna (ćwiczenie). ∎

Definicja 4.14 Filtr F jest *pierwszy* wtedy i tylko wtedy, gdy dla dowolnych $a, b \in B$ takich, że $a \sqcup b \in F$, zachodzi $a \in F$ lub $b \in F$.

Filtr F jest *ultrafiltrem* wtedy i tylko wtedy, gdy dla dowolnego $a \in B$ albo $a \in F$ albo $-a \in F$.

Lemat 4.15 *Następujące warunki są równoważne.*

1) *Filtr F jest maksymalny;*

2) *Filtr F jest pierwszy;*

3) *Filtr F jest ultrafiltrem.*

Dowód: (1)⇒(2) Przypuśćmy, że $a \sqcup b \in F$, ale $a, b \notin F$. Jeśli $f \sqcap a \neq 0$ dla dowolnego $f \in F$, to na mocy lematu 4.13(2), filtr F nie jest maksymalny. Podobnie dla b, a więc istnieją takie $f, g \in F$, że $f \sqcap a = g \sqcap b = 0$. Zatem $0 = (f \sqcap a) \sqcup (g \sqcap b) = (f \sqcup g) \sqcap (f \sqcup b) \sqcap (a \sqcup g) \sqcap (a \sqcup b) \in F$, bo wszystkie cztery człony iloczynu należą do filtru F.

[4]Taki zbiór nazywamy *scentrowanym*.

(2)$\Rightarrow$(3) Oczywiste, bo $a \sqcup -a = 1 \in F$.

(3)$\Rightarrow$(1) Przypuśćmy, że $F \subseteq G$ i G jest właściwy. Jeśli teraz $a \in G$ to $a \in F$, bo w przeciwnym razie $-a \in F \subseteq G$, i byłoby tak: $0 = a \sqcap -a \in G$. ∎

Uwaga: Istnieje silny związek pomiędzy filtrami i kongruencjami. Łatwo zauważyć, że dla dowolnej kongruencji $\sim$, klasa $[1]_\sim$ jest filtrem. Na odwrót, każdy filtr F wyznacza taką kongruencję $\sim_F$:

$$a \sim_F b \quad \text{wtedy i tylko wtedy, gdy} \quad a \sqcap f = b \sqcap f, \text{ dla pewnego } f \in F.$$

Oczywiście $F = [1]_{\sim_F}$. Kongruencję $\sim_F$ można też zdefiniować za pomocą warunku:

$$a \sim_F b \quad \text{wtedy i tylko wtedy, gdy} \quad (a \sqcap b) \sqcup (-a \sqcap -b) \in F.$$

Sprawdzenie tych własności pozostawiamy dociekliwym jako ćwiczenie.

5 Teorie równościowe

Definicja 5.1 Niech E będzie zbiorem równań. Relację $\sim_E$ w zbiorze termów $\mathcal{T}_\Sigma$ definiujemy jako najmniejszą relację równoważności spełniającą warunki:

- Jeśli „$t = u$" $\in E$ to $t \sim_E u$;
- Jeśli $t_i \sim_E t'_i$, dla $i = 1, \ldots, n$ to $f(t_1, \ldots, t_n) \sim_E f(t'_1, \ldots, t'_n)$ (tj. relacja $\sim_E$ jest zamknięta ze względu na działania);
- Jeśli $t \sim_E u$ to $S(t) \sim_E S(u)$ dla dowolnego podstawienia S (tj. relacja $\sim_E$ jest zamknięta ze względu na podstawienia).

Jeśli $t \sim_E u$ to piszemy też $E \vdash_\forall t = u$. Używamy tu specjalnie symbolu „$\vdash_\forall$", żeby się odróżniał od zwykłego „$\vdash$". Ale na razie to nie ma znaczenia.

Warunek $t \sim_E u$ można wyrazić w następujący sposób. Równość „$t = u$" można *udowodnić*, wyprowadzając ją z *aksjomatów równościowych:*

$$t = t \qquad\qquad t = s \quad (\text{dla } \text{“}t = s\text{”} \in E)$$

za pomocą nastepujacych *reguł wnioskowania równościowego:*

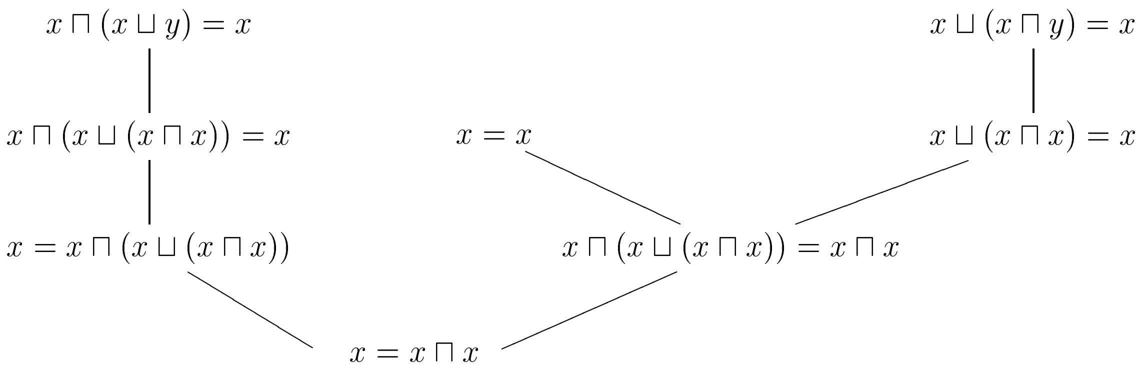
$$\frac{t = s}{s = t} \qquad\qquad \frac{t = s, s = r}{t = r}$$

$$\frac{t = s}{S(s) = S(t)} \qquad\qquad \frac{t_1 = s_1, \ldots, t_n = s_n}{f(t_1, \ldots, t_n) = f(s_1, \ldots, s_n)}$$

Równania zapisane nad kreską to *przesłanki* danej reguły, a równanie pod kreską to jej *konkluzja*. Przez *wyprowadzenie* (czyli *dowód*) równania „$t = u$” należy rozumieć drzewo etykietowane równaniami w ten sposób, że:

- Etykietą korzenia jest „$t = u$”;
- Etykietą każdego liścia jest jakiś aksjomat;
- Etykieta każdego wierzchołka wewnętrznego jest konkluzją pewnej reguły, której przesłanki są etykietami jego dzieci.

Przykład 5.2 Niech E bedzie zbiorem rownań definiujących kraty (Definicja 4.1). Wówczas $E \vdash_\forall x = x \sqcap x$ (czyli $x \sim_E x \sqcap x$), a wyprowadzenie tej równości to takie drzewo:



Zamiast rysować krawędzie drzewa, zwykle piszemy poziome kreski tam, gdzie w wyprowadzeniu zastosowano reguły wnioskowania. A więc nasze wyprowadzenie zapiszemy tak:

$$\cfrac{\cfrac{\cfrac{x \sqcap (x \sqcup y) = x}{x \sqcap (x \sqcup (x \sqcap x)) = x}}{x = x \sqcap (x \sqcup (x \sqcap x))} \qquad \cfrac{x = x \qquad \cfrac{x \sqcup (x \sqcap y) = x}{x \sqcup (x \sqcap x) = x}}{x \sqcap (x \sqcup (x \sqcap x)) = x \sqcap x}}{x = x \sqcap x}$$

Pozostawiamy czytelnikowi przyjemność ustalenia, jakie reguły zostały tu użyte.

Twierdzenie 5.3 (o poprawności) *Jeśli* $t \sim_E u$ *to* $\mathrm{Mod}(E) \models t = u$ *(tj.* $\mathcal{A} \models t = u$ *zachodzi zawsze gdy* $\mathcal{A} \models E$*).*

Jeśli umówimy się, że będziemy pisać $E \models_\forall t = u$, gdy $\mathrm{Mod}(E) \models t = u$, to powyższe twierdzenie przyjmuje następującą postać:

$$\textit{Jeśli } E \vdash_{\forall} t = u \textit{ to } E \models_{\forall} t = u.$$

Należy je rozumieć tak: wszystko to co można wyprowadzić z aksjomatów równościowych E jest prawdą wszędzie tam, gdzie prawdziwe są te aksjomaty. A więc wnioskowanie równościowe jest *poprawne*.

Dowód: Dowód twierdzenia o poprawności przebiega przez indukcję *ze względu na definicję* relacji $\sim_E$, co w istocie oznacza indukcję ze względu na rozmiar wyprowadzenia równania „$t = u$" (liczbę wierzchołków drzewa).

Krok bazowy to obserwacja, że wszystkie aksjomaty są prawdziwe w klasie $\mathrm{Mod}(E)$. Krok indukcyjny polega na sprawdzeniu, że konkluzja każdej reguły jest prawdziwa w $\mathrm{Mod}(E)$, pod warunkiem, że prawdziwe są przesłanki. Na przykład, jeśli $\mathrm{Mod}(E) \models t = u$, to także $\mathrm{Mod}(E) \models S(t) = S(u)$, dla dowolnego podstawienia S. Szczegóły pozostawiamy jako ćwiczenie. ∎

Pełność wnioskowania równościowego

Nastepująca obserwacja wynika wprost z definicji $\sim_E$.

Fakt 5.4 *Relacja $\sim_E$ jest kongruencją w algebrze termów $\mathcal{T}_\Sigma$. (Jest to najmniejsza kongruencja w $\mathcal{T}_\Sigma$, taka że $t \sim_E s$ dla wszystkich równań $(t = s) \in E$, która jest zamknięta ze względu na podstawienia.)*

Skoro mamy kongruencję, to mamy algebrę ilorazową $\mathcal{T}_\Sigma/_{\sim_E}$, którą dla uproszczenia oznaczamy przez $\mathcal{T}_\Sigma/_E$. Podobnie, zamiast $[t]_{\sim_E}$ piszemy $[t]_E$.

Lemat 5.5 *Jeśli S jest podstawieniem, a v jest wartościowaniem w $\mathcal{T}_\Sigma/_E$ spełniającym warunek $v(x) = [S(x)]_E$ dla dowolnej zmiennej x, to dla dowolnego termu t mamy także $v(t) = [S(t)]_E$.*

Dowód: Niech $\kappa(t) = [t]_E$. Wtedy wartościowania v i $\kappa \circ S$ są homomorfizmami z $\mathcal{T}_\Sigma$ do $\mathcal{T}_\Sigma/_E$, które pokrywają się na zmiennych, tj. na generatorach. Zatem teza wynika z Lematu 1.18. ∎

Lemat 5.6 $\mathcal{T}_\Sigma/_E \models E$

Dowód: Niech „$\ell = r$" będzie równaniem z E i niech v będzie dowolnym wartościowaniem w $\mathcal{T}/_E$. Wybierzmy podstawienie S tak, aby dla dowolnej zmiennej x zachodził warunek $S(x) \in v(x)$, czyli inaczej $v(x) = [S(x)]_E$. Na mocy Lematu 1.18, wartościowanie v jest identyczne ze złożeniem S i homomorfizmu kanonicznego, więc $v(t) = [S(t)]_E$ zachodzi dla dowolnego termu t.

Ponieważ $E \vdash_\forall \ell = r$ więc także $E \vdash_\forall S(\ell) = S(r)$, a stąd także $v(\ell) = v(r)$, czyli $\mathcal{T}/_E, v \models \ell = r$. I o to chodzi. ∎

Morał 5.7 Każda teoria równościowa E ma model. Jest nim algebra ilorazowa $\mathcal{T}/_E$. Czasami jednak ten model jest trywialny, tj. jednoelementowy. Dzieje się tak, gdy $E \vdash_\forall x = y$ dla dwóch różnych zmiennych x, y. Wtedy nasza teoria równościowa ma *tylko* jednoelementowe modele (dlaczego?).

Twierdzenie 5.8 (Birkhoffa o pełności) *Następujące warunki są równoważne:*

1) $E \models_\forall t = u$;

2) $\mathcal{T}_\Sigma/_E \models t = u$;

3) $E \vdash_\forall t = u$.

Dowód: (1) ⇒ (2): Ponieważ $\mathcal{T}_\Sigma/_E \in \mathrm{Mod}(E)$ na mocy Lematu 5.6, więc $\mathcal{T}_\Sigma/_E \models t = u$.

(2) ⇒ (3): Niech $v(x) = [x]_E$. Ponieważ $\mathcal{T}_\Sigma/_E \models t = u$, więc w szczególności $v(t) = v(u)$. Korzystając z Lematu 5.5, otrzymujemy, że $[t]_E = v(t) = v(u) = [u]_E$. A więc $t \sim_E u$, czyli $E \vdash_\forall t = u$.

(3) ⇒ (1): Twierdzenie 5.3 o poprawności. ∎

Przepisywanie termów

Zbiór aksjomatów równościowych E, czytany z lewej do prawej, można uważać za zbiór reguł przekształcania termów. Ściślej, zbiór E określa pewną relację $\to_E$ w zbiorze termów.

Definicja 5.9 Relacja $\to_E$ to najmniejsza relacja w zbiorze termów spełniająca następujące warunki:

1) Jeśli „$\ell = r$” jest w E to $\ell \to_E r$;

2) Jeśli $t \to_E t'$ to $S(t) \to_E S(t')$ (tj. relacja $\to_E$ jest zamknięta ze względu na podstawienia);

3) Jeśli $t_i \to_E t'_i$, dla pewnego $i \leq n$, to $f(t_1, \ldots, t_n) \to_E f(t_1, \ldots, t_{i-1}, t'_i, t_{i+1}, \ldots, t_n)$ (tj. relacja $\to_E$ jest zamknięta ze względu na konteksty).

Relacja $\to_E$ jest to relacja *jednokrokowego przepisywania* wyznaczona przez reguły z E. Nietrudno zauważyć, że $t \to_E u$ zachodzi wtedy i tylko wtedy, gdy dla pewnego równania „$\ell = r$" ze zbioru E i dla pewnego podstawienia S:

- term t zawiera podterm postaci $S(\ell)$, oraz
- term u powstaje z t przez wymianę tego podtermu na term $S(r)$.

Na przykład, jeśli do E należy równanie $(\mathsf{K} \cdot x) \cdot y = x$, a term $(\mathsf{K} \cdot ((\mathsf{K} \cdot x) \cdot y)) \cdot z$ oznaczymy przez t, to możemy napisać zarówno $t \to_E (\mathsf{K} \cdot x) \cdot y$ jak też $t \to_E (\mathsf{K} \cdot x) \cdot z$.

Definicja 5.10 Przez $\twoheadrightarrow_E$ oznaczamy najmniejszą relację zwrotną i przechodnią zawierającą $\to_E$ (domknięcie przechodnie sumy relacji $\to_E$ i relacji identycznościowej). Symbol $\twoheadleftarrow\!\!\twoheadrightarrow_E$ oznacza najmniejszą relację równoważności zawierającą $\to_E$.

Lemat 5.11 *Warunek $t \twoheadleftarrow\!\!\twoheadrightarrow_E u$ zachodzi wtedy i tylko wtedy, gdy istnieje skończony ciąg termów $t = t_0, t_1, \ldots, t_n = u$, o tej własnosci, że dla każdego $i \leq n-1$, albo $t_i \to_E t_{i+1}$ albo $t_{i+1} \to_E t_i$.*

Dowód: Ćwiczenie. ∎

Uwaga*: Mówimy, że relacja $\to_E$ ma *własność Churcha-Rossera*, gdy $t \twoheadleftarrow\!\!\twoheadrightarrow_E u$ implikuje $t \twoheadrightarrow_E s \,{}_E\!\!\twoheadleftarrow u$, dla pewnego u. Nie każdy system przepisywania termów ma tę własność. Na przykład taki: $f(x) \Rightarrow c$, $f(x) \Rightarrow d$, gdzie c i d to dwie różne stałe. A oto mniej oczywisty przykład (tutaj $ar(D) = 2$, $ar(C) = 1$ i $ar(a) = ar(e) = 0$).

- $D(x, x) \Rightarrow e$;
- $C(x) \Rightarrow D(x, C(x))$;
- $a \Rightarrow C(a)$.

Wtedy $C(a) \to_E D(a, C(a)) \to_E D(C(a), C(a)) \to_E e$, ale także $C(a) \to_E C(C(a)) \to_E C(D(a, C(a))) \to_E C(D(C(a), C(a))) \to_E C(e) \to_E D(e, C(e)) \to_E D(e, D(e, C(e))) \to_E \cdots$

Twierdzenie 5.12 *Relacje $\sim_E$ i $\twoheadleftarrow\!\!\twoheadrightarrow_E$ są identyczne.*

Dowód: Zauważmy, że relacja $\sim_E$ spełnia warunki, o których mowa w Definicji 5.9, a więc $\to_E \;\subseteq\; \sim_E$, bo $\to_E$ jest najmniejszą relacją o tych własnościach. Ponadto $\sim_E$ jest relacją równoważności, a najmniejszą relacją równoważności zawierającą $\to_E$ jest $\twoheadleftarrow\!\!\twoheadrightarrow_E$. Stąd relacja $\twoheadleftarrow\!\!\twoheadrightarrow_E$ zawiera się w relacji $\sim_E$.

Aby udowodnić inkluzję w przeciwną stronę, na mocy Faktu 5.4, wystarczy pokazać, że relacja $\twoheadleftarrow\!\!\twoheadrightarrow_E$ jest kongruencją w $\mathcal{T}_\Sigma$, zamkniętą ze względu na podstawienia. To znaczy, że wystarczy udowodnić następujące implikacje:

a) Jeżeli $t \twoheadleftarrow\!\!\twoheadrightarrow_E u$ to $S(t) \twoheadleftarrow\!\!\twoheadrightarrow_E S(u)$.

b) Jeżeli $t_i \twoheadleftarrow\!\!\twoheadrightarrow_E t'_i$, dla $i = 1, \ldots, n$, to $f(t_1, \ldots, t_n) \twoheadleftarrow\!\!\twoheadrightarrow_E f(t'_1, \ldots, t'_n)$.

Dla dowodu warunku (a) użyjemy Lematu 5.11. Niech $t = t_0, t_1, \ldots, t_n = u$, gdzie dla każdego $i = 0, \ldots, n-1$, albo $t_i \to_E t_{i+1}$ albo $t_{i+1} \to_E t_i$. Wtedy oczywiście także $S(t_i) \to_E S(t_{i+1})$ albo $S(t_{i+1}) \to_E S(t_i)$, i w ten sposób dostajemy $S(t) \twoheadleftarrow\!\!\twoheadrightarrow_E S(u)$.

Aby pokazać warunek (b), załóżmy, że dla dowolnego $i = 1, \ldots, n$ istnieje taki ciąg $t_i = t_i^0, t_i^1, t_i^2, \ldots, t_i^{k_i} = t'_i$, w którym $t_i^j \to_E t_i^{j+1}$ lub $t_i^{j+1} \to_E t_i^j$ zachodzi dla wszystkich $j = 0, \ldots, k_i - 1$. Wtedy możemy skonstruować ciąg skończony

$$\begin{aligned} f(t_1, \ldots, t_n) = f(t_1^0, t_2, \ldots, t_n), f(t_1^1, t_2, \ldots, t_n), \ldots \\ \ldots f(t_1^{k_1}, t_2, t_3, \ldots, t_n) = f(t'_1, t_2^0, t_3, \ldots, t_n), f(t'_1, t_2^1, t_3, \ldots, t_n), \ldots \\ \ldots f(t'_1, \ldots, t'_{n-1}, t_n^{k_n}) = f(t'_1, \ldots, t'_n), \end{aligned}$$

w którym kolejne wyrazy pozostają w relacji $\to_E$ lub relacji odwrotnej. ∎

Bezpośrednio z Twierdzenia 5.12 i twierdzenia Birkhoffa o pełności otrzymujemy:

Wniosek 5.13 *Następujące warunki są równoważne:*

1) $E \models_\forall t = u$;

2) $\mathcal{T}_\Sigma/_E \models t = u$;

3) $E \vdash_\forall t = u$;

4) $t \twoheadleftarrow\!\!\twoheadrightarrow_E u$.

Morał 5.14 Jeśli w każdej algebrze z klasy $\mathrm{Mod}(E)$ prawdziwe jest równanie „$t = u$”, to można faktycznie przepisać term t w term u, używając równań ze zbioru E jako (obustronnych) reguł przepisywania.

Algebry wolne

Definicja 5.15 Klasa algebr $\mathfrak{K}$ jest *trywialna* wtedy i tylko wtedy, gdy wszystkie algebry z tej klasy są jednoelementowe. W przeciwnym razie klasa $\mathfrak{K}$ jest *nietrywialna*.

Ponieważ algebry jednoelementowe to dokładnie te algebry, w których prawdziwe jest równanie „$x_1 = x_2$”, więc mamy nastepujący oczywisty fakt:

Fakt 5.16 *Klasa $\mathfrak{K}$ jest trywialna wtedy i tylko wtedy, gdy $\mathfrak{K} \models x_1 = x_2$.*

Od tej pory zakładamy, że klasy, o których mowa, są nietrywialne.

Definicja 5.17 Algebra $\mathcal{A} \in \mathfrak{K}$ jest *algebrą wolną* w klasie $\mathfrak{K}$, o zbiorze *wolnych generatorów* X wtedy i tylko wtedy, gdy zbiór $X \subseteq |\mathcal{A}|$ generuje algebrę $\mathcal{A}$, oraz dla dowolnej algebry $\mathcal{B} \in \mathfrak{K}$, każde przekształcenie $\zeta : X \to |\mathcal{B}|$ rozszerza się (jednoznacznie) do homomorfizmu $\overline{\zeta} : \mathcal{A} \to \mathcal{B}$.

Algebrę wolną w klasie $\mathfrak{K}$ o pustym zbiorze wolnych generatorów nazywamy algebrą *początkową* w $\mathfrak{K}$.

Uwaga: Słowo „ jednoznacznie” w Definicji 5.17 można pominąć, bo na mocy Lematu 1.18 funkcję określoną na generatorach można rozszerzyć do homomorfizmu co najwyżej na jeden sposób.

Zauważmy też, że jeśli $\mathcal{A}$ jest początkowa w $\mathfrak{K}$, to dla dowolnej algebry $\mathcal{B} \in \mathfrak{K}$ istnieje dokładnie jeden homomorfizm z $\mathcal{A}$ do $\mathcal{B}$.

Przykład 5.18

- Algebra słów $\langle \{a, b\}^*, \cdot, \varepsilon \rangle$ jest algebrą wolną w klasie wszystkich półgrup (półgrupą wolną), o dwóch wolnych generatorach a i b.
- Algebra $\langle \mathbb{N}, \cdot, 1 \rangle$ nie jest półgrupą wolną, jakkolwiek by nie wybierać zbioru generatorów X. Jeśli bowiem $n, m \in X$ oraz przyjmiemy $\zeta(m) = aa$ i $\zeta(n) = bb$, to poszukiwany przez nas homomorfizm $\overline{\zeta}$ z $\mathbb{N}$ do $\{a, b\}^*$ musiałby spełniać warunek $aabb = \overline{\zeta}(m \cdot n) = \overline{\zeta}(n \cdot m) = bbaa$. Pozostaje więc tylko taka możliwość, że X jest jednoelementowy. Ale oczywiście nasza algebra nie jest generowana przez żaden pojedynczy element.
- Algebra $\langle \mathbb{N}, \mathbf{s}, 0 \rangle$, gdzie $\mathbf{s}$ jest funkcją następnika ($\mathbf{s}(n) = n + 1$), jest algebrą początkową w klasie wszystkich algebr swojej sygnatury.
- Algebra termów $\mathcal{T}_\Sigma$ jest wolna w klasie wszystkich algebr, a zbiorem generatorów jest zbiór wszystkich zmiennych.

Fakt 5.19 *Algebry wolne w tej samej klasie, o zbiorach wolnych generatorów tej samej mocy, są izomorficzne.*

Dowód: Niech X i Y będą zbiorami wolnych generatorów algebr wolnych $\mathcal{A}$ i $\mathcal{B}$, i niech $\zeta : X \xrightarrow[\text{na}]{1-1} Y$. Wtedy ζ rozszerza się do homomorfizmu $\overline{\zeta} : \mathcal{A} \to \mathcal{B}$. Ale przekształcenie odwrotne ζ^{-1} też rozszerza się do homomorfizmu $\overline{\zeta^{-1}} : \mathcal{B} \to \mathcal{A}$. Złożenia $\overline{\zeta} \circ \overline{\zeta^{-1}} : \mathcal{B} \to \mathcal{B}$ oraz $\overline{\zeta^{-1}} \circ \overline{\zeta} : \mathcal{A} \to \mathcal{A}$ pokrywają się z przekształceniami identycznościowymi na zbiorach generatorów, zatem na mocy Lematu 1.18 są po prostu wzajemnie odwrotne. Oznacza to, że są izomorfizmami. ∎

Fakt 5.20 *Załóżmy, że $\mathcal{A}$ jest algebrą wolną w klasie $\mathfrak{K}$ o co najmniej k wolnych generatorach. Jeśli $\mathcal{A} \models t = s$ i w równaniu „$t = s$" występuje tylko k zmiennych, to $\mathfrak{K} \models t = s$.*

Dowód: Przyjmijmy, że w równaniu „$t = s$" występują tylko zmienne $x_1, \ldots, x_k$. Niech $\mathcal{B} \in \mathfrak{K}$ i niech v będzie wartościowaniem w $\mathcal{B}$. Homomorfizm v można przedstawić jako złożenie $v = h \circ w$ gdzie $w(x_1), \ldots, w(x_k)$ są wolnymi generatorami algebry $\mathcal{A}$, a h jest homomorfizmem rozszerzającym przyporządkowanie $w(x_i) \mapsto v(x_i)$, dla $i = 1, \ldots, k$. Skoro $w(t) = w(s)$, to także $v(t) = v(s)$. A więc $\mathcal{B}, v \models t = s$, dla dowolnego v. ∎

Fakt 5.21 *Algebra $\mathcal{T}_\Sigma/_E$ jest algebrą wolną w klasie* $\mathrm{Mod}(E)$*, o zbiorze wolnych generatorów mocy $\aleph_0$. (A zatem jest też algebrą wolną w każdej podklasie $\mathfrak{K} \subseteq \mathrm{Mod}(E)$, do której sama należy. Na przykład, gdy $E = \mathrm{Eq}(\mathfrak{K})$.)*

Dowód: Łatwo zgadnąć, że zbiór $X = \{[x]_E \mid x \in V\}$ powinien być zbiorem wolnych generatorów. Aby to sprawdzić, przypuśćmy, że $\zeta : X \to |\mathcal{B}|$ dla pewnego $\mathcal{B} \in \mathrm{Mod}(E)$. Niech v będzie wartościowaniem w algebrze $\mathcal{B}$ zadanym przez warunek $v(x) = \zeta([x]_E)$. Homomorfizm $\overline{\zeta}$ definiujemy tak:

$$\overline{\zeta}([t]_E) = v(t).$$

Definicja jest poprawna, bo dla $t \sim_E t'$ zachodzi $\mathcal{B} \models t = t'$ więc $v(t) = v(t')$. Nietrudno zaś sprawdzić, że $\overline{\zeta}$ jest homomorfizmem. ∎

Oczywiście zbiór generatorów algebry $\mathcal{T}_\Sigma/_E$ jest mocy $\aleph_0$ tylko dlatego, że przyjęliśmy założenie o przeliczalności zbioru zmiennych. Można się jednak umówić inaczej.

Fakt 5.22 *Jeśli zbiór zmiennych V jest mocy $\mathfrak{m}$, oraz $\mathcal{T}_\Sigma/_E \in \mathfrak{K} \subseteq \mathrm{Mod}(E)$, to $\mathcal{T}_\Sigma/_E$ jest algebrą wolną w klasie $\mathfrak{K}$, mającą $\mathfrak{m}$ wolnych generatorów. Podobnie, algebra $\mathcal{T}_\Sigma(n)/_E$ termów n-argumentowych jest algebrą wolną o n wolnych generatorach.*

Dowód: Identyczny z dowodem Faktu 5.21 ∎

Twierdzenie Birkhoffa

Zacznijmy od nastepującej prostej obserwacji:

Fakt 5.23 *Jeśli klasa $\mathfrak{K}$ jest definiowalna równościowo, to jest zamknięta ze względu na podalgebry, obrazy homomorficzne i produkty, tj. zachodzą inkluzje $H(\mathfrak{K}) \subseteq \mathfrak{K}$, $S(\mathfrak{K}) \subseteq \mathfrak{K}$ oraz $P(\mathfrak{K}) \subseteq \mathfrak{K}$.*

Dowód: Ćwiczenie. ∎

Znacznie mniej oczywiste jest to, że zachodzi także twierdzenie odwrotne do Faktu 5.23. Zamkniętość ze względu na wymienione operacje jest warunkiem wystarczającym na to, aby dana klasa była rozmaitością. Z Lematu 3.2 wynika, że w istocie chodzi tu o równość $HSP(\mathfrak{K}) = \mathfrak{K}$. Otrzymujemy w ten sposób czysto semantyczną charakteryzację klas definiowalnych równościowo, zwaną twierdzeniem Birkhoffa.

Twierdzenie 5.24 (Twierdzenie Birkhoffa) *Następujące warunki są równoważne:*

1) Klasa $\mathfrak{K}$ jest definiowalna równościowo;

2) Klasa $\mathfrak{K}$ jest zamknięta ze względu na podalgebry, obrazy homomorficzne i produkty;

3) $HSP(\mathfrak{K}) = \mathfrak{K}$.

Implikacja (1) $\Rightarrow$ (2) stanowi treść Faktu 5.23, a równoważność warunków (2) i (3) jest łatwym wnioskiem z Lematu 3.2. Pozostaje przed nami dowód najtrudniejszej implikacji (2) $\Rightarrow$ (1).

Lemat 5.25 *Załóżmy, że $\mathfrak{K}$ jest nietrywialną klasą spełniającą warunek $HSP(\mathfrak{K}) = \mathfrak{K}$. Wtedy klasa $\mathfrak{K}$ ma algebry wolne o dowolnej niezerowej (skończonej lub nieskończonej) liczbie wolnych generatorów*

Dowód: Pokażemy jak skonstruować algebrę wolną o zbiorze wolnych generatorów mocy $\aleph_0$. Dla $\mathfrak{m} \neq \aleph_0$ dowód wymaga rozważania termów nad zbiorem zmiennych mocy $\mathfrak{m}$, ale przebiega w podobny sposób.

Niech $E = \mathrm{Eq}(\mathfrak{K})$. Rozpatrzmy następujący zbiór relacji:

$$\mathcal{Z} = \{\rho \mid \rho \text{ jest kongruencją w } \mathcal{T}_\Sigma \text{ oraz } \mathcal{T}_\Sigma/_\rho \in \mathfrak{K}\}.$$

Na początek zauważmy, że $\mathcal{Z} \neq \emptyset$. W tym celu weźmy dowolne wartosciowanie v w jakiejś algebrze $\mathcal{A} \in \mathfrak{K}$. Jest to oczywiście homomorfizm, a jego zbiór wartości jest podalgebrą w $\mathcal{A}$, izomorficzną z algebrą ilorazową $\mathcal{T}_\Sigma/_{\ker(v)}$ (Twierdzenie 1.27). Ponieważ klasa $\mathfrak{K}$

jest zamknięta ze względu na podalgebry, więc iloraz $\mathcal{T}_\Sigma/_{\ker(v)}$ należy do $\mathfrak{K}$ a jądro $\ker(v)$ jest elementem zbioru $\mathcal{Z}$.

Skoro więc rodzina $\mathcal{Z}$ jest niepusta, to jej iloczyn jest kongruencją w algebrze termów. Oznaczmy ten iloczyn przez $\approx$.

Niech teraz $\mathcal{B} = \prod_{\rho \in \mathcal{Z}} \mathcal{T}_\Sigma/_\rho$. Algebra $\mathcal{B}$ należy do klasy $\mathfrak{K}$, bo $\mathfrak{K}$ jest zamknięta ze względu na produkty. Zdefiniujemy przekształcenie $h : \mathcal{T}_\Sigma/_\approx \to \mathcal{B}$ warunkiem

$$h([t]_\approx)(\rho) = [t]_\rho,$$

dla dowolnego $\rho \in \mathcal{Z}$. To przekształcenie jest monomorfizmem (różnowartościowym homomorfizmem). W rzeczy samej, jeśli klasy $[t]_\approx$ i $[t']_\approx$ są różne, to $\langle t, t' \rangle \notin \rho$ przynajmniej dla jednej relacji $\rho \in \mathcal{Z}$. Jako ćwiczenie pozostawiamy sprawdzenie, że h jest dobrze określone i że zachowuje operacje.

A zatem algebra ilorazowa $\mathcal{T}_\Sigma/_\approx$ jest izomorficzna z pewną podalgebrą algebry $\mathcal{B}$ i sama też należy do $\mathfrak{K}$. Inaczej mówiąc, relacja $\approx$ jest (najmniejszym) elementem rodziny $\mathcal{Z}$.

Relacja $\approx$ jest w istocie identyczna z relacją $\sim_E$. Pokażemy najpierw inkluzję $\sim_E \subseteq \approx$.

Niech $t \sim_E u$ czyli $E \vdash_\forall t = u$. Stąd $\mathfrak{K} \models t = u$, bo $\mathfrak{K} \models E$. Jeśli więc $\rho \in \mathcal{Z}$ to także $\mathcal{T}_\Sigma/_\rho \models t = u$, w szczególności $\mathcal{T}_\Sigma/_\rho, \kappa \models t = u$, gdzie κ jest wartościowaniem kanonicznym: $\kappa(s) = [s]_\rho$ dla $s \in \mathcal{T}_\Sigma$. Ale to oznacza, że $[t]_\rho = [u]_\rho$ czyli $t \ \rho \ u$. Tak jest dla wszystkich $\rho \in \mathcal{Z}$, a więc $t \approx u$.

Przypuśćmy teraz, że $t \approx u$. Dla wykazania, że $t \sim_E u$ wystarczy wiedzieć, że $\mathfrak{K} \models t = u$, bo wtedy „$t = u$" $\in E$. Ale jeśli v jest wartościowaniem w jakiejś algebrze $\mathcal{B} \in \mathfrak{K}$, to $\ker(v) \in \mathcal{Z}$, więc skoro $t \approx u$ to też $(t, u) \in \ker(v)$ czyli $\mathcal{B}, v \models t = u$.

Konkluzja jest taka: Ponieważ $\approx$ to to samo co $\sim_E$, więc $\mathcal{T}_\Sigma/_E = \mathcal{T}_\Sigma/_\approx \in \mathfrak{K}$. Na mocy Faktu 5.21 mamy poszukiwaną algebrę wolną. ∎

Dowód twierdzenia Birkhoffa: Niech $E = \mathrm{Eq}(\mathfrak{K})$. Pokażemy, że $\mathfrak{K} = \mathrm{Mod}(E)$. Inkluzja z lewej do prawej jest oczywista. Niech więc $\mathcal{B} \in \mathrm{Mod}(E)$ będzie algebrą o mocy $\mathfrak{m}$. W klasie $\mathfrak{K}$ istnieje algebra wolna $\mathcal{A}$, o zbiorze wolnych generatorów G mocy $\mathfrak{m}$. Dowolna bijekcja $h : G \xrightarrow[\text{na}]{1-1} |\mathcal{B}|$ rozszerza się do homomorfizmu, który jest oczywiście na $|\mathcal{B}|$. A więc nasza algebra jest obrazem homomorficznym algebry $\mathcal{A}$. Klasa $\mathfrak{K}$ jest zamknięta ze względu na obrazy homomorficzne, a stąd $\mathcal{B} \in \mathfrak{K}$. ∎

6 Unifikacja

Definicja 6.1 Przez *układ równań* rozumiemy skończony ciąg równań.[5]

Rozwiązaniem równania „$t = u$” nazywamy dowolne podstawienie S, spełniające warunek $S(t) = S(u)$. Mówimy wtedy też, że podstawienie S jest *unifikatorem* termów t i s, lub że je *unifikuje*. *Rozwiązanie* układu równań to takie podstawienie, które jest rozwiązaniem wszystkich równań tego układu.

Przykład 6.2 Załóżmy, że $ar(f) = ar(g) = 2$ oraz $ar(c) = ar(d) = 0$.

- Rozwiązaniem równania $f(g(x, y), x) = f(z, g(y, c))$ jest na przykład podstawienie $S(x) = g(y, c)$, $S(y) = y$, $S(z) = g(g(y, c), y)$.
- Równanie $f(g(y, c), y) = f(x, g(x, d))$ nie ma rozwiązania. Istotnie, gdyby S było takim rozwiązaniem, oraz $S(x) = t$, to term t musiałby być postaci $g(g(t, d), c)$, czyli byłby dłuższy sam od siebie.
- Równanie $f(x, f(y, c)) = f(g(y, c), x)$ też nie ma rozwiązania, ale z innego powodu. Gdyby S było rozwiązaniem, oraz $S(y) = t$, to termy $g(t, c)$ i $f(t, c)$ musiałyby być identyczne, a nie zgadzają się ich symbole początkowe.

Zajmiemy się teraz algorytmicznym rozwiązaniem *problemu unifikacji*, polegającego na ustaleniu, czy dany układ równań ma rozwiązanie i jakie. Algorytm polega na upraszczaniu danego układu równań i jednoczesnym konstruowaniu rozwiązania. W każdej fazie pracy mamy więc do czynienia z parą postaci $\langle E, S\rangle$, gdzie E jest układem równań a S jest podstawieniem. Rozpoczynamy od pary $\langle E, \mathrm{id}\rangle$, a naszym celem jest uzyskanie pary postaci $\langle \emptyset, S\rangle$, gdzie S jest poszukiwanym rozwiązaniem. Nasz algorytm polega na iterowaniu operacji Θ, którą zaraz zdefiniujemy.

Przyjmujemy taką notację: Przez $(t = u); E$ (odpowiednio: $E; (t = u)$) oznaczymy układ powstający z E przez dodanie na początku (odpowiednio: na końcu) równania „$t = u$”. Notacja $E[x := t]$ oznacza układ powstały z E przez zastosowanie podstawienia $[x := t]$ do obu stron wszystkich równań. Podobnie uogólniamy inne oznaczenia, np. $FV(E)$ oznacza zbiór wszystkich zmiennych występujących w układzie E.

Definicja 6.3 Niech $\langle E, S\rangle$ będzie taką parą, że $E \neq \emptyset$. Jeśli się uda, to określimy nową parę $\Theta(E, S) = \langle E', S'\rangle$. Mamy kilka przypadków, w zależności od pierwszego równania układu E.

Przypadek 1: Układ E jest postaci $(x = t); F$, przy tym $x \notin FV(t)$. Wówczas $E' = F[x := t]$ oraz $S' = [x := t] \circ S$, tj. $S'(u) = S(u)[x := t]$ dla dowolnego termu u.

[5] Często przyjmuje się, że układ równań to po prostu zbiór równań. Nam wygodniej przyjąć, że równania w układzie są ponumerowane.

Przypadek 2: Układ E jest postaci $(t = x); F$, gdzie t nie jest zmienną i $x \notin FV(t)$. Para $\langle E', S'\rangle$ jest określona tak samo jak w przypadku 1.

Przypadek 3: Układ E jest postaci $(x = x); F$. Wtedy $E' = F$ oraz $S' = S$.

Przypadek 4: Układ E jest postaci $(f(t_1, \ldots, t_k) = f(u_1, \ldots, u_k)); F$, gdzie $ar(f) = k$. Wtedy $E' = F; (t_1 = u_1); \ldots; (t_k = u_k)$ oraz $S' = S$.

Przypadek 5: Układ E jest postaci $(x = t); F$ lub postaci $(t = x); F$, gdzie t nie jest zmienną i $x \in FV(t)$. Wtedy $\Theta(E, S)$ jest nieokreślone.

Przypadek 6: Układ E jest postaci $(f(t_1, \ldots, t_k) = g(u_1, \ldots, u_l)); F$, gdzie $f \neq g$. Wtedy $\Theta(E, S)$ też jest nieokreślone.

Łatwo sprawdzić, że przypadki 1–6 wyczerpują wszystkie możliwości, i że w przypadkach 5 i 6 pierwsze równanie (a zatem cały układ) nie ma rozwiązania.

Twierdzenie 6.4 (Terminacja algorytmu) *Dla dowolnych E i S istnieje takie n, że $\Theta^n(E, S)$ ma postać $\langle \emptyset, P\rangle$ lub jest nieokreślone.*

Dowód: Dla dowolnego E, przez $\#(E)$ oznaczymy parę liczb $\langle p, q\rangle$, gdzie:

- p to liczba zmiennych występujących w E;
- q to suma długości wszystkich termów w E.

Przypuśćmy, że $\Theta(E, S) = \langle E', S'\rangle$ i niech $\#(E) = \langle p, q\rangle$, oraz $\#(E') = \langle p', q'\rangle$. W przypadkach 1 i 2 mamy $p' < p$, a w przypadkach 3 i 4 zachodzi $p' \leq p$ i $q' < q$. A więc $\#(E') \prec \#(E)$ w porządku leksykograficznym. Ponieważ jest to dobry porządek, więc iteracja operacji Θ nie może się ciągnąć w nieskończoność. ∎

Trzeba jeszcze udowodnić, że nasz algorytm faktycznie robi to, czego od niego oczekujemy. Ustalmy pewien układ E i przyjmijmy oznaczenie $\langle E_n, S_n\rangle = \Theta^n(E, \mathrm{id})$. W szczególności mamy $E = E_0$ i $S = \mathrm{id}$. Na początek kilka łatwych obserwacji.

Lemat 6.5

1. *Jeśli $z \in FV(E_n)$ to $S_n(z) = z$.*
2. *Jeśli $z \in FV(S_n(y))$, to $S_n(z) = z$. Zatem $S_n \circ S_n = S_n$.*
3. *Jeśli R jest rozwiązaniem E_n to jest też rozwiązaniem E_{n+1}.*
4. *Jeśli R jest takim podstawieniem, że $R(x) = R(t)$, to $R = R \circ [x := t]$.*
5. *Jeśli R jest rozwiązaniem E_n oraz $R = R \circ S_n$ to $R = R \circ S_{n+1}$.*

Dowód: Dowód części (4) polega na prostym sprawdzeniu. Pozostałych części dowodzimy przez łatwą indukcję ze względu na n. Oczywiście mamy każdorazowo tyle przypadków ile ich jest w Definicji 6.3. Dla przykładu rozpatrzmy przypadek 1 w części (5). Wtedy $S_{n+1} = [x := t] \circ S_n$. Ponieważ R jest rozwiązaniem E_n więc $R(x) = R(t)$. Z założenia indukcyjnego i części (4) mamy $R = R \circ S_n = R \circ [x := t] \circ S_n = R \circ S_{n+1}$. ∎

Twierdzenie 6.6 (Poprawność algorytmu)

1) Jeśli $\Theta^m(E, \mathrm{id})$ jest dla pewnego m nieokreślone, to układ E nie ma rozwiązania.

2) Jeśli $\Theta^m(E, \mathrm{id}) = \langle \emptyset, R \rangle$, to R jest rozwiązaniem układu E.

Dowód: Część (1) wynika dość natychmiastowo z Lematu 6.5(3). Jeśli bowiem $\Theta^m(E, \mathrm{id})$ nie jest określone, ale $\Theta^{m-1}(E, \mathrm{id}) = \langle E_{m-1}, S_{m-1} \rangle$ jeszcze jest określone, to znaczy, że układ E_{m-1} nie ma rozwiązania. Ale z Lematu 6.5(3) wynika, że wtedy także samo E nie ma rozwiązania. Zajmijmy się więc częścią (2).

Przez indukcję ze względu na różnicę $m - n$ pokażemy, że dla $n = 0, \ldots, m$:

- R jest rozwiązaniem układu E_n;
- $R = R \circ S_n$.

Teza twierdzenia wynika z przypadku $n = 0$.

Oczywiście R jest rozwiązaniem układu pustego i na dodatek $R = S_m = S_m \circ S_m$, na mocy Lematu 6.5(2). Stąd wynika teza w przypadku początkowym, gdy $n = m$.

Przypuśćmy, że R jest rozwiązaniem E_{n+1}, i że $R = R \circ S_{n+1}$.

Przypadek 1: Układ E_n jest postaci $(x = t); F$, gdzie $x \notin FV(t)$. Wtedy $E_{n+1} = F[x := t]$ oraz $S_{n+1} = [x := t] \circ S_n$.

Sprawdzamy, czy $R(x) = R(t)$. Otóż $R(x) = R \circ S_{n+1}(x) = R \circ [x := t] \circ S_n(x) = R(t)$, bo $x \in FV(E_n)$ i na mocy Lematu 6.5(1) zachodzi $S_n(x) = x$.

Sprawdzamy, czy R jest rozwiązaniem układu F. Weźmy dowolne równanie „$u = s$" z tego układu. Z założenia indukcyjnego mamy $R(u[x := t]) = R(s[x := t])$. Ponieważ już wiemy, że $R(x) = R(t)$, więc z Lematu 2.11 wnioskujemy, że $R(u) = R_x^{R(t)}(u) = R(u[x := t])$ i analogicznie $R(s) = R_x^{R(t)}(s) = R(s[x := t])$. A zatem $R(u) = R(s)$.

Pozostaje wykazać, że $R = R \circ S_n$. Wiemy jednak, że $R = R \circ S_{n+1} = R \circ [x := t] \circ S_n$. Ponadto z Lematu 6.5(4) mamy $R \circ [x := t] = R$, bo $R(x) = R(t)$. A więc $R = R \circ S_n$.

Przypadek 2: Analogiczny.

Przypadek 3 i 4: Oczywiste, bo układy E_n i E_{n+1} mają te same rozwiązania a podstawienia S_n i S_{n+1} są takie same. ∎

Treść powyższego twierdzenia można wzmocnić. Rozwiązanie znajdowane przez nasz algorytm jest tzw. rozwiązaniem *głównym*: każde inne rozwiązanie można z niego otrzymać na drodze podstawienia.

Fakt 6.7 *Niech* $\Theta^m(E, \mathrm{id}) = \langle \emptyset, R \rangle$ *i niech* R' *będzie rozwiązaniem układu* E. *Wtedy* $R' = Q \circ R$ *dla pewnego podstawienia* Q.

Dowód: W istocie jako Q można wybrać samo R'. Należy zauważyć na początku, że $R' = R' \circ \mathrm{id}$, a następnie zastosować Lemat 6.5(5). ∎

7 Język logiki predykatów

Słowo „predykat" oznacza wyrażenie opisujące pewną własność rozważanych obiektów. System logiczny, którym będziemy się zajmować, jest nazywany *logiką* (lub *rachunkiem*) *predykatów* (pierwszego rzędu), bo występują w nim symbole relacyjne, interpretowane jako własności przedmiotów. W odniesieniu do logiki predykatów używamy też określenia „rachunek kwantyfikatorów".

Od tej pory umawiamy się, że jeśli mówimy o sygnaturze Σ, to symbol równości „=" nie należy do Σ. Symbol „=" nie jest zwykłym symbolem relacyjnym, ale jest traktowany na specjalnych prawach.

Definicja 7.1 *Formuły atomowe* sygnatury Σ są następujące:

- symbol „$\bot$" (na oznaczenie fałszu);
- napisy postaci „$r(t_1, \ldots, t_n)$", gdzie $r \in \Sigma_n^R$ oraz $t_1, \ldots, t_n \in \mathcal{T}_\Sigma$;
- napisy postaci „$t_1 = t_2$", gdzie $t_1, t_2 \in \mathcal{T}_\Sigma$.

W szczególności do formuł atomowych zaliczamy zeroargumentowe symbole relacyjne. Nazywamy je *symbolami zdaniowymi* (lub *zmiennymi zdaniowymi*). Uwaga: symbol „$\bot$" nie jest zmienną zdaniową, ale *stałą logiczną*.

Definicja 7.2 *Formuły* sygnatury Σ definiujemy jako elementy najmniejszego zbioru $\mathcal{F}_\Sigma$ (ozn. też $\mathcal{F}$) spełniającego warunki:

- formuły atomowe należą do $\mathcal{F}_\Sigma$;
- jeśli $\varphi, \psi \in \mathcal{F}_\Sigma$ to także $(\varphi \to \psi) \in \mathcal{F}_\Sigma$;
- jeśli $\varphi \in \mathcal{F}_\Sigma$ i $x \in V$ (x jest zmienną indywiduową) to także $(\forall x \varphi) \in \mathcal{F}_\Sigma$.

W powyższej definicji, dla prostoty dalszych rozważań, ograniczyliśmy się do implikacji i fałszu oraz kwantyfikatora ogólnego $\forall$. Ale oczywiście na ogół posługujemy się też innymi spójnikami i kwantyfikatorem szczegółowym $\exists$ (nazywanym też kwantyfikatorem egzystencjalnym). Możemy je jednak wyrazić za pomocą pozostałych operatorów i uważać za skróty notacyjne. Przyjmujemy więc, że:

Napis	$(\neg\varphi)$	oznacza	$(\varphi \to \bot)$;
	$\top$		$(\neg\bot)$;
	$(\varphi \vee \psi)$		$((\neg\varphi) \to \psi)$;
	$(\varphi \wedge \psi)$		$(\neg(\varphi \to (\neg\psi)))$;
	$(\varphi \leftrightarrow \psi)$		$((\varphi \to \psi) \wedge (\psi \to \varphi))$;
	$(\exists x \varphi)$		$\neg(\forall x \neg\varphi)$.

Konwencje notacyjne: Niepotrzebne nawiasy pomijamy, stosując przy tym następujące priorytety:

1. Kwantyfikatory i negacja;
2. Koniunkcja i alternatywa;
3. Implikacja.

Zatem na przykład:

- Wyrażenie „$\neg\varphi \vee \psi \to \vartheta$" oznacza $(((\neg\varphi) \vee \psi) \to \vartheta)$;
- Napis „$\varphi \vee \psi \wedge \vartheta$" jest niepoprawny;
- Napis „$\forall x P(x) \to R(x)$" należy rozumieć jako $((\forall x P(x)) \to R(x))$ (a nie jako $(\forall x(P(x) \to R(x))))$.

Uwaga:* Jeśli po kwantyfikatorze stoi kropka, to oznacza ona lewy nawias, którego zasięg rozciaga się do oporu w prawo (tj. do następnego prawego nawiasu, lub do końca formuły). Na przykład „$\forall x.P(x) \to Q(x)$" oznacza „$\forall x(P(x) \to Q(x))$". My jednak nie będziemy stosować tej konwencji.

Definicja 7.3 Zbiór *zmiennych wolnych* formuły φ, oznaczany przez $FV(\varphi)$, jest określony przez indukcję:

- $FV(r(t_1, \ldots, t_n)) = FV(t_1) \cup \cdots \cup FV(t_n)$;
- $FV(t_1 = t_2) = FV(t_1) \cup FV(t_2)$;
- $FV(\bot) = \emptyset$;
- $FV(\varphi \to \psi) = FV(\varphi) \cup FV(\psi)$.
- $FV(\forall x \varphi) = FV(\varphi) - \{x\}$.

Łatwo sprawdzić, że:

$$FV(\varphi \wedge \psi) = FV(\varphi \vee \psi) = FV(\varphi) \cup FV(\psi), \qquad \text{oraz} \qquad FV(\exists x \varphi) = FV(\varphi) - \{x\}.$$

Definicja 7.4 Mówimy, że formuła jest *otwarta*, jeśli nie występują w niej kwantyfikatory. Natomiast formuła *zamknięta* (albo *zdanie*) to taka formuła φ, która nie ma zmiennych wolnych (tj. $FV(\varphi) = \emptyset$). Uwaga: niektóre formuły są jednocześnie zamknięte i otwarte!

Zmienną x, występującą w kontekście „$\forall x \varphi$" nazywamy *związaną*. (Dotyczy to też zmiennej x w kontekście „$\exists x \varphi$".)

Uwaga: ta sama zmienna może występować w formule kilkakrotnie, zarówno jako wolna jak i związana (i to różnymi kwantyfikatorami), jak np. zmienna x w formule

$$\text{„}\forall x \exists y P(x, y) \to (Q(x) \vee \exists x R(x))\text{"}. \tag{1}$$

Ponieważ napis $\forall x\, \varphi$ czytamy „dla każdego x zachodzi φ", jest intuicyjnie dość oczywiste, że np. znaczenie formuł $\forall x\, P(x, z)$ i $\forall y\, P(y, z)$ powinno być takie samo. Istotnie, zaraz się okaże, że tak jest, i dlatego wygodnie jest utożsamiać ze sobą formuły, różniące się od siebie tylko zmiennymi związanymi. Takie utożsamienie nazywamy *alfa-konwersją*. Dla porządku wprowadzimy to pojęcie w sposób bardziej ścisły.

Definicja 7.5 Jeśli w jest dowolnym napisem, to przez $w(x \wr y)$ oznaczymy napis, który otrzymujemy z w przez wymianę wszystkich wystąpień x na y i odwrotnie. Na przykład $(\forall x\, P(x, y, z))(x \wr y) = \forall y\, P(y, x, z)$.

Definicja 7.6 Relacja *alfa-konwersji* to najmniejsza relacja równoważności $=_\alpha$ w zbiorze formuł, spełniająca warunki:

- $\forall x\, \varphi =_\alpha \forall y\, \varphi(x \wr y)$, gdy $y \notin FV(\varphi)$;
- Jeśli $\varphi_1 =_\alpha \varphi_2$ to $\forall x\, \varphi_1 =_\alpha \forall x\, \varphi_2$;
- Jeśli $\varphi_1 =_\alpha \varphi_2$ oraz $\psi_1 =_\alpha \psi_2$ to $\varphi_1 \to \psi_1 =_\alpha \varphi_2 \to \psi_2$.

Formuły równoważne ze względu na alfa-konwersję to właśnie formuły „różniące się tylko zmiennymi związanymi". Od tej pory przyjmujemy następującą umowę:

Jeśli $\varphi =_\alpha \psi$ to formuły φ i ψ uważamy za identyczne.

8 Znaczenie formuł

Znaczeniem formuły jest wartość logiczna „prawda” (1) lub „fałsz” (0). Formułom sygnatury Σ możemy przypisywać znaczenia w dowolnej strukturze tej sygnatury, zależnie od wybranego wartościowania termów. Przypomnijmy, że każde wartościowanie $v : \mathcal{T}_\Sigma \to \mathcal{A}$ jest jednoznacznie wyznaczone przez swoje obcięcie $v|_V : V \to \mathcal{A}$. Dlatego zwykle mówimy o *wartościowaniu zmiennych* indywiduowych i utożsamiamy v z $v|_V$.

Jeśli v jest takim wartościowaniem w strukturze $\mathcal{A}$, oraz $a \in |\mathcal{A}|$, to przez v_x^a oznaczamy wartościowanie określone tak:

$$v_x^a(y) = \begin{cases} a, & \text{gdy } y = x; \\ v(y), & \text{w przeciwnym przypadku.} \end{cases}$$

Definicja 8.1 *Wartość* formuły φ w strukturze $\mathcal{A}$ przy wartościowaniu v oznaczamy przez $v(\varphi)$ i definiujemy przez indukcję (ze względu na budowę formuły):

- $v(\bot) = 0$;
- $v(r(t_1, \ldots, t_n)) = 1$, gdy $\langle v(t_1) \ldots, v(t_n) \rangle \in r^{\mathcal{A}}$;
- $v(r(t_1, \ldots, t_n)) = 0$, w przeciwnym przypadku;
- $v(t_1 = t_2) = 1$, gdy $v(t_1) = v(t_2)$;
- $v(t_1 = t_2) = 0$, w przeciwnym przypadku;
- $v(\varphi \to \psi) = 0$, gdy $v(\varphi) = 1$ i $v(\psi) = 0$;
- $v(\varphi \to \psi) = 1$, w przeciwnym przypadku.
- $v(\forall x \varphi) = \min\{v_x^a(\varphi) \mid a \in |\mathcal{A}|\}$.

Zamiast $v(\varphi)$ piszemy czasem $[\![\varphi]\!]_v$, a jeśli chcemy podkreślić, że chodzi o wartościowanie w strukturze $\mathcal{A}$ to piszemy $[\![\varphi]\!]_v^{\mathcal{A}}$.

Łatwo widzieć, że $v(\varphi \to \psi) = \max\{v(\psi), 1 - v(\varphi)\}$, oraz że dla pozostałych spójników:

$$\begin{aligned} v(\varphi \vee \psi) &= \max\{v(\varphi), v(\psi)\}; \\ v(\varphi \wedge \psi) &= \min\{v(\varphi), v(\psi)\}; \\ v(\neg\varphi) &= 1 - v(\varphi). \end{aligned}$$

Natomiast dla kwantyfikatora szczegółowego otrzymujemy wzór:

$$v(\exists x \varphi) = \max\{v_x^a(\varphi) : a \in |\mathcal{A}|\}.$$

Lemat 8.2 *Niech φ będzie dowolną formułą. Jeśli $u, v : V \to |\mathcal{A}|$ są takimi wartościowaniami, że $u|_{FV(\varphi)} = v|_{FV(\varphi)}$, to $u(\varphi) = v(\varphi)$.*

Dowód: Dowód przebiega przez indukcję ze względu na długość formuły. Przypadek $\varphi = \bot$ jest oczywisty. Jeśli $\varphi = r(t_1, \dots, t_n)$ to założenie $u|_{FV(\varphi)} = v|_{FV(\varphi)}$ oznacza w szczególności, że $u|_{FV(t_i)} = v|_{FV(t_i)}$ dla wszystkich t_i. Zatem na mocy Faktu 2.8 mamy $v(t_i) = u(t_i)$, a stąd łatwo wynika, że $v(r(t_1, \dots, t_n)) = u(r(t_1, \dots, t_n))$. W ten sam sposób postępujemy w przypadku gdy φ jest równaniem.

Jeśli $\varphi = \psi_1 \to \psi_2$ to z założenia indukcyjnego dla ψ_1 i ψ_2 otrzymujemy

$$u(\varphi) = \max\{u(\psi_2), 1 - u(\psi_1)\} = \max\{v(\psi_2), 1 - v(\psi_1)\} = v(\varphi).$$

Pozostaje przypadek, gdy formuła φ jest postaci $\forall x \psi$. Z równości $u|_{FV(\varphi)} = v|_{FV(\varphi)}$, wynika wtedy, że $u^a_x|_{FV(\psi)} = v^a_x|_{FV(\psi)}$, bo $FV(\psi) \subseteq FV(\varphi) \cup \{x\}$. Z założenia indukcyjnego dla ψ dostajemy więc $u^a_x(\psi) = v^a_x(\psi)$ dla dowolnego a. Zatem

$$u(\varphi) = u(\forall x \psi) = \min\{u^a_x(\psi) : a \in |\mathcal{A}|\} = \min\{v^a_x(\psi) : a \in |\mathcal{A}|\} = v(\varphi). \quad ∎$$

Morał z powyższego jest oczywiście taki: wartość formuły zależy tylko od wartości jej zmiennych wolnych. Na przykład wartość formuły (1) nie zależy od wartości y. Dlatego gdy wiemy, że $FV(\varphi) \subseteq \{x_1, \dots, x_n\}$, to czasami zamiast φ piszemy $\varphi(x_1, \dots, x_n)$, dla zaznaczenia, jakie zmienne są tu istotne.

Ponieważ umówiliśmy się, że alfa-równoważne formuły uważamy za identyczne, powinniśmy się jeszcze przekonać, że alfa-konwersja nie zmienia wartości formuły.

Lemat 8.3 *Jeśli $\varphi =_\alpha \psi$ to $v(\varphi) = v(\psi)$ przy dowolnym wartościowaniu v.*

Dowód:* Pozostawiając szczegóły najbardziej dociekliwym czytelnikom, zauważmy tylko, że wygodnie jest pokazać najpierw taką własność transpozycji zmiennych:

Jeśli $v_1(x) = v_2(y)$, $v_1(y) = v_2(x)$ oraz $v_1(z) = v_2(z)$ dla $z \neq x, y$, to $v_1(\varphi) = v_2(\varphi(x \wr y))$.

Dowodzimy tej własności przez indukcję ze względu na długość formuły φ. Wynika z niej łatwo, że $v(\forall x\, \varphi) = v(\forall y\, \varphi(x \wr y))$ dla $y \notin FV(\varphi)$. Dalej postępujemy przez indukcję ze względu na definicję relacji $\varphi =_\alpha \psi$. ∎

Definicja 8.4 Mówimy, że formuła φ jest *spełniona* w $\mathcal{A}$ przez wartościowanie v, gdy $v(\varphi) = 1$. Piszemy wtedy

$$\mathcal{A}, v \models \varphi.$$

Jeśli $FV(\varphi) \subseteq \{x_1, \dots, x_k\}$, oraz $v(x_1) = a_1, \dots, v(x_k) = a_k$ i φ jest spełniona w $\mathcal{A}$ przez v (czyli $\varphi^{\mathcal{A}}(a_1, \dots, a_n) = 1$), to zamiast $\mathcal{A}, v \models \varphi$ piszemy też

$$\mathcal{A}, \{a_1/x_1, \ldots, a_n/x_n\} \models \varphi, \text{ lub po prostu } \mathcal{A}, a_1, \ldots, a_n \models \varphi,$$

gdy kolejność argumentów jest oczywista. Podobnie, zamiast $[\![\varphi]\!]_v$ można wtedy napisać $[\![\varphi]\!]\{a_1/x_1, \ldots, a_n/x_n\}$. W szczególności gdy formuła jest zamknięta, zamiast $[\![\varphi]\!]_v$ piszemy po prostu $[\![\varphi]\!]$.

Uwaga: Zamiast $\mathcal{A}, \{a_1/x_1, \ldots, a_n/x_n\} \models \varphi$ czasem *nieformalnie* piszemy

$$\mathcal{A} \models \varphi(a_1, \ldots, a_n),$$

ale ten zapis w zasadzie nie jest poprawny. Pamiętajmy, że $a_1, \ldots, a_n$ są elementami modelu (a nie częścią składni), więc „$\varphi(a_1, \ldots, a_n)$" nie oznacza żadnej formuły.

Definicja 8.5 Formuła jest *spełnialna (spełnialna w $\mathcal{A}$)* jeśli jest spełniona w pewnym modelu (w modelu $\mathcal{A}$) przez pewne wartościowanie. Zbiór formuł jest *spełnialny* (w $\mathcal{A}$) jeśli wszystkie formuły z tego zbioru są spełnione przez to samo wartościowanie w pewnym modelu (w modelu $\mathcal{A}$).

Formuła φ jest *prawdziwa w $\mathcal{A}$* (piszemy $\mathcal{A} \models \varphi$), jeżeli jest spełniona w $\mathcal{A}$ przez wszystkie wartościowania. Formuła φ jest *prawdziwa* (jest *tautologią*) jeżeli jest prawdziwa w każdym modelu $\mathcal{A}$. Wtedy piszemy po prostu $\models \varphi$.

Przykład 8.6

- Formuła $\forall x(y \leq x)$ jest spełniona w $\langle \mathbb{N}, \leq \rangle$ dla $v(y) = 0$, ale nie jest spełnialna w modelu $\langle \mathbb{R}, \leq \rangle$.
- Formuła $\exists x(\neg r(x) \wedge r(x))$ nie jest spełnialna.
- Formuła $\exists x(y \leq x)$ jest prawdziwa w $\langle \mathbb{R}, \leq \rangle$.
- Formuła $p(x) \wedge \exists y \neg p(y)$ jest spełnialna, ale nie jest w żadnej strukturze prawdziwa.

Definicja 8.7 Niech Γ będzie zbiorem formuł. Wówczas:

Piszemy	$\mathcal{A}, v \models \Gamma$,	jeżeli	$\mathcal{A}, v \models \varphi$, dla dowolnego $\varphi \in \Gamma$;
"	$\mathcal{A} \models \Gamma$,	"	$\mathcal{A}, v \models \Gamma$, dla dowolnego v;
"	$\models \Gamma$,	"	$\mathcal{A} \models \Gamma$, dla dowolnego $\mathcal{A}$;
"	$\Gamma \models \psi$,	"	$\mathcal{A}, v \models \psi$, dla dowolnych $\mathcal{A}, v$, takich że $\mathcal{A}, v \models \Gamma$;
"	$\Gamma \models_{\forall} \psi$,	"	$\mathcal{A} \models \psi$, dla dowolnego $\mathcal{A}$, takiego że $\mathcal{A} \models \Gamma$.

Uwaga: Niech $\Gamma^{\forall} = \{\forall\gamma \mid \gamma \in \Gamma\}$, gdzie $\forall\gamma$ oznacza formułę otrzymaną z γ przez dopisanie z przodu tylu kwantyfikatorów ogólnych, aby wszystkie zmienne w γ zostały związane. Wtedy $\Gamma \models_{\forall} \psi$ wtedy i tylko wtedy, gdy $\Gamma^{\forall} \models \psi$. W szczególności stwierdzenia $\Gamma \models_{\forall} \psi$ i $\Gamma \models \psi$ są równoważne, gdy Γ jest zbiorem zdań.

Operacja podstawienia

Przez $t[x := s]$ oznaczaliśmy term powstały z termu t przez podstawienie termu s w miejsce wszystkich wystąpień zmiennej x.

W podobny sposób można łatwo zdefiniować podstawienie termu w miejsce zmiennej do formuły otwartej. Podstawianie termu w miejsce zmiennej w formule z kwantyfikatorami wymaga jednak pewnej ostrożności. Po pierwsze, podstawienie powinno dotyczyć tylko wolnych wystąpień zmiennych. Po drugie, „naiwne” podstawienie może prowadzić do użycia tej samej zmiennej w dwóch znaczeniach. Na przykład formuły „$\exists y(y < x)$” oraz „$\exists z(z < x)$” są równoważne. Tymczasem „naiwne” podstawienie y w miejsce x w obu tych formułach daje w wyniku odpowiednio „$\exists y(y < y)$” i „$\exists z(z < y)$”, a te dwie formuły znaczą całkiem co innego. Przyczyną jest to, że w pierwszym przypadku zmienną y wstawiono w zasięg kwantyfikatora $\exists y$. Jedno z możliwych rozwiązań tego problemu jest takie: Przed dokonaniem podstawienia należy wymienić zmienne związane, które taki konflikt mogłyby spowodować, na „nowe zmienne”. My przyjmiemy tym razem inne rozwiązanie: Podstawienie jest określone tylko wtedy, gdy nie powoduje konfliktu nazw.

Definicja 8.8 Poniższe warunki należy czytać tak: Jeśli prawa strona jest określona, to lewa strona też jest określona.

- $\bot[x := s] = \bot$, gdy $x \notin FV(\varphi)$;
- $r(t_1, \ldots, t_n)[x := s] = r(t_1[x := s], \ldots, t_n[x := s])$;
- $(t_1 = t_2)[x := s] =$ „$t_1[x := s] = t_2[x := s]$”;
- $(\varphi \to \psi)[x := s] = \varphi[x := s] \to \psi[x := s]$;
- $(\forall x\, \varphi)[x := s] = \forall x\, \varphi$;
- $(\forall y\, \varphi)[x := s] = \forall y\, \varphi[x := s]$, gdy $y \neq x$, oraz $y \notin FV(s)$;
- W pozostałych przypadkach podstawienie jest nieokreślone.

Oczywiście istnieje związek między podstawieniem i transpozycją.

Lemat 8.9 *Jeśli $y \notin FV(\varphi)$ i $\varphi[x := y]$ jest okreslone, to $\varphi[x := y] =_\alpha \varphi(x \wr y)$.*

Można teraz sprawdzić (żmudny dowód opuszczamy), że podstawienie zachowuje alfa-równoważność.

Lemat 8.10 *Jeśli $\varphi_1 =_\alpha \varphi_2$ to $\varphi_1[x := s] =_\alpha \varphi_2[x := s]$, o ile obie strony są określone.*

Następny lemat jest uogólnieniem Lematu 2.11 i wyraża semantyczny sens podstawienia: wartość formuły powstającej przez podstawienie równa jest wartości formuły początkowej przy wartościowaniu uwzględniającym wartość podstawianego termu.

Lemat 8.11 *Jeśli podstawienie $\varphi[x := s]$ jest określone, to dla dowolnego wartościowania v zachodzi równość $v(\varphi[x := s]) = v_x^{v(s)}(\varphi)$.*

Dowód: Dla dowolnej formuły atomowej $r(t_1, \ldots, t_n)$, wartość $v(r(t_1, \ldots, t_n)[x := s]) = v(r(t_1[x := s], \ldots, t_n[x := s])$ jest równa 1 wtedy i tylko wtedy, gdy wektor wartości $\langle v(t_1[x := s]), \ldots, v(t_n[x := s])\rangle$ należy do $r^{\mathcal{A}}$. Ale na mocy Lematu 2.11, dla dowolnego termu t zachodzi równość $v(t[x := s]) = v_x^{v(s)}(t)$. Stąd $v(r(t_1, \ldots, t_n)[x := s]) = 1$ wtedy i tylko wtedy, gdy $\langle v_x^{v(s)}(t_1), \ldots, v_x^{v(s)}(t_n)\rangle \in r^{\mathcal{A}}$, czyli wtedy i tylko wtedy, gdy $v_x^{v(s)}(r(t_1, \ldots, t_n)) = 1$. Przypadek równości jest podobny.

Dla $\varphi = \psi \to \vartheta$ mamy takie równości: $v(\varphi[x := s]) = v(\psi[x := s] \to \vartheta[x := s]) = \max\{1 - v(\psi[x := s]), v(\vartheta[x := s])\} = \max\{1 - v_x^{v(s)}(\psi), v_x^{v(s)}(\vartheta)\} = v_x^{v(s)}(\varphi)$.

Rozpatrzmy na koniec przypadek gdy φ rozpoczyna się od kwantyfikatora uniwersalnego. Jeśli $\varphi = \forall x\psi$, to $v(\varphi[x := s]) = v(\varphi) = v_x^{v(s)}(\varphi)$ z Lematu 8.2, bo $v|_{FV(\varphi)} = v_x^{v(s)}|_{FV(\varphi)}$.

Jeśli $\varphi = \forall y\psi$, gdzie $x \neq y \notin FV(s)$ to korzystamy z założenia indukcyjnego dla ψ i mamy: $v(\varphi[x{:=}s]) = v(\forall y\psi[x{:=}s]) = \min\{v_y^a(\psi[x{:=}s]) \mid a \in |\mathcal{A}|\} = \min\{(v_y^a)_x^{v_y^a(s)}(\psi) \mid a \in |\mathcal{A}|\} = \min\{(v_y^a)_x^{v(s)}(\psi) \mid a \in |\mathcal{A}|\} = \min\{(v_x^{v(s)})_y^a(\psi) \mid a \in |\mathcal{A}|\} = v_x^{v(s)}(\forall y\psi) = v_x^{v(s)}(\varphi)$. (Korzystaliśmy z tego, że $v_y^a(s) = v(s)$, bo $y \notin FV(s)$.) ∎

Jak dotąd, stwierdzenia dotyczące podstawień musieliśmy opatrywać zastrzeżeniami „o ile podstawienie jest określone". Ale skoro utożsamiamy alfa-równoważne formuły, to możemy zapomnieć o zastrzeżeniach. Mamy bowiem taki fakt:

Lemat 8.12 *Niech φ będzie dowolną formułą i niech s będzie dowolnym termem. Istnieje formuła φ', dla której podstawienie $\varphi'[x := s]$ jest określone i taka, że $\varphi =_\alpha \varphi'$.*

Dowód: Dowód jest przez indukcję ze względu na długość formuły. Jedyny nieoczywisty przypadek ma miejsce dla $\varphi = \forall y\psi$, gdzie $x \neq y \in FV(s)$. Dobierzmy zmienną z tak, aby $z \notin FV(\psi)$ oraz $z \notin FV(s)$. Z założenia indukcyjnego istnieje taka formuła ψ_1, że $\psi_1 =_\alpha \psi(y \wr z)$ oraz podstawienie $\psi_1[x := s]$ jest określone. A więc wystarczy przyjąć $\varphi' = \forall z\psi_1$. ∎

Odtąd zakładamy, że podstawienie jest zawsze określone, bo w razie potrzeby możemy wymienić każdą zmienną związaną na „nową". A więc, na przykład:

- $(\exists z(z < x) \to \exists x(x < y))[x := f(y)] =$ „$\exists z(z < f(y)) \to \exists x(x < y)$";

- $(\exists y(y < x))[x := y] =$ „$\exists z(z < y)$”.

Uwaga: Warto zauważyć analogię pomiędzy zmiennymi wolnymi i związanymi w formułach oraz identyfikatorami lokalnymi i nielokalnymi w blokach i procedurach. Dokonanie operacji podstawienia w formule odpowiada przy tej analogii wywołaniu procedury z parametrem przekazanym *przez nazwę*.

9 Logika formalna i jezyk polski

Systemy logiki formalnej są, jak już mówiliśmy, tylko pewnymi modelami, czy też przybliżeniami rzeczywistych sposobów wyrażania różnych stwierdzeń i wnioskowania o ich poprawności. Poziom dokładności takich przybliżeń może być większy lub mniejszy. Większy tam, gdzie mamy do czynienia z dobrze określoną teorią matematyczną, lub językiem programowania. Mniejszy wtedy, gdy używamy logiki do weryfikacji poprawności stwierdzeń języka potocznego, chocby takiego jak podręcznikowy przykład: *„Janek idzie do szkoły.”* Oczywiście przypisanie temu stwierdzeniu wartości logicznej jest zgoła niemożliwe, nie wiemy bowiem, który Janek do jakiej idzie szkoły i czy może już doszedł? Więcej sensu ma zastosowanie logiki predykatów do analizy np. takiego rozumowania:

Każdy cyrulik sewilski goli tych wszystkich mężczyzn w Sewilli, którzy się sami nie golą.
Ale nie goli żadnego z tych, którzy golą się sami.
A zatem w Sewilli nie ma ani jednego cyrulika.

W tym przypadku aparat logiki formalnej może być pomocny. Łatwiej zrozumieć o co chodzi, tłumacząc nasz problem na język logiki predykatów, i przedstawiając go jako pytanie o poprawność pewnego stwierdzenia postaci $\Gamma \models \varphi$. Można więc zapytać, czy

$$\forall x(C(x) \wedge S(x) \to \forall y(G(x,y) \leftrightarrow S(y) \wedge \neg G(y,y))) \models \neg\exists x(C(x) \wedge S(x))?$$

Stwierdziwszy poprawność powyższego stwierdzenia, wywnioskujemy, że w Sewilli cyrulika nie ma. I będzie to wniosek… błędny, bo cyrulik być może jest kobietą.

W powyższym przykładzie po prostu źle ustalono logiczną interpretację naszego zadania, zapominając o jednym z jego istotnych elementów. Błąd ten można łatwo naprawić, co jest zalecane jako ćwiczenie. Ale nie zawsze język logiki formalnej wyraża ściśle to samo, co potoczny język polski, a nawet język w którym pisane są prace matematyczne. Zarówno składnia jak i semantyka tych języków rządzi się zupełnie innymi prawami, i o tym należy pamiętać tłumacząc jeden na drugi.

Implikacja materialna i związek przyczynowo-skutkowy

Implikacja, o której mówimy w logice klasycznej to tzw. *implikacja materialna.* Wartość logiczna, którą przypisujemy wyrażeniu „$\alpha \to \beta$" zależy wyłącznie od wartości logicznych przypisanych jego częściom składowym α i β. Nie zależy natomiast zupełnie od treści tych wyrażeń, czy też jakichkolwiek innych związków pomiędzy α i β. W szczególności, wypowiedzi α i β mogą mówić o zajściu jakichś zdarzeń i wtedy wartość logiczna implikacji „$\alpha \to \beta$" nie ma nic wspólnego z ich ewentualnym następstwem w czasie, lub też z tym, że jedno z tych zdarzeń spowodowało drugie. W języku polskim stwierdzenie *„jeśli α to β"* oczywiście sugeruje taki związek, np. w zdaniu:

Jeśli zasilanie jest włączone, to terminal działa.

Tymczasem implikacja materialna nie zachodzi, o czym dobrze wiedzą użytkownicy terminali. Co więcej, w istocie materialną prawdą jest stwierdzenie odwrotne:

Jeśli terminal działa to zasilanie jest włączone.

Natomiast zdanie

Terminal działa, ponieważ zasilanie jest włączone,

stwierdza nie tylko związek przyczynowo-skutkowy, ale też faktyczne zajście wymienionych zdarzeń i w ogóle nie ma odpowiednika w logice klasycznej.

Inne spójniki w mniejszym stopniu grożą podobnymi nieporozumieniami. Ale na przykład spójnik „i" w języku polskim może też sugerować następstwo czasowe[6] zdarzeń: *„Poszedł i zrobił."* A wyrażenie *„A, chyba że B"* ma inny odcień znaczeniowy niż proste *„A lub B"*. Przy tej okazji zwróćmy uwagę na to, że słowo *„albo"* bywa czasem rozumiane w znaczeniu alternatywy wykluczającej. My jednak umawiamy się, że rozumiemy je tak samo jak *„lub"*.

Konfuzje składniowe

Przy tłumaczeniu z języka polskiego na język logiki formalnej i na odwrót mozna łatwo popełnić błąd nawet wtedy gdy nie powstają problemy semantyczne. Składnia tych języków jest oparta na innych zasadach. Na przykład te dwa zdania mają bardzo podobną budowę:

Każdy kot ma wąsy.
Pewien kot ma wąsy.

Ale ich tłumaczenia na język rachunku predykatów nie są już takie podobne:

$$\forall x(Kot(x) \to MaWąsy(x));$$
$$\exists x(Kot(x) \wedge MaWąsy(x)).$$

[6] W językach programowania jest podobnie, ale to już inna historia.

Zadziwiająco częstym błędem jest właśnie mylenie koniunkcji z implikacją w zasięgu działania kwantyfikatora. A oto inny przykład: Zdania

„Liczba n jest parzysta";

„Liczba n jest dwukrotnością pewnej liczby"

oznaczają to samo. Zaprzeczeniem pierwszego z nich jest oczywiście zdanie

„Liczba n nie jest parzysta",

ale zaprzeczeniem drugiego nie jest zdanie

„Liczba n nie jest dwukrotnością pewnej liczby",

otrzymane przecież przez analogiczną operację „podstawienia". To zdanie rozumiemy bowiem jako $\exists x(\neg n = 2x)$, a nie jako $\neg\exists x(n = 2x)$, mimo że negacja *„nie"* poprzedza w nim słowo *„pewnej"*.

Innym popularnym błędem jest mylenie koniunkcji z alternatywą w przesłance implikacji, zwłaszcza gdy występuje tam negacja. Mamy bowiem skłonność do powtarzania słowa "nie" w obu członach założenia i nie razi nas zdanie

„Kto nie ma biletu lub nie jest pracownikiem teatru, ten nie wejdzie na przedstawienie".

Ale od tekstu matematycznego oczekujemy więcej ścisłości i w takim tekście zdanie:

„Jeśli x nie jest równe 2 lub nie jest równe 3, to $x^2 - 5x + 6$ nie jest zerem."

powinno być uważane za błędne. Wielu takich błędów unikniemy, gdy przypomnimy sobie, że w języku polskim istnieją takie słowa jak *„ani"* i *„żaden"*.

10 Logika zdaniowa

Czasami do opisu pewnych zjawisk matematycznych nie jest potrzebny cały aparat logiki predykatów. Wystarczy często logika *zdaniowa* (nazywana też *rachunkiem zdań*). Jest to takie uproszczenie logiki predykatów, w którym ignoruje się „przedmioty", a rozważa jedynie wyrażenia orzekające (zdania). Oczywiście, w tej sytuacji kwantyfikatory indywiduowe stają się niepotrzebne, a rolą logiki staje się badanie w jaki sposób znaczenie wyrażeń prostych wpływa na znaczenie wyrażeń złożonych.

Logikę zdaniową można uważać za fragment logiki pierwszego rzędu.

Definicja 10.1 Zdania otwarte dowolnej sygnatury Σ, składającej się wyłącznie z symboli zdaniowych, nazywamy *formułami zdaniowymi*. Pojęcie formuły zdaniowej można więc tak zdefiniować przez indukcję:

- Symbole zdaniowe i $\bot$ są formułami zdaniowymi;
- Jeśli α i β są formułami zdaniowymi to $(\alpha \to \beta)$ jest formułą zdaniową.

W tej sytuacji pojęcie spełnialnej i prawdziwej formuły zdaniowej jest oczywiście określone zgodnie z Definicją 8.5, odwołującą się do struktur algebraicznych. Ale skoro sygnatura składa się tylko z zeroargumentowych symboli relacyjnych, a my nie używamy zmiennych, to w strukturze $\mathcal{A} = \langle A, p_1^{\mathcal{A}}, \ldots, p_n^{\mathcal{A}} \rangle$ interesuje nas tylko to, która z zeroargumentowych jest pusta (równa 0) a która pełna (równa 1). Natomiast sam zbiór A jest całkiem nieważny.

Definicja 10.2 Przez *wartościowanie zdaniowe* rozumiemy dowolną funkcję ρ, która symbolom zdaniowym przypisuje wartości logiczne 0 lub 1. *Wartość formuły* zdaniowej przy wartościowaniu ρ określamy przez indukcję:

- $[\![\bot]\!]_\rho = 0$;
- $[\![p]\!]_\rho = \rho(p)$, gdy p jest symbolem zdaniowym;
- $[\![\alpha \to \beta]\!]_\rho = \max\{[\![\beta]\!]_\rho, 1 - [\![\alpha]\!]_\rho\}$.

Zamiast $[\![\alpha]\!]_\rho$ zwykle piszemy po prostu $\rho(\alpha)$.

Jeśli $\rho(\alpha) = 1$ to piszemy też $\rho \models \alpha$. Jeśli Γ jest zbiorem formuł zdaniowych, oraz $\rho \models \gamma$ dla wszystkich $\gamma \in \Gamma$, to piszemy $\rho \models \Gamma$.

Symbole zdaniowe są często nazywane *zmiennymi zdaniowymi*. Jest to jednak zupełnie inny rodzaj zmiennych, niż zmienne indwiduowe, występujące w formułach rachunku predykatów. Również wartościowanie zdaniowe to trochę co innego niż wartościowanie termów.[7]

Fakt 10.3

- *Jeśli Γ jest zbiorem formuł zdaniowych, a α jest formułą zdaniową, to $\Gamma \models \alpha$ zachodzi wtedy i tylko wtedy, gdy $\rho \models \alpha$, dla dowolnego wartościowania zdaniowego ρ, takiego że $\rho \models \Gamma$. W szczególności:*
- *Formuła zdaniowa jest spełnialna wtedy i tylko wtedy, gdy $\rho(\alpha) = 1$ dla pewnego wartościowania zdaniowego ρ.*
- *Formuła zdaniowa jest prawdziwa wtedy i tylko wtedy, gdy $\rho(\alpha) = 1$ dla każdego wartościowania zdaniowego ρ.*

Widzimy więc, że nasza interpretacja formuł zdaniowych jest zgodna z tą, którą znamy ze szkoły.

[7] Ale może nie całkiem? Ćwiczenie: czy formuły zdaniowe można uważać za termy jakiejś sygnatury? Czy wartościowanie zdaniowe staje sie wtedy wartościowaniem termów w pewnej algebrze?

Tautologie rachunku zdań

Niech S będzie funkcją przypisujacą symbolom zdaniowym pewne formuły. Jeśli α jest formułą zdaniową, to przez $S(\alpha)$ oznaczymy formułę otrzymaną z α przez zamianę każdego wystąpienia zmiennej zdaniowej p na formułę $S(p)$. Mówimy, że formuła $S(\alpha)$ jest *instancją* schematu zdaniowego α. Używamy oznaczenia $S(\Gamma) = \{S(\beta) \mid \beta \in \Gamma\}$.

Fakt 10.4 *Jeżeli Γ jest zbiorem formuł rachunku zdań i $\Gamma \models \alpha$, to także $S(\Gamma) \models S(\alpha)$. W szczególności, jeśli α jest tautologią to $S(\alpha)$ jest też tautologią.*

Dowód: Ćwiczenie. ∎

Fakt 10.5 *Następujące formuły (i wszystkie ich instancje) są tautologiami rachunku zdań:*

1. $\bot \to p$;
2. $p \to (q \to p)$;
3. $(p \to (q \to r)) \to ((p \to q) \to (p \to r))$;
4. $((p \to q) \to p) \to p$;
5. $p \vee \neg p$;
6. $p \to \neg\neg p$ *i* $\neg\neg p \to p$;
7. $(p \to q) \to (\neg q \to \neg p)$ *i* $(\neg q \to \neg p) \to (p \to q)$;
8. $p \to (p \vee q)$, $q \to (p \vee q)$ *oraz* $(p \to r) \to ((q \to r) \to (p \vee q \to r))$;
9. $(p \wedge q) \to p$, $(p \wedge q) \to q$ *oraz* $(r \to p) \to ((r \to q) \to (r \to (p \wedge q)))$;
10. $((p \wedge q) \to r) \leftrightarrow (p \to (q \to r))$;
11. $\neg(p \vee q) \leftrightarrow (\neg p \wedge \neg q)$;
12. $\neg(p \wedge q) \leftrightarrow (\neg p \vee \neg q)$;
13. $(p \to q) \leftrightarrow (\neg p \vee q)$;
14. $((p \leftrightarrow q) \leftrightarrow r) \leftrightarrow (p \leftrightarrow (q \leftrightarrow r))$.

Dowód: Łatwy. ∎

Niektóre z powyższych formuł wskazują na analogię pomiędzy implikacją i uporządkowaniem (np. zawieraniem zbiorów). Implikację „$p \to q$" można odczytać tak: „warunek p jest silniejszy (mniejszy lub równy) od q". Formułę (1) czytamy wtedy: „fałsz jest najsilniejszym warunkiem (najmniejszym elementem)". Formuły (8) stwierdzają, że alternatywa $p \vee q$ jest najsilniejszym warunkiem, który wynika zarówno z p jak i z q (czyli jest kresem górnym pary $\{p, q\}$, jak suma zbiorów). Formuły (9) wyrażają dualną własność koniunkcji: to jest kres dolny, czyli najsłabszy warunek implikujący oba argumenty. Prawa de Morgana (11,12) wskazują też na analogie koniunkcja–iloczyn, alternatywa–suma, negacja–dopełnienie. Ta ostatnia widoczna jest też w prawach wyłączonego środka (5), podwójnej negacji (6) i kontrapozycji (7).

O ile (9) wskazuje na analogię pomiędzy koniunkcją i iloczynem mnogościowym, o tyle warto zauważyć, że koniunkcja ma też własności podobne do iloczynu kartezjańskiego. Jeśli zbiór funkcji z A do B oznaczymy przez $[A \to B]$, to mamy (bardzo naturalną) równoliczność $[A \times B \to C] \sim [A \to [B \to C]]$. Podobieństwo tego związku do formuły (10) nie jest wcale przypadkowe.

Formuła (13) wyraża implikację z pomocą negacji i alternatywy i jest często bardzo przydatna, gdy np. chcemy przekształcić jakąś formułę do prostszej postaci.

Formuła (2) mówi, że dodatkowe założenie można zawsze zignorować. Formuła (3) (prawo Frege) wyraża dystrybutywność implikacji względem siebie samej i może być odczytywana tak: jeśli r wynika z q w kontekście p, to ten kontekst może być włączony do założenia i konkluzji. Formuła (4) (prawo Peirce'a) wyraża przy pomocy samej implikacji zasadniczą własność logiki klasycznej: możliwość rozumowania przez zaprzeczenie. Sens prawa Peirce'a widać najlepiej gdy q jest fałszem, mamy wtedy prawo Claviusa: $(\neg p \to p) \to p$.

Warto zauważyć, że formuły w parach (6,7) nie są wcale tak symetryczne jak się wydaje na pierwszy rzut oka. Na przykład, pierwsza z formuł (6) to w istocie $p \to ((p \to \bot) \to \bot)$. Wiedząc, że p i $p \to \bot$, natychmiast zgadzamy się na $\bot$. Intuicyjne uzasadnienie drugiej formuły jest zaś w istocie związane z prawem (5).

Własnością, która często uchodzi naszej uwagi, jest łączność równoważności (14). W zwiazku z tym, wyrażenie $\alpha \leftrightarrow \beta \leftrightarrow \gamma$ można z czystym sumieniem pisać bez nawiasów. Zwróćmy jednak uwagę na to, że oznacza ono zupełnie co innego niż stwierdzenie że α, β i γ są sobie nawzajem równoważne!

Powyżej pominięto bardziej oczywiste prawa: łączność i przemienność koniunkcji i alternatywy, ich wzajemną dystrybutywność, przechodniość i zwrotność implikacji itp.

Postać normalna formuł

Definicja 10.6 Każdy symbol zdaniowy i negację symbolu zdaniowego nazywamy *literałem*. Mówimy, że formuła zdaniowa α jest w *koniunkcyjnej postaci normalnej*, gdy α jest koniunkcją alternatyw literałów, tj.

$$\alpha = (p_1^1 \vee \cdots \vee p_1^{k_1}) \wedge \cdots \wedge (p_r^1 \vee \cdots \vee p_r^{k_r}), \qquad (*)$$

gdzie $r \geq 1$, $k_i \geq 1$, dla $i = 1, \ldots r$, a wszystkie p_j^i są literałami.

Formułami *równoważnymi* nazywamy takie formuły α i β, że równoważność $\alpha \leftrightarrow \beta$ jest tautologią.

Fakt 10.7 *Dla każdej formuły zdaniowej istnieje równoważna jej formuła w koniunkcyjnej postaci normalnej.*

Dowód: Dowód jest przez indukcję ze względu na budowę formuły. Symbole zdaniowe są oczywiscie w postaci normalnej. Postacią normalną dla formuły $\perp$ jest $p \wedge \neg p$. Jeśli α jest w postaci (*), to $\neg\alpha$ można przekształcić w koniunkcyjną postać normalną stosując prawa de Morgana i prawa dystrybutywności:

$$\beta \vee (\gamma \wedge \delta) \leftrightarrow (\beta \vee \gamma) \wedge (\beta \vee \delta) \qquad\qquad \beta \vee (\gamma \vee \delta) \leftrightarrow (\beta \vee \gamma) \vee (\beta \vee \delta).$$

Podobnie postępujemy z implikacją[8], przekształcając ją z pomocą prawa 10.5(13). ∎

11 Tautologie rachunku predykatów

Wracamy teraz do logiki predykatów. Nietrudno zauważyć, że Fakt 10.4 jest prawdziwy także wtedy gdy formuły $S(p)$ niekoniecznie są zdaniowe. Każda instancja tautologii zdaniowej jest więc tautologią rachunku predykatów. Oczywiście jest wiele interesujących tautologii, w których istotną rolę odgrywają kwantyfikatory.

Fakt 11.1 *Dla dowolnych φ i ψ, następujące formuły są tautologiami:*

1. $\forall x(\varphi \to \psi) \to (\forall x\varphi \to \forall x\psi)$;
2. $\varphi \to \forall y\varphi$, *gdzie* $y \notin FV(\varphi)$;
3. $\forall x\varphi \to \varphi[x := s]$.

[8] Ta procedura jest dosyć pracochłonna. Można ją nieco uprościć, ale najgorszy przypadek i tak pozostanie wykładniczy.

Dowód: (1) Niech $\mathcal{A}$ będzie modelem i v niech będzie wartościowaniem w $\mathcal{A}$. Jeśli $v(\forall x\varphi) = 0$ lub $v(\forall x(\varphi \to \psi)) = 0$ to nasza formuła jest oczywiście spełniona przez v. Załóżmy więc, że $v(\forall x(\varphi \to \psi)) = v(\forall x\varphi) = 1$. Należy pokazać, że $v(\forall x\psi) = 1$. Weźmy dowolne $a \in |\mathcal{A}|$. Skoro $v(\forall x\varphi) = 1$, to musi być $v_x^a(\varphi) = 1$. Podobnie $v_x^a(\varphi \to \psi) = 1$. Ale wtedy także $v_x^a(\psi) = 1$. Zatem $v(\forall x\psi) = \min\{v_x^a(\psi) \mid a \in |\mathcal{A}|\} = 1$, i dobrze.

(2) Można założyć, że $v(\varphi) = 1$. Dla dowolnego $a \in |\mathcal{A}|$, na mocy Lematu 8.2, mamy równość $v_y^a(\varphi) = v(\varphi)$, bo $FV(\varphi) = FV(\forall y\varphi)$. Zatem $v(\forall y\varphi) = \min\{1\} = 1$.

(3) Na mocy Lematu 8.11, dla dowolnej struktury $\mathcal{A}$ i dowolnego wartościowania v zachodzi: $v(\varphi[x := s]) = v_x^{v(s)}(\varphi) \geq \min\{v_x^a(\varphi) \mid a \in |\mathcal{A}|\} = v(\forall x\varphi)$. ∎

Jak się niebawem okaże, trzy tautologie, o których mowa powyżej, wyrażają podstawowe własności kwantyfikatora ogólnego. Pierwsza z nich to prawo rozkładu kwantyfikatora ogólnego (rozdzielność $\forall$ względem implikacji). Druga pozwala na dopisanie kwantyfikatora, który nie wiąże żadnej zmiennej, a trzecia, zwana *dictum de omni*, mówi że uniwersalna zasada obowiązuje każdego.

Kolejne dwa ważne przykłady tautologii dotyczą równości.

Fakt 11.2 *Następujące formuły (gdzie φ jest dowolną formułą) są tautologiami:*

1. $\forall x(x = x)$;
2. $\forall x\forall y(x = y \to (\varphi[z := y] \to \varphi[z := x]))$.

Dowód: Część pierwsza jest oczywista, a część druga wynika z Lematu 8.11, bo dla dowolnego wartościowania v, jeśli $v(x) = v(y)$ to $v(\varphi[z := y]) = v_z^{v(y)}(\varphi) = v_z^{v(x)}(\varphi) = v(\varphi[z := x])$. ∎

Uwaga: Jeśli w drugiej części Faktu 11.2 jako φ wybierzemy formułę „$z = u$” to otrzymamy

$$\forall x\forall y(x = y \to (y = u \to x = u)).$$

Natomiast dla $\varphi =$ „$y = z$” dostaniemy

$$\forall x\forall y(x = y \to (y = y \to y = x)).$$

Pierwsza z tych formuł wyraża przechodniość równości, a druga jej symetrię.

Oczywiście dowód, że dana formuła jest tautologią polega na analizie jej spełniania w dowolnych modelach. Natomiast wykazanie, że tak nie jest polega na podaniu odpowiedniego kontrprzykładu. Takiego jak ten:

Przykład 11.3 Formuła $(\forall xp(x) \to \forall xq(x)) \to \forall x(p(x) \to q(x))$ nie jest tautologią. Rozpatrzmy bowiem model $\mathcal{A} = \langle \mathbb{N}, p^{\mathcal{A}}, q^{\mathcal{A}} \rangle$, w którym:

- $n \in p^{\mathcal{A}}$, wtedy i tylko wtedy, gdy n jest parzyste;
- $n \in q^{\mathcal{A}}$, wtedy i tylko wtedy, gdy n jest nieparzyste;

Wtedy $\mathcal{A} \not\models \forall x p(x)$ więc $\mathcal{A} \models \forall x p(x) \to \forall x q(x)$. Ale $\mathcal{A} \not\models \forall x(p(x) \to q(x))$, bo $\mathcal{A}, 2 \not\models p(x) \to q(x)$. Rzeczywiście, $2 \in p^{\mathcal{A}} - q^{\mathcal{A}}$.

Przyjrzyjmy się jeszcze kilku ważnym tautologiom.

Fakt 11.4 *Następujące formuły są tautologiami (dla dowolnych φ i ψ).*

1. $\forall x(\varphi \to \psi) \to (\exists x\varphi \to \exists x\psi)$;
2. $\exists x\varphi \to \varphi$, *o ile* $x \notin FV(\varphi)$;
3. $\varphi[x := s] \to \exists x\varphi$;
4. $\neg\forall x\varphi \leftrightarrow \exists x\neg\varphi$;
5. $\neg\exists x\varphi \leftrightarrow \forall x\neg\varphi$;
6. $\forall x(\varphi \wedge \psi) \leftrightarrow \forall x\varphi \wedge \forall x\psi$;
7. $\exists x(\varphi \vee \psi) \leftrightarrow \exists x\varphi \vee \exists x\psi$;
8. $\forall x(\varphi \vee \psi) \leftrightarrow \varphi \vee \forall x\psi$, *o ile* $x \notin FV(\varphi)$;
9. $\exists x(\varphi \wedge \psi) \leftrightarrow \varphi \wedge \exists x\psi$, *o ile* $x \notin FV(\varphi)$;
10. $\forall x\varphi \to \exists x\varphi$;
11. $\forall x\forall y\varphi \leftrightarrow \forall y\forall x\varphi$;
12. $\exists x\exists y\varphi \leftrightarrow \exists y\exists x\varphi$;
13. $\exists x\forall y\varphi \to \forall y\exists x\varphi$;

Dowód: Ćwiczenie. ∎

Formuły (1)–(3) powyżej wyrażają własności kwantyfikatora szczegółowego i są odpowiednikami formuł z Faktu 11.1. Zauważmy, że zamiast rozdzielności kwantyfikatora szczegółowego, mamy tu jeszcze jedno prawo rozkładu kwantyfikatora ogólnego. Symetria pomiędzy $\forall$ i $\exists$ nie jest wcale całkowita! Niemniej istnieje, co wyrażają prawa de Morgana (4) i (5).

Kolejne dwie tautologie przypominają o bliskim związku kwantyfikatora ogólnego z koniunkcją i kwantyfikatora szczegółowego z alternatywą. (Uwaga: zmienna x może być wolna w φ i ψ.) Analogiczna rozdzielność kwantyfikatora ogólnego względem alternatywy (8) i kwantyfikatora szczegółowego względem koniunkcji (9) nie zawsze jest prawdą, ale zachodzi pod warunkiem, że zmienna wiązana kwantyfikatorem nie występuje w jednym z członów formuły. (Prawo (8) nazywane bywa prawem Grzegorczyka lub aksjomatem Gabbaya.)

Formuła (10) jest odbiciem naszego założenia o niepustości świata. Jest to tautologia, ponieważ umówiliśmy się, że rozważamy tylko niepuste struktury.

Prawa (11)–(13) charakteryzują możliwości permutowania kwantyfikatorów. Implikacja odwrotna do (13) zazwyczaj nie jest tautologią.

Stosując równoważności (4–9) możemy każdą formułę sprowadzić do postaci, w której wszystkie kwantyfikatory znajdują się na początku. Mówimy, że formuła φ jest w *preneksowej postaci normalnej*, gdy

$$\varphi = Q_1 y_1 Q_2 y_2 \ldots Q_n y_n \psi,$$

gdzie każde z Q_i to $\forall$ lub $\exists$, a ψ jest formułą otwartą. (Oczywiście n może być zerem.)

Fakt 11.5 *Dla każdej formuły pierwszego rzędu istnieje równoważna jej formuła w preneksowej postaci normalnej.*

Dowód: Indukcja (ćwiczenie). ∎

12 Dowodzenie twierdzeń

Sprawdzenie, czy dana formuła rachunku zdań jest tautologią, wymaga obliczenia jej wartości dla 2^n różnych wartościowań, gdzie n jest liczbą zmiennych zdaniowych tej formuły. Dla rachunku predykatów nie istnieje w ogóle żaden algorytm sprawdzania czy dana formuła jest tautologią. W obu przypadkach są jednak metody *dowodzenia* pozwalające na wyprowadzenie każdej prawdziwej formuły z pomocą pewnego ustalonego systemu reguł wnioskowania i aksjomatów. Jeden z takich systemów opisany jest poniżej. (Zakładamy, że ustalona jest sygnatura Σ, a formuły różniące się tylko zmiennymi związanymi uważamy za identyczne.) *Aksjomatami* naszego systemu są wszystkie formuły postaci:

A1) $\varphi \to (\psi \to \varphi)$;

A2) $(\varphi \to (\psi \to \vartheta)) \to ((\varphi \to \psi) \to (\varphi \to \vartheta))$;

A3) $((\varphi \to \psi) \to \varphi) \to \varphi$;

A4) $\bot \to \varphi$.

A5) $\forall x(\varphi \to \psi) \to (\forall x\varphi \to \forall x\psi)$;

A6) $\varphi \to \forall y\varphi$, gdy $y \notin FV(\varphi)$;

A7) $\forall x\varphi \to \varphi[t/x]$;

A8) $\forall x(x = x)$;

A9) $\forall x\forall y(x = y \to \varphi[z := y] \to \varphi[z := x])$.

Powyżej, symbole φ, ψ i ϑ oznaczają dowolne formuły sygnatury Σ. A zatem nasz zbiór aksjomatów nie składa się z dziewięciu formuł, ale jest zbiorem nieskończonym. (Niemniej, jest to *efektywny* (lub *obliczalny*) zbiór aksjomatów, bo istnieje łatwe kryterium pozwalające odróżnić co jest aksjomatem a co nie jest.)

Przyjmujemy dwie reguły wnioskowania. Pierwszą jest *reguła odrywania (modus ponens)*, zapisywana tak:

$$\frac{\alpha,\ \alpha \to \beta}{\beta}$$

a drugą *reguła generalizacji*, którą zapisujemy tak:

$$\frac{\varphi}{\forall x\varphi}$$

Reguły należy rozumieć z grubsza w ten sposób: jeśli formuły nad kreską (przesłanki reguły) są już wyprowadzone, to można wywnioskować formułę pod kreską (konkluzję). Z grubsza, bo stosowanie reguły generalizacji jest ograniczone (patrz niżej).

Definicja 12.1 *Dowodem formalnym* lub *wyprowadzeniem* (w sensie Hilberta) formuły φ ze zbioru założeń Γ nazywamy dowolny ciąg formuł $\varphi_1, \varphi_2, \ldots, \varphi_n$, spełniający następujące warunki:

- $\varphi_n = \varphi$;
- Dla każdego $i \leq n$ zachodzi jedna z czterech możliwości:
  - φ_i jest aksjomatem;
  - $\varphi_i \in \Gamma$;
  - są takie $\ell, j < i$, że $\varphi_\ell = \varphi_j \to \varphi_i$ (tj. φ_i jest otrzymana przez zastosowanie modus ponens do formuł φ_ℓ i φ_j);
  - jest takie $j < i$, że $\varphi_i = \forall x\varphi_j$, oraz $x \notin FV(\Gamma)$ (tj. φ_i jest otrzymana przez generalizację formuły φ_j).

Jeśli taki dowód istnieje, to piszemy $\Gamma \vdash \varphi$ i mówimy, że φ można udowodnić z założeń Γ, lub że φ jest (syntaktyczną) *konsekwencją* Γ. Jeśli $\emptyset \vdash \varphi$ to piszemy tylko $\vdash \varphi$ i mówimy, że φ jest *twierdzeniem* rachunku predykatów.

Jak widać z powyższej definicji, stosowanie reguły generalizacji jest ograniczone do przypadku, gdy zmienna wiązana kwantyfikatorem nie jest wolna w założeniach ze zbioru Γ. Dlatego regułę generalizacji należy ściślej zapisać tak:

$$\frac{\varphi}{\forall x\varphi} \quad (x \notin FV(\Gamma))$$

Reguła generalizacji może być stosowana m.in. wtedy, gdy Γ nie ma zmiennych wolnych (w szczególności, gdy $\Gamma = \emptyset$).

Uwaga: Ten, kto nie chce utożsamiać alfa-równoważnych formuł, musi nieco zmodyfikować powyższy system dowodzenia, poprzez pewne wzmocnienie reguły generalizacji. Na przykład takie:

$$\frac{\varphi(x \wr y)}{\forall x\varphi} \quad (y \notin FV(\Gamma))$$

Przykład 12.2

(1) Następujący ciąg formuł (gdzie jako β można przyjąć jakąkolwiek formułę) jest dowodem formuły $\alpha \to \alpha$ (ze zbioru pustego).

1. $(\alpha \to (\beta \to \alpha) \to \alpha) \to ((\alpha \to (\beta \to \alpha)) \to (\alpha \to \alpha))$ (aksjomat A2);
2. $\alpha \to (\beta \to \alpha) \to \alpha$ (aksjomat A1);
3. $(\alpha \to (\beta \to \alpha)) \to (\alpha \to \alpha)$ (odrywanie (2) od (1);
4. $\alpha \to (\beta \to \alpha)$ (aksjomat A1);
5. $\alpha \to \alpha$ (odrywanie (4) od (3)).

(2) Następujący ciąg formuł jest dowodem formuły γ ze zbioru $\{\alpha \to \beta, \beta \to \gamma, \alpha\}$:

1. $\alpha \to \beta$ (założenie);
2. α (założenie);
3. β (odrywanie (2) od (1));
4. $\beta \to \gamma$ (założenie);
5. γ (odrywanie (3) od (4)).

(3) Następujący ciąg formuł jest dowodem formuły $\forall x \forall y P(x,y) \to \forall x P(x,x)$:

1. $\forall y P(x,y) \to P(x,x)$ (aksjomat A7);
2. $\forall x(\forall y P(x,y) \to P(x,x))$ (generalizacja (1));
3. $\forall x(\forall y P(x,y) \to P(x,x)) \to (\forall x \forall y P(x,y) \to \forall x P(x,x))$ (aksjomat A5);
4. $\forall x \forall y P(x,y) \to \forall x P(x,x)$ (odrywanie (2) od (3)).

Tradycyjna definicja hilbertowskiego dowodu formalnego jako ciągu formuł jest często wygodna i dlatego wciąż popularna. Ale oczywiście każdy dowód w sensie Hilberta ukrywa w sobie strukturę drzewa. Na przykład dowód z Przykładu 12.2(3) możemy napisać tak:

$$\dfrac{\dfrac{\alpha \qquad \alpha \to \beta}{\beta} \qquad \beta \to \gamma}{\gamma}$$

Lemat 12.3 *Przypuśćmy, że ciąg formuł $\varphi_1, \varphi_2, \ldots, \varphi_n$ jest dowodem formuły φ_n ze zbioru założeń Γ. Jeśli $x, y \notin FV(\Gamma)$, a na dodatek $y \notin FV(\varphi_i)$ dla $i = 1, \ldots, n$, to ciąg formuł $\varphi_1[x := y], \varphi_2[x := y], \ldots, \varphi_n[x := y]$ jest dowodem formuły $\varphi_n[x := y]$ z założeń Γ.*

Dowód: Łatwa indukcja ze względu na n. W przypadku gdy w ostatnim kroku mamy generalizację ze względu na $z \neq x$ należy zauważyć, że $(\forall z\, \varphi_i)[x := y] = \forall z\, \varphi_i[x := y]$. Przy generalizacji ze względu na x korzystamy z tego, że $\forall x\, \varphi_i =_\alpha \forall y\, \varphi_i[x := y]$. ∎

Lemat 12.4 (o osłabianiu) *Jeśli $\Gamma \vdash \varphi$ oraz $\Gamma \subseteq \Gamma'$ to także $\Gamma' \vdash \varphi$.*

Dowód: Postępujemy przez indukcję ze względu na długość dowodu formuły φ z założeń Γ. Jedyny nieoczywisty przypadek, to zastosowanie generalizacji. Może się bowiem zdarzyć, że $\varphi_n = \forall x \varphi_i$ gdzie $x \notin FV(\Gamma)$ ale $x \in FV(\Gamma')$. Ale wtedy możemy skorzystać z Lematu 12.3 i skonstruować taki dowód, w którym zamiast formuły φ_i jest $\varphi_i[x := y]$, a zmienna y nie należy do $FV(\Gamma')$. Możemy teraz generalizować i otrzymujemy formułę $\forall y\, \varphi_i[x := y] =_\alpha \forall x \varphi_i$. ∎

Twierdzenie 12.5 (o poprawności) *Jeżeli $\Gamma \vdash \varphi$ to $\Gamma \models \varphi$. W szczególności, każde twierdzenie jest tautologią.*

Dowód: Dowód przebiega przez indukcję ze względu na długość najkrótszego dowodu formalnego formuły φ ze zbioru Γ. Oczywiście $\Gamma \models \varphi$, dla każdego $\varphi \in \Gamma$. Z Faktów 10.5 i 10.4 natychmiast wynika, że aksjomaty (A1)–(A4) są tautologiami. Pozostałe aksjomaty są tautologiami na mocy Faktu 11.1 i Faktu 11.2. Jeśli więc dowód formuły φ ma długość 1, to teza zachodzi. Jeśli dowód składa się z więcej niż jednego kroku, to φ otrzymano za pomocą jednej z reguł dowodzenia.

Załóżmy najpierw, że ostatnią zastosowaną regułą bylo *modus ponens*. To znaczy, że wcześniej w naszym dowodzie pojawiły się pewna formuła ψ i formuła $\psi \to \varphi$. Z założenia indukcyjnego mamy $\Gamma \models \psi$ i $\Gamma \models \psi \to \varphi$. Stąd oczywiście wynika, że $\Gamma \models \varphi$. (Każde wartościowanie spełniające ψ i $\psi \to \varphi$ musi też spełniać φ.)

Jeżeli ostatnia była reguła generalizacji, to φ jest postaci $\forall x\psi$, gdzie $x \notin FV(\Gamma)$. Z założenia indukcyjnego $\Gamma \models \psi$. Chcemy pokazać, że wtedy $\Gamma \models \forall x\psi$. Niech więc $\mathcal{A}, v \models \Gamma$. Wtedy także $\mathcal{A}, v_x^a \models \Gamma$, przy dowolnym $a \in |\mathcal{A}|$, bo $x \notin FV(\Gamma)$. Zatem zawsze $\mathcal{A}, v_x^a \models \psi$, więc $v(\forall x\psi) = \min\{v_x^a(\psi) : a \in |\mathcal{A}|\} = 1$. ∎

Konwencja notacyjna: napis Γ, φ oznacza zbiór $\Gamma \cup \{\varphi\}$, podobnie zamiast $\{\varphi, \beta, \gamma\} \vdash \delta$ piszemy $\varphi, \beta, \gamma \vdash \delta$, itd.

Twierdzenie 12.6 (o dedukcji) *Warunki* $\Gamma \vdash \varphi \to \psi$ *i* $\Gamma, \varphi \vdash \psi$ *są równoważne.*

Dowód: Implikacja z lewej do prawej jest oczywista: aby z Γ, φ otrzymać ψ, należy wyprowadzić $\varphi \to \psi$ i oderwać φ. Dowód implikacji odwrotnej przebiega przez indukcję ze względu na długość dowodu formalnego formuły ψ ze zbioru Γ, φ.

Jeśli ψ jest aksjomatem lub $\psi \in \Gamma$, to dowód formuły $\varphi \to \psi$ jest otrzymany przez oderwanie ψ od aksjomatu $\psi \to (\varphi \to \psi)$. Jeśli $\psi = \varphi$ to korzystamy z tego, że $\vdash \varphi \to \varphi$ (Przykład 12.2(1)) i tym bardziej $\Gamma \vdash \varphi \to \varphi$.

Przypuśćmy więc, że ψ otrzymano przez odrywanie. Znaczy to, że $\Gamma, \varphi \vdash \vartheta \to \psi$ i $\Gamma, \varphi \vdash \vartheta$ dla pewnej formuły ϑ, i że odpowiednie dowody są krótsze. Z założenia indukcyjnego otrzymujemy, że formuły $\varphi \to (\vartheta \to \psi)$ i $\varphi \to \vartheta$ mają dowody ze zbioru Γ. Aby otrzymać dowód dla $\varphi \to \psi$, należy te dwie formuły kolejno oderwać od aksjomatu (A2) w postaci $(\varphi \to (\vartheta \to \psi)) \to ((\varphi \to \vartheta) \to (\varphi \to \psi))$.

Pozostaje przypadek, gdy dowód formuły ψ ze zbioru Γ, φ kończy się generalizacją, tj gdy $\psi = \forall x\vartheta$ i mamy krótszy dowód dla $\Gamma, \varphi \vdash \vartheta$, a przy tym $x \notin FV(\Gamma, \varphi)$, tj. $x \notin FV(\Gamma)$ i $x \notin FV(\varphi)$. Z założenia indukcyjnego mamy $\Gamma \vdash \varphi \to \vartheta$. Ponieważ $x \notin FV(\Gamma)$, więc z pomocą reguły generalizacji otrzymamy $\Gamma \vdash \forall x(\varphi \to \vartheta)$. Użyjemy następnie aksjomatu (A5) i odrywania aby dostać $\Gamma \vdash \forall x\varphi \to \forall x\vartheta$. Stąd z pomocą (A1) uzyskamy $\Gamma \vdash \varphi \to (\forall x\varphi \to \forall x\vartheta)$. Ponieważ $x \notin \mathrm{FV}(\varphi)$, więc formuła $\varphi \to \forall x\varphi$ jest aksjomatem (A6), w szczególności ma dowód z Γ. Pozostaje wykonać dwa odrywania od aksjomatu (A2) w postaci $(\varphi \to (\forall x\varphi \to \forall x\vartheta)) \to ((\varphi \to \forall x\varphi) \to (\varphi \to \forall x\vartheta))$ i dostajemy $\Gamma \vdash \varphi \to \forall x\vartheta$. ∎

Twierdzenie o dedukcji pozwala na łatwe przekonanie się o istnieniu pewnych dowodów bez ich wypisywania w całości. Na przykład, łatwo przekonać się o tym, że dla dowolnych α, β i γ, formuła

$$(\alpha \to (\beta \to \gamma)) \to (\beta \to (\alpha \to \gamma)),$$

jest twierdzeniem, sprawdzając tylko, że $\alpha \to (\beta \to \gamma), \beta, \alpha \vdash \gamma$. W podobny sposób otrzymamy

Lemat 12.7 *Jeśli* $\Gamma \vdash \varphi \to \psi$ *oraz* $\Gamma \vdash \psi \to \vartheta$ *to także* $\Gamma \vdash \varphi \to \vartheta$*.*

Dowód: Najpierw pokazujemy, że $\varphi \to \psi, \psi \to \vartheta, \varphi \vdash \vartheta$, a potem trzy razy stosujemy Twierdzenie 12.6. ∎

13 Co można udowodnić

Naszym celem jest twierdzenie odwrotne do Twierdzenia 12.5, czyli twierdzenie o pełności. Dowód tego twierdzenia wymaga pewnych przygotowań. Musimy pokazać, że w naszym systemie wnioskowania potrafimy wyprowadzić cały szereg rozmaitych formuł. Następująca teraz seria lematów jest poświęcona temu zadaniu. Najpierw zrobimy pewną dość oczywistą obserwację. Będziemy z niej wielokrotnie korzystać.

Lemat 13.1 *Jeśli* $\Gamma \vdash \alpha$ *oraz* $\Delta, \alpha \vdash \beta$ *to* $\Gamma, \Delta \vdash \beta$*.*

Dowód: Łatwy. Ale trzeba użyć Lematu 12.4. ∎

Następujący lemat stwierdza, że najważniejsze własności negacji są twierdzeniami naszego systemu:

Lemat 13.2 *Dla dowolnych formuł* α *i* β*:*

1. $\alpha \vdash \neg\neg\alpha$;
2. $\neg\alpha, \alpha \vdash \beta$;
3. $\neg\neg\alpha \vdash \alpha$;
4. $\alpha \to \beta, \neg\alpha \to \beta \vdash \beta$.

Dowód:

(1) Ponieważ $\alpha, \alpha \to \bot \vdash \bot$, więc z twierdzenia o dedukcji $\alpha \vdash (\alpha \to \bot) \to \bot$, czyli właśnie $\alpha \vdash \neg\neg\alpha$.

(2) Ponieważ $\neg\alpha, \alpha \vdash \bot$ (zob. (1)) oraz $\bot \vdash \beta$, więc także $\neg\alpha, \alpha \vdash \beta$.

(3) Korzystając z (2) otrzymamy $\neg\neg\alpha, \neg\alpha \vdash \alpha$, i z twierdzenia o dedukcji $\neg\neg\alpha \vdash \neg\alpha \to \alpha$. Prawo Peirce'a w postaci $((\alpha \to \bot) \to \alpha) \to \alpha$ jest aksjomatem, a więc przez odpowiednie odrywanie dostajemy (3).

(4) Zauważmy na początek, że $\alpha \to \beta, \neg\alpha \to \beta, \beta \to \bot \vdash \alpha \to \bot$ (Lemat 12.7). Ponieważ $\alpha \to \bot$ to to samo co $\neg\alpha$, i oczywiście $\alpha \to \beta, \neg\alpha \to \beta, \beta \to \bot \vdash \neg\alpha \to \beta$, więc łatwo otrzymujemy $\alpha \to \beta, \neg\alpha \to \beta, \beta \to \bot \vdash \beta$. Z twierdzenia o dedukcji otrzymamy $\alpha \to \beta, \neg\alpha \to \beta \vdash \neg\neg\beta$. Na mocy (3) wnioskujemy, że $\alpha \to \beta, \neg\alpha \to \beta \vdash \beta$. ∎

Kolejne dwa lematy dotyczą własności koniunkcji i alternatywy, wyprowadzalnych w naszym systemie dowodzenia. (Przypomnijmy, że $\alpha \wedge \beta = \neg(\alpha \to (\neg\beta))$ oraz $\alpha \vee \beta = \neg\alpha \to \beta$.)

Lemat 13.3 *Dla dowolnych formuł α, β i γ:*

1. $\alpha \wedge \beta \vdash \alpha$;
2. $\alpha, \beta \vdash \alpha \wedge \beta$;
3. $\alpha \wedge \beta \vdash \beta$;
4. $\gamma \to \alpha, \gamma \to \beta \vdash \gamma \to \beta \wedge \alpha$;

Dowód: (1) Ten dowód poprowadzimy „od końca", aby pokazać metody jakimi można się posługiwać dla konstrukcji wyprowadzenia. Mamy udowodnić, że $\alpha \wedge \beta \vdash \alpha$, czyli, że $(\alpha \to \neg\beta) \to \bot \vdash \alpha$. Nie bardzo wiadomo jak się do tego zabrać, bo stosując *modus ponens* do formuły $(\alpha \to \neg\beta) \to \bot$ na pewno nie dostaniemy formuły α. Ale wiemy, że wystarczy udowodnić $\neg\neg\alpha$, czyli formułę $(\alpha \to \bot) \to \bot$. Mamy też twierdzenie o dedukcji, czyli wystarczy jeśli otrzymamy to:

$$(\alpha \to \neg\beta) \to \bot, \alpha \to \bot \vdash \bot. \tag{2}$$

Mamy szansę wyprowadzić $\bot$, jeśli wyprowadzimy przesłankę któregoś z założeń. Spróbujmy więc, czy uda się uzyskać to:

$$(\alpha \to \neg\beta) \to \bot, \alpha \to \bot \vdash \alpha \to \neg\beta. \tag{3}$$

Z twierdzenia o dedukcji wiemy, że wystarczy coś takiego:

$$(\alpha \to \neg\beta) \to \bot, \alpha \to \bot, \alpha \vdash \neg\beta. \tag{4}$$

Ale to już jest łatwe. Wystarczy się powołać na Lemat 13.2(2). Stąd dostaniemy (3) i dalej (2).

Uwaga: to co zrobiliśmy, to nie była konstrukcja dowodu w sensie Hilberta. To było tylko uzasadnienie, że taki dowód można skonstruować.

(2) Ponieważ $\alpha, \beta, \alpha \to (\beta \to \bot) \vdash \bot$, więc $\alpha, \beta \vdash (\alpha \to (\beta \to \bot)) \to \bot$, a to jest właśnie to co trzeba.

(3) Ponieważ $\neg\beta \vdash \alpha \to \neg\beta$ więc $(\alpha \to \neg\beta) \to \bot, \neg\beta \vdash \bot$, a stąd $\alpha \wedge \beta \vdash \neg\neg\beta$, i pozostaje skorzystać z Lematu 13.2(3).

(4) Łatwa konsekwencja części (2). ∎

Lemat 13.4 *Dla dowolnych formuł α, β i γ:*

1. $\beta \vdash \alpha \vee \beta$;
2. $\alpha \vdash \alpha \vee \beta$;
3. $\alpha \to \gamma, \beta \to \gamma \vdash \alpha \vee \beta \to \gamma$;

Dowód: (1) i (2) Oczywiste, bo przecież $\alpha \vee \beta = \neg\alpha \to \beta$.

(3) Mamy $\alpha \to \gamma, \beta \to \gamma, \neg\alpha \to \beta \vdash \alpha \to \gamma$ oraz $\alpha \to \gamma, \beta \to \gamma, \neg\alpha \to \beta \vdash \neg\alpha \to \gamma$. Z Lematu 13.2(4) wynika $\alpha \to \gamma, \beta \to \gamma, \neg\alpha \to \beta \vdash \gamma$, i pozostaje zastosować twierdzenie o dedukcji. ∎

Lemat 13.5 *Dla dowolnych formuł α, β, γ:*

1. $(\alpha \vee \beta) \wedge \gamma \vdash (\alpha \wedge \gamma) \vee (\beta \wedge \gamma)$;
2. $(\alpha \wedge \gamma) \vee (\beta \wedge \gamma) \vdash (\alpha \vee \beta) \wedge \gamma$;
3. $(\alpha \wedge \beta) \vee \gamma \vdash (\alpha \vee \gamma) \wedge (\beta \vee \gamma)$;
4. $(\alpha \vee \gamma) \wedge (\beta \vee \gamma) \vdash (\alpha \wedge \beta) \vee \gamma$;

Dowód: (1) Ponieważ $\alpha, \gamma \vdash \alpha \wedge \gamma$, więc $\alpha, \gamma, \alpha \wedge \gamma \to \bot \vdash \bot$, czyli $\gamma, \alpha \wedge \gamma \to \bot \vdash \alpha \to \bot$. Stąd $\gamma, \alpha \wedge \gamma \to \bot, (\alpha \to \bot) \to \beta \vdash \beta$. Ponieważ oczywiście $\gamma, \ldots \vdash \gamma$, więc $\gamma, \alpha \wedge \gamma \to \bot, \alpha \vee \beta \vdash \beta \wedge \gamma$, bo przecież $(\alpha \to \bot) \to \beta = \alpha \vee \beta$. Z twierdzenia o dedukcji $\gamma, (\alpha \to \bot) \to \beta \vdash \neg(\alpha \wedge \gamma) \to (\beta \wedge \gamma)$, a ponieważ $\gamma \wedge (\alpha \vee \beta) \vdash \gamma$ i $\gamma \wedge (\alpha \vee \beta) \vdash \alpha \vee \beta$ (Lemat 13.3(1, 3)), więc ostatecznie $\gamma \wedge (\alpha \vee \beta) \vdash (\alpha \wedge \gamma) \vee (\beta \wedge \gamma)$.

(2) Ponieważ $\alpha \vdash \alpha \vee \beta$ i $\gamma \vdash \gamma$, więc $\alpha \wedge \gamma \vdash (\alpha \vee \beta) \wedge \gamma$, z Lematu 13.3(1–2). Podobnie $\beta \wedge \gamma \vdash (\alpha \vee \beta) \wedge \gamma$, więc teza wynika z Lematu 13.4(3).

(3) Podobnie jak w (2) należy najpierw zauważyć, że $\gamma \vdash (\alpha \vee \gamma) \wedge (\beta \vee \gamma)$, bo $\gamma \vdash (\alpha \vee \gamma)$ i $\gamma \vdash (\beta \vee \gamma)$, oraz że $\alpha \wedge \beta \vdash (\alpha \vee \gamma) \wedge (\beta \vee \gamma)$, bo $\alpha \wedge \beta \vdash (\alpha \vee \gamma)$ i $\alpha \wedge \beta \vdash (\alpha \vee \gamma)$. Następnie używamy Lematu 13.4(3).

(4) Ponieważ $\gamma \vee \alpha = (\gamma \to \bot) \to \alpha$, więc $\gamma \vee \alpha, \neg\gamma \vdash \alpha$. Podobnie $\gamma \vee \beta, \neg\gamma \vdash \beta$, więc $\gamma \vee \alpha, \gamma \vee \beta, \neg\gamma \vdash \alpha \wedge \beta$. Stąd także $\alpha \vee \gamma, \beta \vee \gamma \vdash \neg\gamma \to (\alpha \wedge \beta)$. Zatem $(\alpha \vee \gamma) \wedge (\beta \vee \gamma) \vdash \gamma \vee (\alpha \wedge \beta)$. ∎

Algebra Lindenbauma

Ustalmy pewien zbiór formuł Γ. Określimy relację $\sim_\Gamma$ w zbiorze wszystkich formuł:

$$\alpha \sim_\Gamma \beta \qquad \text{wtedy i tylko wtedy, gdy} \qquad \Gamma \vdash \alpha \leftrightarrow \beta.$$

Łatwo widzieć, że relacja $\sim_\Gamma$ jest relacją równoważności, bo wiemy już, że:

$$\vdash \alpha \to \alpha \qquad\qquad \text{oraz} \qquad\qquad \alpha \to \beta, \beta \to \gamma \vdash \alpha \to \gamma$$

Ponadto relacja $\sim_\Gamma$ zachowuje spójniki logiczne $\vee$, $\wedge$ i $\neg$:

Lemat 13.6 *Jeżeli $\alpha \sim_\Gamma \alpha'$ i $\beta \sim_\Gamma \beta'$ to:*

1. $\neg\alpha \sim_\Gamma \neg\alpha'$;
2. $\alpha \vee \beta \sim_\Gamma \alpha' \vee \beta'$;
3. $\alpha \wedge \beta \sim_\Gamma \alpha' \wedge \beta'$;

Dowód: Ćwiczenie.[9] ∎

Powyższy lemat pozwala na określenie operacji $\cup$, $\cap$, $-$ na klasach abstrakcji relacji $\sim_\Gamma$. Definiujemy je tak:

$$\begin{aligned} [\alpha]_{\sim_\Gamma} \cup [\beta]_{\sim_\Gamma} &= [\alpha \vee \beta]_{\sim_\Gamma}; \\ [\alpha]_{\sim_\Gamma} \cap [\beta]_{\sim_\Gamma} &= [\alpha \wedge \beta]_{\sim_\Gamma}; \\ -[\alpha]_{\sim_\Gamma} &= [\neg\alpha]_{\sim_\Gamma}. \end{aligned}$$

Lemat 13.6 gwarantuje poprawność tych operacji (wynik nie zależy od wyboru reprezentantów). Możemy też określić relację $\leq$ w zbiorze $\mathcal{F}_\Sigma/_{\sim_\Gamma}$:

$$[\alpha]_{\sim_\Gamma} \leq [\beta]_{\sim_\Gamma} \quad \text{wtedy i tylko wtedy, gdy} \quad \Gamma \vdash \alpha \to \beta.$$

[9] Wskazówka: najpierw należy pokazać, że relacja $\sim_\Gamma$ zachowuje implikację.

Lemat 13.7 *Relacja $\leq$ jest częściowym porządkiem w $\mathcal{F}_\Sigma/_{\sim_\Gamma}$, a dla dowolnych α i β:*

- $[\alpha \vee \beta]_{\sim_\Gamma} = \sup\{[\alpha]_{\sim_\Gamma}, [\beta]_{\sim_\Gamma}\}$;
- $[\alpha \wedge \beta]_{\sim_\Gamma} = \inf\{[\alpha]_{\sim_\Gamma}, [\beta]_{\sim_\Gamma}\}$.

Dowód: Łatwy wniosek z Lematów 13.3 i 13.4. ∎

Lemat 13.8 *Dla dowolnej formuły α,*

- $\alpha \sim_\Gamma \top$ *wtedy i tylko wtedy, gdy* $\Gamma \vdash \alpha$. *W szczególności* $\alpha \sim_\Gamma \top$ *dla* $\alpha \in \Gamma$.
- $\alpha \sim_\Gamma \bot$ *wtedy i tylko wtedy, gdy* $\Gamma \vdash \neg\alpha$.

Dowód: Łatwy. ∎

Fakt 13.9 *Zbiór $\mathcal{F}_\Sigma/_{\sim_\Gamma}$, złożony ze wszystkich klas abstrakcji relacji $\sim_\Gamma$, tworzy algebrę Boole'a z operacjami $\cup$, $\cap$, $-$, i stałymi $\mathbf{0} = [\bot]_{\sim_\Gamma}$ oraz $\mathbf{1} = [\top]_{\sim_\Gamma}$.*

Dowód: Na mocy Lematu 13.7, algebra $\mathcal{F}_\Sigma/_{\sim_\Gamma}$ jest kratą. Dystrybutywność wynika z Lematu 13.5. Pozostaje sprawdzić następujące cztery warunki:

1. $\alpha \vee \bot \sim_\Gamma \alpha$;
2. $\alpha \wedge \top \sim_\Gamma \alpha$;
3. $\alpha \vee \neg\alpha \sim_\Gamma \top$;
4. $\neg\alpha \wedge \alpha \sim_\Gamma \bot$;

Warunki (1) i (2) wynikają stąd, że $\bot$ i $\top$ są odpowiednio najmniejszym i największym elementem. Warunek (3) wynika z równości $\alpha \vee \neg\alpha = \neg\alpha \to \neg\alpha$, a warunek (4) z oczywistego związku $\alpha, \alpha \to \bot \vdash \bot$. ∎

Definicja 13.10 Algebrę Boole'a $\mathfrak{L}_\Sigma = \langle \mathcal{F}_\Sigma/_{\sim_\Gamma}, \cup, \cap, -, \mathbf{0}, \mathbf{1}\rangle$, o której mowa w Fakcie 13.9, nazywamy *algebrą Lindenbauma*.

14 Twierdzenia o pełności

Twierdzenie 14.1 (o pełności rachunku zdań) *Załóżmy, że Γ jest zbiorem formuł zdaniowych i niech α będzie formułą zdaniową. Jeżeli $\Gamma \models \alpha$ to $\Gamma \vdash \alpha$. W szczególności, każda tautologia zdaniowa jest twierdzeniem.*

Dowód: Przypuśćmy, że $\Gamma \not\vdash \alpha$. Oznacza to w szczególności, że $\Gamma, \neg\alpha \not\vdash \bot$, czyli, że $[\neg\alpha]_{\sim_\Gamma} \neq 0$. Na mocy Lematu 4.13, istnieje więc taki filtr maksymalny F, że $[\neg\alpha]_{\sim_\Gamma} \in F$. Filtr F jest też ultrafiltrem (Lemat 4.15), tj. dla dowolnej formuły φ, jedna z klas $[\varphi]_{\sim_\Gamma}$ lub $[\neg\varphi]_{\sim_\Gamma}$ musi należeć do F. Zauważmy jeszcze, że dla $\gamma \in \Gamma$ mamy $[\gamma]_{\sim_\Gamma} = 1 \in F$.

Niech teraz w będzie takim wartościowaniem zdaniowym, że dla dowolnego symbolu zdaniowego p

$$w(p) = 1 \quad \text{wtedy i tylko wtedy, gdy} \quad [p]_{\sim_\Gamma} \in F.$$

Udowodnimy, że dla dowolnej formuły φ zachodzi

$$w(\varphi) = 1 \quad \text{wtedy i tylko wtedy, gdy} \quad [\varphi]_{\sim_\Gamma} \in F. \tag{5}$$

Postępujemy przez indukcję ze względu na budowę formuły φ. Oczywiste jest, że warunek (5) zachodzi dla $\bot$ i dla symboli zdaniowych. Niech więc $\varphi = \psi \to \vartheta$.

Załóżmy, że $w(\psi \to \vartheta) = 1$. Wtedy albo $w(\psi) = 0$ albo $w(\vartheta) = 1$. W pierwszym przypadku, z założenia indukcyjnego $[\psi]_{\sim_\Gamma} \notin F$, czyli $[\neg\psi]_{\sim_\Gamma} \in F$, bo F jest ultrafiltrem. W drugim przypadku $[\vartheta]_{\sim_\Gamma} \in F$. Ale ponieważ zarówno $[\neg\psi]_{\sim_\Gamma} \leq [\psi \to \vartheta]_{\sim_\Gamma}$ jak też i $[\vartheta]_{\sim_\Gamma} \leq [\psi \to \vartheta]_{\sim_\Gamma}$, więc w obu przypadkach $[\psi \to \vartheta]_{\sim_\Gamma} \in F$.

Na odwrót, niech $[\psi \to \vartheta]_{\sim_\Gamma} \in F$ i niech $w(\psi) = 1$. Z założenia indukcyjnego $[\psi]_{\sim_\Gamma} \in F$, a więc także $[\vartheta]_{\sim_\Gamma} \in F$, bo $[\psi]_{\sim_\Gamma} \cap [\psi \to \vartheta]_{\sim_\Gamma} \leq [\vartheta]_{\sim_\Gamma}$. Na mocy założenia indukcyjnego wnioskujemy, że $w(\vartheta) = 1$.

Z warunku (5) wynika, że $w(\alpha) = 0$, oraz $w(\gamma) = 1$ dla $\gamma \in \Gamma$. Stąd $\Gamma \not\models \alpha$. ∎

Pełność rachunku predykatów

Dowód twierdzenia o pełności rachunku predykatów jest nieco trudniejszy od rozważanego poprzednio dowodu pełności rachunku zdań, chociaż zasadniczy tok rozumowania pozostaje podobny. Znowu będzie mowa o algebrze Lindenbauma. Tym razem jednak nie szukamy zerojedynkowego wartościowania zdaniowego, ale musimy skonstruować odpowiednią strukturę algebraiczną. Na dodatek, byle jaki filtr maksymalny tym razem niestety nie wystarczy. Potrzebny będzie filtr spełniający dość wyrafinowane zachcianki i to stanowi zasadniczą trudność techniczną. Można ją rozwiązać na dwa sposoby. Pierwszy jest bardziej elegancki, bo czysto algebraiczny, i opiera się na twierdzeniu, które tradycyjnie wbrew gramatyce nazywane jest „lematem Rasiowa-Sikorski”. Ten sposób ma jednak pewną wadę: stosuje się tylko do przeliczalnych języków (sygnatura i zbiór zmiennych muszą być przeliczalne). Drugi sposób, zwany „metodą stałych Henkina” nie ma tego ograniczenia.

Definicja 14.2 Zbiór formuł Γ jest *sprzeczny* wtedy i tylko wtedy, gdy $\Gamma \vdash \bot$.

Lemat 14.3 *Jeśli $\Gamma \vdash \varphi$ to istnieje taki skończony podzbiór $\Gamma_0 \subseteq \Gamma$, że $\Gamma_0 \vdash \varphi$. W szczególności, jeśli Γ jest sprzeczny to ma skończony podzbiór, który też jest sprzeczny.*

Dowód: W wyprowadzeniu występuje tylko skończenie wiele formuł ze zbioru Γ. Reszta jest niepotrzebna. ∎

Lemat 14.4 *Załóżmy, że $x \notin FV(\Gamma)$ i że stała c nie występuje w formule φ ani w żadnej formule ze zbioru Γ. Jeśli $\Gamma \vdash \varphi[x := c]$, to $\Gamma \vdash \forall x\varphi$.*

Dowód: Przez indukcję ze względu na długość wyprowadzenia $\Gamma \vdash \varphi[x := c]$ pokazujemy, że $\Gamma \vdash \varphi$. A teraz wystarczy zastosować regułę generalizacji. ∎

Jeśli Σ jest dowolną sygnaturą, a C jest zbiorem stałych nie należących do Σ, to przez $\Sigma \cup C$ oznaczymy sygnaturę powstającą z Σ przez dołączenie stałych z C do Σ.

Twierdzenie 14.5 (Lemat Henkina) *Niech Δ będzie dowolnym niesprzecznym zbiorem formuł sygnatury Σ i niech $x \notin FV(\Delta)$. Istnieje zbiór stałych C i funkcja $c : \mathcal{F}_{\Sigma\cup C} \xrightarrow{1-1} C$, o tej własności, że zbiór formuł*

$$\Delta \cup \{\varphi[x := c(\varphi)] \to \forall x\varphi \mid \varphi \in \mathcal{F}_{\Sigma\cup C}\}$$

jest niesprzeczny.

Dowód:* Konstruujemy przez indukcję sygnatury Σ_n, zbiory stałych C_n i przyporządkowania $c_n : \mathcal{F}_{\Sigma_n} \xrightarrow{1-1} C_n$ w ten sposób aby niesprzeczne były zbiory formuł:

$$\Delta_n = \Delta \cup \{\varphi[x := c_n(\varphi)] \to \forall x\varphi \mid \varphi \in \mathcal{F}_{\Sigma_n}\},$$

Zaczynamy od $\Sigma_0 = \Sigma$. Dla jednostajności naszej konstrukcji przyjmijmy jeszcze $C_{-1} = \emptyset$ i $c_{-1} = \emptyset$. Gdy $n > 0$ to przyjmujemy $\Sigma_n = \Sigma_{n-1} \cup C_{n-1}$.

Dla dowolnego n określamy teraz C_n i $c_n : \mathcal{F}_{\Sigma_n} \xrightarrow{1-1} C_n$ jakkolwiek, byle tylko zachodziły zawierania $C_{n-1} \subseteq C_n$ i $c_{n-1} \subseteq c_n$. (Trzeba po prostu wziąć odpowiednio dużo nowych stałych). Należy teraz wykazać niesprzeczność zbioru Δ_n.

Jeśli zbiór Δ_n jest sprzeczny to na mocy Lematu 14.3 ma skończony podzbiór sprzeczny. Wybierzmy minimalny taki podzbiór Θ. Skoro sam zbiór Δ jest niesprzeczny to $\Theta - \Delta \neq \emptyset$ i możemy napisać

$$\Theta = \Theta' \cup \{\varphi[x := c_n(\varphi)] \to \forall x\varphi\}.$$

Zbiór Θ' jest niesprzeczny z minimalności Θ. Mamy teraz $\Theta', \varphi[x := c_n(\varphi)] \to \forall x\varphi \vdash \bot$, skąd wynika $\Theta' \vdash \neg(\varphi[x := c_n(\varphi)] \to \forall x\varphi)$. Ale to oznacza, że $\Theta' \vdash \varphi[x := c_n(\varphi)]$ oraz $\Theta' \vdash \neg\forall x\varphi$.

Stała $c_n(\varphi)$ nie występuje w żadnej formule zbioru Θ', ani też w samej formule φ. Z lematu 14.4 wynika więc sprzeczność.

Ostatecznie określamy $C = \bigcup\{C_n \mid n \in \mathbb{N}\}$ oraz $c = \bigcup\{c_n \mid n \in \mathbb{N}\}$. Niesprzeczność naszego zbioru formuł wynika z Lematu 14.3: w przeciwnym razie któryś ze zbiorów Δ_n byłby już sprzeczny. ∎

Twierdzenie 14.6 (o pełności rachunku predykatów) *Jeżeli* $\Gamma \models \phi$ *to* $\Gamma \vdash \phi$.

Dowód: Przypuśćmy, że $\Gamma \not\vdash \phi$. To znaczy, że zbiór $\Delta = \Gamma \cup \{\neg\phi\}$ jest niesprzeczny. Bez straty ogólności możemy założyć, że istnieje nieskończenie wiele zmiennych nie występujących w Γ ani w φ. (W przeciwnym razie należy „przenumerować” zmienne, używając np. tylko tych o parzystych numerach.) Niech C będzie zbiorem stałych, o którym mowa w lemacie Henkina i niech c będzie odpowiednią funkcją. Tak samo jak w przypadku zdaniowym, rozważamy algebrę Lindenbauma $\mathfrak{L}$, tym razem jednak jest to algebra formuł sygnatury $\Sigma \cup C$. Poza tym definicja algebry $\mathfrak{L}$ jest dokładnie taka sama.

Ważną własnością naszej algebry jest to, że kwantyfikatory ogólne odpowiadają kresom dolnym. Dokładniej, dla dowolnej formuły φ zachodzi równość

$$[\forall x\varphi]_{\sim_\Gamma} = \inf\{[\varphi[x := t]]_{\sim_\Gamma} \mid t \in \mathcal{T}_{\Sigma\cup C}\}. \qquad (*)$$

Istotnie, $[\forall x\varphi]_{\sim_\Gamma} \leq [\varphi[x := t]_{\sim_\Gamma}$, bo $\vdash \forall x\varphi \to \varphi[x := t]$ (aksjomat A7). Pozostaje więc sprawdzić, że $[\forall x\varphi]_{\sim_\Gamma}$ jest najmniejszym ograniczeniem. Przypuśćmy więc, że dla dowolnego t zachodzi $\Gamma \vdash \zeta \to \varphi[x := t]$. Wtedy w szczególności $\Gamma \vdash \zeta \to \varphi[x := y]$, gdzie zmienna y nie występuje ani w ζ ani w φ ani w Γ (także jako zmienna związana). A zatem $\Gamma, \zeta \vdash \varphi[x := y]$ i stosując generalizację dostajemy $\Gamma, \zeta \vdash \forall y\varphi[x := y]$. Korzystając z aksjomatu (A7) otrzymujemy $\Gamma, \zeta \vdash \varphi[x := y][y := x]$. Ale y nie występuje gdzie nie trzeba, więc przy podstawieniu $\varphi[x := y]$ nie uległa zmianie żadna zmienna związana. Zmienna y nie pojawiła się też w zasięgu kwantyfikatora wiążącego x. Zatem kolejne podstawienie $\varphi[x := y][y := x]$ także nie spowoduje wymiany zmiennych i w konsekwencji otrzymamy $\varphi[x := y][y := x] = \varphi$. Ostatecznie $[\zeta]_{\sim_\Gamma} \leq [\forall x\varphi]_{\sim_\Gamma}$.

Podobnie jak w dowodzie Twierdzenia 14.1 będzie nam teraz potrzebny filtr maksymalny F, do którego należy $[\neg\phi]_{\sim_\Gamma}$. Ale teraz to jeszcze za mało. Chcemy, aby w naszym filtrze były też wszystkie elementy postaci $[\varphi[x := c(\varphi)] \to \forall x\varphi]_{\sim_\Gamma}$ dla $\varphi \in \mathcal{F}_{\Sigma\cup C}$. Istnienie takiego filtru wynika z lematu Henkina. Nietrudno bowiem zauważyć, że jeśli $\Gamma \cup \Pi$ jest zbiorem niesprzecznym, to $\{[\vartheta]_{\sim_\Gamma} \mid \vartheta \in \Pi\}$ jest scentrowanym podzbiorem algebry $\mathfrak{L}$.

Potrzebujemy teraz modelu $\mathcal{A}$ i wartościowania v, spełniającego wszystkie formuły z naszego filtru F. Na początek rozpatrzmy strukturę termów $\mathcal{T}$, w której relacje są określone tak: Jeśli r jest k-argumentowym symbolem relacyjnym sygnatury Σ, to dla dowolnych termów $t_1, \ldots, t_k$:

$$\langle t_1, \ldots, t_k \rangle \in r^{\mathcal{T}} \qquad \text{wtedy i tylko wtedy, gdy} \qquad [r(t_1, \ldots, t_k)]_{\sim_\Gamma} \in F.$$

W strukturze $\mathcal{T}$ określimy teraz relację $\approx$ warunkiem:

$$t \approx u \qquad \text{wtedy i tylko wtedy, gdy} \qquad [t = u]_{\sim_\Gamma} \in F.$$

Ta relacja jest kongruencją w $\mathcal{T}$. Wynika to z aksjomatów (A8) i (A9). Możemy więc utworzyć strukturę ilorazową $\mathcal{A} = \mathcal{T}/_{\approx}$. I to jest właśnie model, którego potrzebujemy. W tym modelu określimy wartościowanie kanoniczne $v(t) = [t]_{\approx}$. Naszym celem jest teraz następujaca równoważność:

$$\mathcal{A}, v \models \psi \qquad \text{wtedy i tylko wtedy, gdy} \qquad [\psi]_{\sim_\Gamma} \in F. \qquad (**)$$

Dowód $(**)$ przebiega oczywiście przez indukcję ze względu na φ i jest podobny do dowodu równoważności (5) występującej w dowodzie Twierdzenia 14.1. Jedyny istotnie nowy przypadek występuje wtedy gdy $\psi = \forall x \varphi$.

Załóżmy najpierw, że $[\forall x\varphi]_{\sim_\Gamma} \in F$. Na mocy $(*)$ mamy wtedy $[\varphi[x := t]]_{\sim_\Gamma} \in F$ dla wszystkich t. Z założenia indukcyjnego $\mathcal{A}, v \models \varphi[x := t]$ dla dowolnego t. Ale $v(\varphi[x := t]) = v_x^{v(t)}(\varphi) = v_x^{[t]_\approx}(\varphi)$, a każdy element $\mathcal{A}$ jest postaci $[t]_{\approx}$. Stąd $\mathcal{A}, v_x^a \models \varphi$ dla wszystkich $a \in \mathcal{A}$ czyli $\mathcal{A}, v \models \forall x\varphi$.

Implikacja odwrotna jest kluczową częścią naszego dowodu. Przypuśćmy, że $\mathcal{A}, v \models \forall x\varphi$. To znaczy, że $\mathcal{A}, v \models \varphi[x := t]$ dla dowolnego termu t i z założenia indukcyjnego mamy $[\varphi[x := t]]_{\sim_\Gamma} \in F$. W szczególności $[\varphi[x := c(\varphi)]]_{\sim_\Gamma} \in F$. Ale do filtru F należy też klasa $[\varphi[x := c(\varphi)] \to \forall x\varphi]_{\sim_\Gamma}$. Stąd $[\forall x\varphi]_{\sim_\Gamma}$ też musi należeć do filtru.

Z warunku $(**)$ wynika od razu, że $\mathcal{A}, v \models \Gamma$ oraz $\mathcal{A}, v \models \neg\phi$, w szczególności $\mathcal{A}, v \not\models \phi$. A zatem $\Gamma \not\models \phi$. ∎

Wniosek 14.7 *Zbiór formuł Γ jest spełnialny wtedy i tylko wtedy, gdy jest niesprzeczny.*

15 Teoria modeli

Wniosek 15.1 (Dolne twierdzenie Skolema-Löwenheima) *Jeśli zbiór formuł Γ (w języku przeliczalnym) jest spełnialny to ma model przeliczalny.*

Dowód: Skoro Γ jest spełnialny, to jest niesprzeczny, więc $\Gamma \not\vdash \bot$. Powtarzając konstrukcję z dowodu Twierdzenia 14.6 otrzymamy strukturę termów $\mathcal{A}$, spełniającą Γ. Ważne, że nośnik $|\mathcal{A}| = \mathcal{T}_\Sigma/_\approx$ jest zbiorem przeliczalnym. ∎

Uwaga: W przypadku języka nieprzeliczalnego, gdy zbiór wszystkich termów jest mocy $\mathfrak{m}$, można uogólnić powyższy wniosek tak: każdy spełnialny zbiór formuł ma model mocy co najwyżej $\mathfrak{m}$.

Twierdzenie Skolema-Löwenheima było kiedyś nazywane „paradoksem Skolema-Löwenheima", z powodu swoich zaskakujących konsekwencji. Przykładowo, wiadomo, że liczb rzeczywistych jest nieprzeliczalnie wiele. Na mocy dolnego twierdzenia Skolema-Löwenheima istnieje jednak przeliczalna struktura $\langle \mathbb{P}, +, \cdot, 0, 1\rangle$ spełniająca dokładnie te same zdania co zwykłe ciało liczb rzeczywistych $\langle \mathbb{R}, +, \cdot, 0, 1\rangle$. Struktury te są więc całkowicie nieodróżnialne z punktu widzenia zwykłej logiki predykatów.

Jeszcze bardziej zaskakujące jest istnienie przeliczalnego modelu dla teorii mnogości, tj. struktury przeliczalnej $\langle S, \epsilon, =\rangle$, w której dwuargumentowa relacja ϵ spełnia dokładnie wszystkie twierdzenia wynikające z aksjomatów teorii mnogości. W strukturze S prawdziwe są np. zdania stwierdzające istnienie zbiorów nieprzeliczalnych!

Twierdzenie 15.2 (o zwartości) *Jeśli $\Gamma \models \varphi$ to istnieje taki skończony podzbiór $\Gamma_0 \subseteq \Gamma$, że $\Gamma_0 \models \varphi$. W szczególności, jeśli każdy skończony podzbiór zbioru Γ jest spełnialny, to Γ jest spełnialny.*

Dowód: Natychmiastowa konsekwencja twierdzenia o pełności i Lematu 14.3. ∎

Klasycznym zastosowaniem twierdzenia o zwartości jest następujące twierdzenie:

Fakt 15.3 *Jeśli Γ ma dowolnie duże modele skończone to ma model nieskończony.*

Dowód: Niech $\{y_i : i \in \mathbb{N}\}$ będą różnymi zmiennymi, nie występującymi wolno w formułach z Γ. Rozpatrzmy zbiór formuł $\Delta = \{$„$\neg(y_i = y_j)$" $\mid i \neq j\}$.

Jeśli $\Gamma_0 \subseteq \Gamma \cup \Delta$ jest zbiorem skończonym to istnieje takie j, że dla $i \geq j$, zmienne y_i nie występują w Γ_0. Istnieją takie $\mathcal{A}, v$, że $\mathcal{A}, v \models \Gamma$ oraz $|\mathcal{A}|$ ma co najmniej j różnych elementów $a_0, \ldots, a_{j-1}$. Niech v' będzie takim wartościowaniem, że $v'(y_i) = a_i$ dla $i < j$ oraz $v'(x) = v(x)$ dla innych zmiennych x. Wtedy oczywiście $\mathcal{A}, v' \models \Gamma_0$.

Zatem każdy skończony podzbiór zbioru $\Gamma \cup \Delta$ jest spełnialny. Z twierdzenia o zwartości także zbiór $\Gamma \cup \Delta$ jest spełnialny. Ale cały zbiór $\Gamma \cup \Delta$ może być jednocześnie spełniony tylko w nieskończonym modelu: jeżeli bowiem $\mathcal{A}, v \models \Delta$, to wszystkie elementy $v(y_i)$ muszą być różne. ∎

Wniosek 15.4 *Nie istnieje taki zbiór zdań Γ, że dla dowolnego modelu $\mathcal{A}$:*

$$\mathcal{A} \models \Gamma \quad \textit{wtedy i tylko wtedy, gdy} \quad |\mathcal{A}| \textit{ jest skończone.}$$

Dowód: Prosta konsekwencja Faktu 15.3 ∎

A zatem nie można zdefiniować pojęcia skończoności przy pomocy formuł pierwszego rzędu.

Twierdzenie 15.5 (Twierdzenie Skolema-Löwenheima)

Jeśli zbiór formuł Γ (w przeliczalnym języku) ma model nieskończony to ma modele dowolnej mocy nieskończonej.

Dowód: Podobny do dowodu Faktu 15.3. Niech $\mathfrak{m} \geq \aleph_0$. Dodajemy do języka zbiór Y nowych zmiennych (lub stałych) o mocy $\mathfrak{m}$ i rozszerzamy zbiór Γ o aksjomaty „$\neg(y = y')$” dla wszystkich różnych $y, y' \in Y$. Z twierdzenia o zwartości wynika[10] istnienie modelu mocy co najmniej $\mathfrak{m}$. Co więcej, nasz rozszerzony zbiór ma wyłącznie takie modele. Z dolnego twierdzenia Skolema-Löwenheima (por. uwaga po Tw. 15.1) wnioskujemy, że istnieje model mocy równej $\mathfrak{m}$. ∎

Ciekawym przykładem zastosowania twierdzenia o zwartości jest twierdzenie Kfoury'ego-Parka, które przedstawimy na przykładzie (dla uniknięcia długich definicji). Rozpatrzmy następujący prosty **while**-program $P(x)$ z jedną zmienną wejściową x:

$$\mathbf{begin}\ y := c;\ \mathbf{while}\ x \neq y\ \mathbf{do}\ y := f(y)\ \mathbf{end}$$

Niech $\mathcal{A} = \langle A, f^{\mathcal{A}}, c^{\mathcal{A}} \rangle$ będzie dowolnym modelem sygnatury $\{f, c\}$. Dla wartości wejściowej $a \in A$ rozpatrzmy obliczenie $P(a)$ programu P. Nietrudno zauważyć, że obliczenie $P(a)$ nie zatrzymuje się wtedy i tylko wtedy, gdy $\mathcal{A}, a \models \Delta$, gdzie Δ składa się ze wszystkich formuł postaci $\neg(x = f(f(\cdots(c)\cdots))))$.

Szukamy takiego zbioru zdań Γ aby dla dowolnej interpretacji $\mathcal{A} = \langle A, f^{\mathcal{A}}, c^{\mathcal{A}} \rangle$ zachodziło wynikanie:

$$\text{Jeśli} \quad \mathcal{A} \models \Gamma \quad \text{to obliczenie } P(a) \text{ zatrzymuje się dla dowolnego } a \in |\mathcal{A}|.$$

Przypuśćmy, że Γ jest zbiorem zdań o tej własności. Wówczas zbiór formuł

$$\Gamma \cup \{\text{„}\neg(x = f^n(c))\text{”} \mid n \in \mathbb{N}\}$$

jest zbiorem niespełnialnym. (Oczywiście $f^n(c)$ oznacza term $f(f(\cdots(c)\cdots))$, w którym symbol f występuje n razy.)

Na mocy twierdzenia o zwartości istnieje skończony podzbiór niespełnialny. Istnieje więc takie n, że dla dowolnego modelu $\mathcal{A}$ spełniającego Γ, i każdego elementu $a \in |\mathcal{A}|$ zachodzi jedna z równości $a = c$, $a = f^{\mathcal{A}}(c^{\mathcal{A}})$, …, $a = f^{\mathcal{A}}(f^{\mathcal{A}}(\cdots f^{\mathcal{A}}(c^{\mathcal{A}})\cdots))$, gdzie liczba iteracji funkcji $f^{\mathcal{A}}$ jest co najwyżej równa n.

[10] I to się nazywa górnym twierdzeniem Skolema-Löwenheima.

A zatem każdy warunek wystarczający na to aby **while**-program był *totalny*, tj. zawsze się zatrzymywał, w istocie określa także pewne ograniczenie górne na liczbę kroków tego programu. Nie sposób więc skonstruować zbioru formuł rachunku predykatów, którego spełnienie byłoby warunkiem *koniecznym* i wystarczającym na zatrzymywanie się danego **while**-programu, przy dowolnej interpretacji występujących w nim symboli funkcyjnych.

Dowolność interpretacji, o której mowa powyżej, jest istotna. Jeśli się ograniczyć na przykład do *standardowego modelu arytmetyki*, tj. struktury $\mathcal{N} = \langle \mathbb{N}, +, \cdot, 0, \boldsymbol{s} \rangle$ (gdzie $\boldsymbol{s}$ oznacza funkcję nastepnika) to mamy następujące

Twierdzenie 15.6 (tw. Gödla o reprezentacji) *Dla dowolnego* **while**-*programu* $P(x)$, *w którym występują tylko symbole funkcyjne* $+, \cdot, 0, \boldsymbol{s}$ *i symbol równości, istnieje taka formuła* φ, *że*

$$\mathcal{N}, n \models \varphi \quad \textit{wtedy i tylko wtedy, gdy obliczenie } P(n) \textit{ zatrzymuje się w } \mathcal{N}$$

Konflikt pomiędzy treścią Twierdzenia 15.6 i twierdzenia Kfoury'ego i Parka jest pozorny. Okazuje się, że w języku rachunku predykatów nie można zdefiniować jednoznacznie standardowego modelu arytmetyki.

Fakt 15.7 *Istnieje* niestandardowy model arytmetyki, *tj. przeliczalna struktura*

$$\mathcal{M} = \langle \mathbb{M}, \oplus, \otimes, \mathbf{0}, \mathbf{S} \rangle,$$

o takich własnościach:

- *dla dowolnego zdania* φ, *warunki* $\mathcal{N} \models \varphi$ *i* $\mathcal{M} \models \varphi$ *są równoważne;*
- *struktury* $\mathcal{N}$ *i* $\mathcal{M}$ *nie są izomorficzne.*

Dowód: Niech $\mathbf{Th}(\mathcal{N})$ oznacza zbiór wszystkich zdań prawdziwych w modelu $\mathcal{N}$ i niech Δ składa się ze wszystkich formuł postaci $x \neq \boldsymbol{s}(\boldsymbol{s}(\ldots \boldsymbol{s}(\mathbf{0}) \ldots))$. Nietrudno pokazać, że każdy skończony podzbiór zbioru $\mathbf{Th}(\mathcal{N}) \cup \Delta$ jest spełnialny w modelu $\mathcal{N}$ przez dostatecznie dużą wartość x. Zatem całość jest spełnialna w pewnym modelu $\mathcal{M}$ przez pewne wartościowanie v. Wtedy $\mathcal{M}$ spełnia te same zdania co $\mathcal{N}$, ale element $v(x)$ nie ma odpowiednika w modelu $\mathcal{N}$, bo każdy element $\mathcal{N}$ można otrzymać z zera za pomocą nastepnika. ∎

Arytmetyka Peano

Skoro nie można zdefiniować jednoznacznie modelu standardowego, może można chociaż, za pomocą odpowiednich aksjomatów, scharakteryzować zdania które są w nim prawdziwe? Przez PA (od „Peano Arithmetics”) oznaczymy zbiór aksjomatów złożony z formuł:

- $\forall x \neg(\boldsymbol{s}(x) = 0)$
- $\forall x \forall y\, (\boldsymbol{s}(x) = \boldsymbol{s}(y) \to x = y)$;
- $\forall x\, (x + 0 = x)$;
- $\forall x \forall y\, (x + \boldsymbol{s}(y) = \boldsymbol{s}(x + y))$;
- $\forall x\, (x \cdot 0 = 0)$;
- $\forall x \forall y\, (x \cdot \boldsymbol{s}(y) = (x \cdot y) + x)$;
- $\forall x\, (\varphi(x) \to \varphi(\boldsymbol{s}(x))) \to (\varphi(0) \to \forall x\, \varphi(x))$,

gdzie $\varphi(x)$ może być dowolną formułą. Pierwsze dwa aksjomaty mówią, że zero nie jest następnikiem żadnej liczby i że operacja następnika jest różnowartościowa (to gwarantuje nieskończoność). Kolejne dwa aksjomaty stanowią indukcyjną definicję dodawania, a następne dwa — indukcyjną definicję mnożenia. Na końcu mamy nie pojedyńczy aksjomat, ale schemat aksjomatu, nazywany schematem *indukcji*. Zatem zbiór aksjomatów Peano jest w istocie nieskończony. Ale zbiór ten jest *efektywny*, tj. można mechanicznie ustalić co jest aksjomatem a co nie jest.

Oczywiście standardowy model arytmetyki jest modelem arytmetyki Peano:

$$\mathcal{N} \models \mathrm{PA}.$$

Inaczej mówiąc, wszystkie konsekwencje aksjomatów Peano (twierdzenia teorii PA) są prawdziwe w standardowym modelu. A na odwrót? Kiedyś przypuszczano, że PA jest *teorią zupełną*, tj. że każde zdanie prawdziwe w $\mathcal{N}$ jest twierdzeniem arytmetyki Peano.[11]

Przypuszczenie to okazało sie fałszywe dzieki odkryciu dokonanemu przez Gödla, a mianowicie dzięki metodzie *arytmetyzacji* (numeracji Gödla). Wszystkie symbole języka arytmetyki numerujemy na przykład tak:

Symbol:	0	$\boldsymbol{s}$	$+$	$\cdot$	$\perp$	$\to$	$=$	$\forall$	$($	$)$	x_0	x_1	$\ldots$
Numer:	1	2	3	4	5	6	7	8	9	10	11	12	$\ldots$

Każdemu ciągowi znaków, w tym każdej formule, dowodowi itp., można teraz przypisać kod liczbowy. Jeśli przez $\#a$ oznaczymy numer znaku a, to kodem napisu „$a_1 a_2 \ldots a_n$” jest liczba

$$\mathit{Kod}(a_1 a_2 \ldots a_n) = 2^{\#a_1} 3^{\#a_2} 5^{\#a_3} 7^{\#a_4} \cdots p_n^{\#a_n},$$

[11] Teoria T jest *zupełna*, wtedy i tylko wtedy, gdy dla dowolnego zdania φ (w języku tej teorii) zachodzi albo $T \models \varphi$ albo $T \models \neg\varphi$. Zbiór zdań prawdziwych w ustalonym modelu jest oczywiście zawsze teorią zupełną.

gdzie p_n oznacza n-tą liczbę pierwszą. Odkrycie Gödla oparte jest na obserwacji, że własności formuł arytmetyki mogą być wyrażane w języku samej arytmetyki jako teorioliczbowe własności kodów. Zamiast np. mówić o własnościach formuły „$\forall x_0((x_1 + x_0 = 0) \to \bot)$”, można mówić o własnościach jej kodu, tj. liczby

$$Kod(\forall x_0((x_1 + x_0 = 0) \to \bot)) = 2^8 3^{11} 5^9 7^9 11^{12} 13^3 17^{11} 19^7 23^1 29^{10} 31^6 37^5 41^{10}.$$

Niech symbol $\underline{n}$ oznacza term $\boldsymbol{s}^n(\mathbf{0})$. Oczywiście znaczeniem termu $\underline{n}$ w $\mathcal{N}$ jest liczba n. Można teraz np. napisać taką formułę $F(x)$ o jednej zmiennej wolnej x, że dla dowolnego $n \in \mathbb{N}$ zachodzi równoważność

$\mathcal{N} \models F(\underline{n})$, wtedy i tylko wtedy, gdy n jest numerem formuły o jednej zmiennej wolnej.

Oczywiście wiele rozmaitych własności syntaktycznych możemy wyrazić w podobny sposób.[12] Przydatna jest np. formuła $T(x, y)$ o takiej własności:

$\mathcal{N} \models T(\underline{n}, \underline{m})$, wtedy i tylko wtedy, gdy

- m jest numerem pewnej formuły $\zeta(x)$ o jednej zmiennej wolnej,
- n jest numerem zdania $\zeta(\underline{m})$.

W skrócie zapiszemy to tak:

$\mathcal{N} \models T(\underline{n}, \underline{m})$, wtedy i tylko wtedy, gdy n jest numerem zdania $\varphi_m(\underline{m})$.

Nie każda własność formuł może jednak być wyrażona w jezyku arytmetyki.

Twierdzenie 15.8 (tw. Tarskiego o niewyrażalności prawdy) *Nie istnieje formuła wyrażająca prawdziwość formuł w standardowym modelu, tj. taka formuła $P(x)$, że*

$\mathcal{N} \models P(\underline{n})$ *wtedy i tylko wtedy, gdy n jest numerem zdania prawdziwego w $\mathcal{N}$.*

Dowód: Dowód twierdzenia polega na wyrażeniu znanego *paradoksu kłamcy*[13] w języku arytmetyki. Rozpatrzmy nastepującą formułę

$$T(x) \equiv \exists y\, (T(y, x) \land \neg P(y)).$$

Wówczas $\mathcal{N} \models T(\underline{n})$ wtedy i tylko wtedy, gdy

- n jest numerem pewnej formuły $\zeta(x)$ o jednej zmiennej wolnej,
- zdanie $\zeta(\underline{n})$ jest fałszywe w $\mathcal{N}$.

[12]Istotnym problemem technicznym jest konieczność kodowania ciągów liczb o nieznanej z góry długości. Używa się w tym celu tzw. chińskiego twierdzenia o resztach.

[13]Stwierdzenie *„To zdanie jest fałszywe”* nie może być ani prawdziwe ani fałszywe.

Mniej ściśle, ale prościej:

$$\mathcal{N} \models T(\underline{n}) \text{ wtedy i tylko wtedy, gdy } \mathcal{N} \models \neg\varphi_n(\underline{n}).$$

Formuła $T(x)$ też ma numer, powiedzmy, że $T(x) = \varphi_k(x)$. A zatem możemy napisać

$$\mathcal{N} \models T(\underline{k}) \text{ wtedy i tylko wtedy, gdy } \mathcal{N} \models \neg\varphi_k(\underline{k}).$$

Możemy to napisac z czystym sumieniem, bo warunek

- k jest numerem pewnej formuły $\zeta(x)$ o jednej zmiennej wolnej,

jest oczywiście spełniony. Ale przecież $\varphi_k(\underline{k})$ to właśnie formuła $T(\underline{k})$. A zatem:

$$\mathcal{N} \models T(\underline{k}) \text{ wtedy i tylko wtedy, gdy } \mathcal{N} \models \neg T(\underline{k}).$$

No jasne: zdanie $T(\underline{k})$ stwierdza *„Ja jestem fałszywe!”* Ze znanym skutkiem ... ∎

Twierdzenie Gödla o niezupełności arytmetyki otrzymamy po nieznacznej modyfikacji powyższego rozumowania. Zamiast niemożliwego do zdefiniowania pojecia prawdy, użyjemy wyrażalnej własności „mieć dowód w arytmetyce Peano”. Otrzymamy w ten sposób zdanie Z, które mówi: *„Ja nie mam dowodu!”*.

Uwaga: Twierdzenie Tarskiego podpowiada rozstrzygnięcie paradoksu kłamcy: Problem leży w niewyrażalności pojęcia “zdania prawdziwego”, także w języku polskim. A skoro pytamy o własność, której nie umiemy zdefiniować, to nie dziwmy się, że nie ma odpowiedzi.

Twierdzenie 15.9 (Gödla o niezupełności) *Istnieje takie zdanie Z w języku arytmetyki, że $PA \nvdash Z$ i $PA \nvdash \neg Z$.*

Dowód: Postępujemy jak w poprzednim dowodzie, używając formuły $P'(x)$ o własności

$\mathcal{N} \models P'(\underline{n})$ wtedy i tylko wtedy, gdy n jest numerem zdania, które ma dowód w PA.

Otrzymamy w końcu taką konkluzję: $\mathcal{N} \models T(\underline{k})$ wtedy i tylko wtedy, gdy PA $\nvdash T(\underline{k})$.

Przyjmując $Z = T(\underline{k})$, wnioskujemy, że ani Z ani $\neg Z$ nie może mieć dowodu w PA. Założenie $PA \vdash Z$ prowadzi do sprzeczności, bo jeśli $PA \vdash Z$ to $\mathcal{N} \models Z$. Ale założenie $PA \vdash \neg Z$ też prowadzi do sprzeczności, bo mielibyśmy z jednej strony $\mathcal{N} \models \neg Z$, a z drugiej $\mathcal{N} \models Z$. Uwaga: nietrudno zauważyć, że $\mathcal{N} \models Z$. ∎

Istota twierdzenia Gödla polega nie na tym, że akurat PA jest niezupełna. Jeśli zbiór aksjomatów PA rozszerzymy do innego efektywnego (!) zbioru aksjomatów prawdziwych w $\mathcal{N}$ to nadal będzie istniało zdanie niezależne od tych aksjomatów. Dowód pozostanie prawie bez zmian. A więc nie tylko PA, ale w ogóle każda efektywnie zadana teoria musi byc niezupełna (jeśli tylko jest dostatecznie silna na to, aby dało sie w niej zinterpretować pojęcia arytmetyczne).

Niech m będzie numerem zdania „$0 = \boldsymbol{s}(0)$” i niech **Con** oznacza zdanie $\neg P'(\underline{m})$. Zdanie to wyraża niesprzeczność arytmetyki Peano. Rozumowanie podobne do użytego w dowodzie Twierdzenia 15.9 można... sformalizować w języku arytmetyki. Otrzymamy konkluzję:

$$\mathrm{PA} \vdash \mathbf{Con} \to Z,$$

gdzie Z jest zdaniem z Twierdzenia 15.9. W konsekwencji otrzymujemy:

Wniosek 15.10 $\mathrm{PA} \nvdash \mathbf{Con}$.

Niesprzeczności arytmetyki Peano nie można udowodnić na gruncie samej arytmetyki Peano (chyba, że sama PA jest sprzeczna). Ta sama konkluzja dotyczy każdej dostatecznie silnej teorii.

16 Rachunek sekwentów

Systemy dowodzenia twierdzeń w stylu Hilberta mają pewne zasadnicze wady, związane z tym, że głównym narzędziem w dowodach jest reguła odrywania. Po pierwsze, aby wyprowadzić nawet prostą formułę, trzeba się posługiwać formułami znacznie dłuższymi. Po drugie, jeśli chcemy wyprowadzić formułę φ z pomocą reguły *modus ponens*

$$\frac{\vartheta \to \varphi, \quad \vartheta}{\varphi}$$

to musimy odgadnąć odpowiednią formułę ϑ, a nie mamy żadnych wskazówek co do jej kształtu. Dlatego systemy w stylu Hilberta nie bardzo nadają się do zastosowań praktycznych.

Rachunek sekwentów, wprowadzony przez Gentzena, to inna metoda wyprowadzania twierdzeń rachunku zdań lub rachunku predykatów. Definicje są tu może trochę bardziej skomplikowane, ale za to otrzymujemy system, który prawie nie ma wyżej wymienionych wad.

Sekwentem nazywamy napis postaci „$\alpha_1, \ldots, \alpha_n \vdash \beta_1, \ldots, \beta_m$”, gdzie α_i i β_j są formułami. Zarówno m jak n może być zerem. Piszemy Γ, α lub α, Γ na oznaczenie ciągu powstałego z Γ przez dopisanie α z tyłu lub z przodu. Za *aksjomaty* rachunku sekwentów przyjmujemy sekwenty postaci:

$$\bot \vdash \qquad \text{oraz} \quad \alpha \vdash \alpha,$$

gdzie α jest dowolną formułą. (Czasem mogą być przyjmowane jeszcze dodatkowe aksjomaty, o czym za chwilę. Inne definicje pozostają wtedy bez zmian.) Reguły rachunku sekwentów pozwalają z jednego lub dwóch sekwentów-przesłanek wyprowadzić sekwent-konkluzję.

Reguły przedstawione w tabeli tworzą system dowodzenia sekwentów w rachunku predykatów. Tym razem nie korzystamy z możliwości definiowania jednych spójników przez inne. Warto bowiem zobaczyć w jaki sposób reguły rachunku sekwentów związane są z podstawowymi własnościami poszczególnych operacji logicznych.

Uwaga: (1) Reguły rachunku sekwentów można znacznie uprościć, jeśli zamiast ciągów formuł używa się zbiorów, tj. zaniedbuje się kolejność formuł w ciągu i ich powtórzenia. Wtedy reguły wymiany i skracania stają się niepotrzebne.
(2) Nasze reguły dla kwantyfikatorów są sformułowane w taki sposób, aby nie było potrzebne utożsamianie ze sobą alfa-równoważnych formuł. Przy założeniu alfa-konwersji reguły (P$\forall$) i (L$\exists$) można napisać tak:

$$(\mathrm{P}\forall)\ \frac{\Gamma \vdash \varphi, \Sigma}{\Gamma \vdash \forall x\varphi, \Sigma}\ (x \notin FV(\Gamma \cup \Sigma)) \qquad\qquad (\mathrm{L}\exists)\ \frac{\Gamma, \varphi \vdash \Sigma}{\Gamma, \exists x\varphi \vdash \Sigma}\ (x \notin FV(\Gamma \cup \Sigma))$$

Definicja 16.1 *Wyprowadzeniem* sekwentu $\Gamma \vdash \Sigma$ (lub *dowodem formalnym w sensie Gentzena*) nazywamy drzewo[14] etykietowane sekwentami, które ma następujące własności:

- Korzeń ma etykietę $\Gamma \vdash \Sigma$;
- Etykiety liści są aksjomatami;
- Etykiety pozostałych wierzchołków są otrzymane z etykiet ich następników (synów) za pomocą reguł wnioskowania.

Uwaga: Jak powiedziano wyżej, kolejność i powtórzenia formuł w sekwencie są istotne. W praktyce jednak czasami pomijamy kroki dowodu polegające na zastosowaniu reguł wymiany i skracania (stosujemy je domyślnie), por. Przykład 16.2(3).

Przykład 16.2 **(1)** Dowód sekwentu $\vdash ((p \to q) \to p) \to p$:

$$\cfrac{\cfrac{\cfrac{\cfrac{p \vdash p}{p \vdash q, p}\,(\mathrm{PO})}{\vdash p \to q, p}\,(\mathrm{PI}) \qquad p \vdash p}{(p \to q) \to p \vdash p}\,(\mathrm{LI})}{\vdash ((p \to q) \to p) \to p}\,(\mathrm{PI})$$

[14]Tym razem należy sobie wyobrażać, że drzewo ma korzeń na dole a liście na górze.

Osłabianie: $\dfrac{\Gamma \vdash \Sigma}{\Gamma, \alpha \vdash \Sigma}(\mathrm{LO})$ $\dfrac{\Gamma \vdash \Sigma}{\Gamma \vdash \alpha, \Sigma}(\mathrm{PO})$

Wymiana: $\dfrac{\Gamma, \varphi, \psi, \Delta \vdash \Sigma}{\Gamma, \psi, \varphi, \Delta \vdash \Sigma}(\mathrm{LW})$ $\dfrac{\Gamma \vdash \Delta, \varphi, \psi, \Sigma}{\Gamma \vdash \Delta, \psi, \varphi, \Sigma}(\mathrm{PW})$

Skracanie: $\dfrac{\Gamma, \varphi, \varphi \vdash \Sigma}{\Gamma, \varphi \vdash \Sigma}(\mathrm{LS})$ $\dfrac{\Gamma \vdash \varphi, \varphi, \Sigma}{\Gamma \vdash \varphi, \Sigma}(\mathrm{PS})$

Negacja: $\dfrac{\Gamma \vdash \alpha, \Sigma}{\Gamma, \neg\alpha \vdash \Sigma}(\mathrm{LN})$ $\dfrac{\Gamma, \alpha \vdash \Sigma}{\Gamma \vdash \neg\alpha, \Sigma}(\mathrm{PN})$

Koniunkcja: $\dfrac{\Gamma, \alpha \vdash \Sigma}{\Gamma, \alpha \wedge \beta \vdash \Sigma}(\mathrm{LK})\ \dfrac{\Gamma, \beta \vdash \Sigma}{\Gamma, \alpha \wedge \beta \vdash \Sigma}$ $\dfrac{\Gamma \vdash \alpha, \Sigma \quad \Gamma \vdash \beta, \Sigma}{\Gamma \vdash \alpha \wedge \beta, \Sigma}(\mathrm{PK})$

Alternatywa: $\dfrac{\Gamma, \alpha \vdash \Sigma \quad \Gamma, \beta \vdash \Sigma}{\Gamma, \alpha \vee \beta \vdash \Sigma}(\mathrm{LA})$ $\dfrac{\Gamma \vdash \alpha, \Sigma}{\Gamma \vdash \alpha \vee \beta, \Sigma}(\mathrm{PA})\dfrac{\Gamma \vdash \beta, \Sigma}{\Gamma \vdash \alpha \vee \beta, \Sigma}$

Implikacja: $\dfrac{\Gamma \vdash \alpha, \Sigma \quad \Gamma, \beta \vdash \Sigma}{\Gamma, \alpha \to \beta \vdash \Sigma}(\mathrm{LI})$ $\dfrac{\Gamma, \alpha \vdash \beta, \Sigma}{\Gamma \vdash \alpha \to \beta, \Sigma}(\mathrm{PI})$

Kwantyfikator ogólny: $\dfrac{\Gamma, \varphi[x{:=}t] \vdash \Sigma}{\Gamma, \forall x\varphi \vdash \Sigma}(\mathrm{L}\forall)$ $\dfrac{\Gamma \vdash \varphi[x{:=}y], \Sigma}{\Gamma \vdash \forall x\varphi, \Sigma}(\mathrm{P}\forall)$

Kwantyfikator szczegółowy: $\dfrac{\Gamma, \varphi[x{:=}y] \vdash \Sigma}{\Gamma, \exists x\varphi \vdash \Sigma}(\mathrm{L}\exists)$ $\dfrac{\Gamma \vdash \varphi[x{:=}t], \Sigma}{\Gamma \vdash \exists x\varphi, \Sigma}(\mathrm{P}\exists)$

Cięcie: $\dfrac{\Gamma \vdash \alpha, \Sigma \quad \Gamma, \alpha \vdash \Sigma}{\Gamma \vdash \Sigma}(\mathrm{Ciach!})$

Reguły (P∀) i (L∃) mają następujące ograniczenie: zmienna y nie może występować wolno w żadnej formule należącej do $\Gamma \cup \Sigma$.

(2) Dowód sekwentu $p \vee \neg p$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{p \vdash p}{\vdash \neg p, p}\,(\mathrm{PN})}{\vdash p \vee \neg p, p}\,(\mathrm{PA})}{\vdash p, p \vee \neg p}\,(\mathrm{PW})}{\vdash p \vee \neg p, p \vee \neg p}\,(\mathrm{PA})}{\vdash p \vee \neg p}\,(\mathrm{PS})$$

(3) Ten sam dowód zapisany w uproszczony sposób (domyślna wymiana i skracanie):

$$\dfrac{\dfrac{\dfrac{p \vdash p}{\vdash \neg p, p}\,(\mathrm{PN})}{\vdash p \vee \neg p, p}\,(\mathrm{PA})}{\vdash p \vee \neg p}\,(\mathrm{PW} + \mathrm{PA} + \mathrm{PS})$$

(4) Dowód sekwentu $\forall x P(x) \vdash \neg \exists x (P(x) \to \bot)$. Kilka razy domyślnie użyto reguły (LW).

$$(\mathrm{PN})\dfrac{(\mathrm{L}\exists)\dfrac{(\mathrm{L}\forall)\dfrac{(\mathrm{LI})\dfrac{P(x) \vdash P(x) \qquad \dfrac{\bot \vdash}{\bot, P(x) \vdash}\,(\mathrm{LO})}{P(x), P(x) \to \bot \vdash}}{\forall x P(x), P(x) \to \bot \vdash}}{\forall x P(x), \exists x (P(x) \to \bot) \vdash}}{\forall x P(x) \vdash \neg \exists x (P(x) \to \bot)}$$

Uwaga: Ponieważ mamy do dyspozycji reguły osłabiania, więc reguły (PK), (LA), (LI) oraz reguła cięcia mogą być równie dobrze wybrane tak:

$$\dfrac{\Gamma \vdash \alpha, \Sigma \quad \Delta \vdash \beta, \Pi}{\Gamma, \Delta \vdash \alpha \wedge \beta, \Sigma, \Pi}(\mathrm{PK1}) \qquad \dfrac{\Gamma, \alpha \vdash \Sigma \quad \Delta, \beta \vdash \Pi}{\Gamma, \Delta, \alpha \vee \beta \vdash \Sigma, \Pi}(\mathrm{LA1})$$

$$\dfrac{\Gamma \vdash \alpha, \Sigma \quad \Delta, \beta \vdash \Pi}{\Gamma, \Delta, \alpha \to \beta \vdash \Sigma, \Pi}(\mathrm{LI1}) \qquad \dfrac{\Gamma \vdash \alpha, \Sigma \quad \Delta, \alpha \vdash \Pi}{\Gamma, \Delta \vdash \Sigma, \Pi}(\mathrm{Ciach1})$$

Natomiast inna możliwa postać reguł (LK) i (PA) jest taka:

$$\frac{\Gamma, \alpha, \beta \vdash \Sigma}{\Gamma, \alpha \wedge \beta \vdash \Sigma}\text{(LK1)} \qquad\qquad \frac{\Gamma \vdash \alpha, \beta, \Sigma}{\Gamma \vdash \alpha \vee \beta, \Sigma}\text{(PA1)}$$

Rachunek sekwentów Gentzena jest równoważny systemowi dowodzenia w stylu Hilberta.

Twierdzenie 16.3 *Następujące warunki są równoważne:*

1. *Sekwent* $\Gamma \vdash \beta_1, \ldots, \beta_m$ *ma dowód w sensie Gentzena;*
2. *Formuła* $\beta_1 \vee \cdots \vee \beta_m$ *ma dowód w sensie Hilberta ze zbioru założeń* Γ*;*
3. $\Gamma \models \beta_1 \vee \cdots \vee \beta_m$.

A zatem formuła α *jest tautologią wtedy i tylko wtedy, gdy sekwent „* $\vdash \alpha$*" ma dowód w sensie Gentzena.*

Dowód: (Szkic) Skorzystamy oczywiście z twierdzenia o pełności dla systemu Hilberta. Mamy stąd implikację (3) $\Rightarrow$ (2). Implikacja (1) $\Rightarrow$ (3) jest łatwa: wystarczy sprawdzić kolejno poprawność wszystkich reguł. Dowód implikacji (2) $\Rightarrow$ (1) też nie jest trudny, bo mamy do dyspozycji regułę cięcia. Na przykład mając $\Gamma \vdash \alpha$ i $\Gamma \vdash \alpha \to \beta$, można otrzymać $\Gamma \vdash \beta$ w następujący sposób:

$$\text{(Ciach)}\dfrac{\text{(PO)}\dfrac{\Gamma \vdash \alpha \to \beta}{\Gamma \vdash \alpha \to \beta, \beta} \qquad \text{(LI)}\dfrac{\Gamma \vdash \alpha \qquad \dfrac{\dfrac{\beta \vdash \beta}{\vdots}\text{(LO)}}{\Gamma, \beta \vdash \beta}\text{(LO)}}{\Gamma, \alpha \to \beta \vdash \beta}}{\Gamma \vdash \beta}$$

∎

W dowodzie powyższego twierdzenia istotnie korzystaliśmy z reguły cięcia. Okazuje się jednak, że cięcie nie jest niezbędne. I właśnie o to chodzi.

Twierdzenie 16.4 (o eliminacji cięcia) *Każdy sekwent, który ma dowód w rachunku sekwentów, ma też taki dowód, który nie używa reguły cięcia.*

Główna zaleta dowodów bez cięcia wynika z następującej *własności podformuł:* wszystkie formuły występujące w przesłance dowolnej reguły (poza cięciem) są podformułami formuł występujących w konkluzji. Dlatego np. w dowodzie sekwentu $\vdash \alpha$ mamy do czynienia tylko z podformułami formuły α. Dla danej formuły α, łatwiej więc znaleźć dowód w sensie

Gentzena niż dowód w sensie Hilberta. Dlatego systemy zbliżone do rachunku sekwentów są stosowane w praktyce.

Uwaga: Pojęcie *podformuły* danej formuły, w przypadku formuł rachunku zdań oznacza po prostu taką część danej formuły, która sama jest poprawnie zbudowaną formułą. Zatem każda formuła rachunku zdań ma tylko skończenie wiele podformuł. W przypadku formuł z kwantyfikatorami jest niestety inaczej: za podformułę formuły $\forall x \varphi$ musimy uważać każdą formułę postaci $\varphi[x{:=}t]$, gdzie t jest dowolnym termem. Formuła z kwantyfikatorami ma więc nieskończenie wiele podformuł.

Z własności podformuł wynika własność *konserwatywności* logiki nad swoimi fragmentami: jeśli formuła, w której nie występuje jakiś spójnik jest tautologią, to jej wyprowadzenie nie wymaga reguł związanych z tym spójnikiem.

17 Klauzule Horna i reguła rezolucji

Klauzulą nazywamy sekwent postaci $p_1, \ldots, p_n \vdash q_1, \ldots, q_m$, gdzie $p_1, \ldots, p_n, q_1, \ldots, q_m$ są zmiennymi zdaniowymi, i $m, n \geq 0$. *Klauzulą Horna* nazywamy klauzulę, dla której $m = 1$.

Przypuśćmy teraz, że mamy do czynienia tylko z klauzulami, ale za to oprócz standardowych aksjomatów

$$\bot \vdash \quad \text{oraz} \quad \alpha \vdash \alpha,$$

mamy jeszcze jakieś aksjomaty dodatkowe. Wtedy Twierdzenie 16.4 nie jest już prawdziwe, ale mamy jego następującą wersję:

Twierdzenie 17.1 (o rezolucji) *Każda klauzula, która ma dowód w systemie Gentzena, rozszerzonym o pewne klauzule jako aksjomaty, ma też taki dowód używający tylko reguł osłabiania i reguły cięcia, stosowanej zawsze tak, że lewa przesłanka jest aksjomatem:*

$$\frac{p_1, \ldots, p_n \vdash s_1, \ldots, s_\ell, q, \qquad q, q_1, \ldots, q_k \vdash r_1, \ldots, r_m}{p_1, \ldots, p_n, q_1, \ldots, q_k \vdash s_1, \ldots, s_\ell, r_1, \ldots, r_m}$$

Reguła cięcia, ograniczona jak w Twierdzeniu 17.1, nosi nazwę *reguły rezolucji*. Jej „tłumaczenie” na styl Hilberta może być zapisane tak:

$$\frac{\neg p_1 \vee \cdots \vee \neg p_n \vee s_1 \vee \cdots \vee s_\ell \vee q, \qquad \neg q \vee \neg q_1 \vee \cdots \vee \neg q_k \vee r_1 \vee \ldots \vee r_m}{\neg p_1 \vee \cdots \vee \neg p_n \vee \neg q_1 \vee \cdots \vee \neg q_k \vee s_1 \vee \cdots \vee s_\ell \vee r_1 \vee \cdots \vee r_m}$$

W uproszczeniu:

$$\frac{\mathrm{L} \vee q, \qquad \neg q \vee \mathrm{P}}{\mathrm{L} \vee \mathrm{P}}$$

Jeśli rozważane są tylko klauzule Horna to reguła rezolucji jest stosowana w postaci:

$$\frac{p_1, \ldots, p_n \vdash q, \quad q, q_1, \ldots, q_k \vdash r}{p_1, \ldots, p_n, q_1, \ldots, q_k \vdash r}$$

Zadanie, które chcemy rozwiązywać, jest następujące: przyjmując pewne klauzule Horna jako nowe aksjomaty, chcemy wywnioskować klauzulę postaci $\vdash p$. Przykładowo: wiemy, że drapieżniki (D) mają wąsy (W), i że ten kto ma wąsy i ogon (O) jest kotem (K). Mamy drapieżnika z ogonem i pytamy czy to jest kot.

Naszymi aksjomatami są więc cztery klauzule Horna:

- $W, O \vdash K$;
- $D \vdash W$;
- $\vdash D$;
- $\vdash O$.

Można z nich łatwo wyprowadzić klauzulę $\vdash K$. Jednak proces wyszukiwania potrzebnej informacji jest lepiej widoczny, gdy zamiast dowodzić $\vdash K$ dodamy do aksjomatów klauzulę „$K \vdash$ ” i spróbujemy wyprowadzić sprzeczność, tj klauzulę „ $\vdash$ ”. Można to zrobić tak:

$$\dfrac{\vdash O \qquad \dfrac{\vdash D \qquad \dfrac{D \vdash W \qquad \dfrac{W, O \vdash K \qquad K \vdash}{W, O \vdash}}{D, O \vdash}}{O \vdash}}{\vdash}$$

Zauważmy, że jeśli zamienimy w powyższym dowodzie klauzule postaci $\Gamma \vdash$ na $\Gamma \vdash K$ to otrzymamy dowód „wprost” klauzuli $\vdash K$.

Sekwent postaci „$p \vdash$ ” nazywamy *klauzulą celu*. Klauzulę celu wraz z danymi aksjomatami nazywamy *programem logicznym*. W naszym przykładzie mamy do czynienia z programem, który w często spotykanej notacji byłby zapisany jako pięć klauzul. Są to dwie „reguły”:

$$K :\!-\; W, O. \qquad \text{i} \qquad W :\!-\; D.$$

dwa „fakty”:

$$O :\!-\; . \qquad \text{i} \qquad D :\!-\; .$$

oraz cel:

$$:\!-\; K?$$

Wykonanie programu polega na sprowadzeniu pytania o K do pytania o W, O i ostatecznie do faktów O i D. Interpreter języka programowania udzieli wtedy odpowiedzi „tak”. Nie zawsze to musi być natychmiastowe: w przypadku programu

$$A :- B. \quad C :- A. \quad C :- D. \quad D :- . \quad :- C?$$

próba sprowadzenia pytania o C do pytania o A nie prowadzi do sukcesu i konieczny jest nawrót. W ogólności może być gorzej: interpreter może wpaść w pętlę i nie znaleźć dowodu choć on istnieje. Mówimy wtedy o niezgodności interpretacji „procedurowej” programu z jego interpretacją „logiczną”. Problem ten wystąpi na przykład dla programu:

$$A :- C. \quad C :- A. \quad C :- D. \quad D :- . \quad :- C?$$

Programowanie w logice

Programy logiczne w których występują tylko zmienne zdaniowe, mają umiarkowane zastosowania. (Ostatecznie, kot jaki jest, każdy widzi.) Należy więc metodę rezolucji rozszerzyć na język rachunku predykatów. Klauzule Horna to teraz sekwenty postaci $A_1, \ldots, A_n \vdash B$, gdzie A_i oraz B są formułami atomowymi, a klauzula celu ma postać $C(x_1, \ldots, x_k) \vdash$, gdzie $C \in \Sigma^R_k$. Wyprowadzenie sprzeczności z takiej klauzuli celu, to tyle co dowód formuły $\exists x_1, \ldots, x_k C(x_1, \ldots, x_k)$. W istocie dowód ten jest konstruktywny, tj. polega na znalezieniu konkretnych termów $t_1, \ldots, t_k$, dla których sekwent $\vdash C(t_1, \ldots, t_k)$ ma dowód.

Regułę rezolucji w postaci odpowiedniej dla logiki pierwszego rzędu można zapisać tak:

$$\frac{A_1, \ldots, A_n \vdash B \qquad C, C_1, \ldots, C_k \vdash D}{S(A_1), \ldots, S(A_n), S(C_1), \ldots, S(C_k) \vdash S(D)}$$

gdzie S jest takim podstawieniem, że $S(B) = S(C)$. A zatem reguła rezolucji łączy w sobie elementy reguły cięcia i operacji podstawienia.

Program logiczny, jak poprzednio, składa się z reguł i faktów, oraz klauzuli celu, np:

- **matka**(Zofia, Helena) :− .
- **matka**(Helena, Barbara) :− .
- **babka**(x, y) :− **matka**(x, z),**matka**(z, y).
- :− **babka**(x, Barbara)?

Ustalenie, że Zofia jest babką Barbary polega na zastosowaniu rezolucji z użyciem podstawienia $S(y)$ =Barbara; $S(z)$ = Helena; $S(z)$ = Zofia. Wynikiem działania programu jest stała „Zofia”.

Inny przykład:

- **silnia**$(0,1)$:− .
- **silnia**$(n+1,x)$:− **silnia**$(n,y), x = y * (n+1)$.
- :− **silnia**$(4,x)$?

Programy logiczne mogą być używane do rozwiązywania takich samych zadań jak programy imperatywne. Różnica polega na tym, że program logiczny nie określa w pełni czynności jakie mają być wykonane dla znalezienia wyniku, a raczej podaje specyfikację oczekiwanych związków pomiędzy wejściem i wyjściem, pozostawiając szczegóły interpreterowi.

Przykład 17.2 Zadanie polega na takim pokolorowaniu mapy, aby sąsiednie obszary miały różne kolory. Mapa jest taka:

```
+---------+---------+-----+
| A       |       B | C   |
|         |         |     |
|    +----+----+    |     |
|    | D       |    |     |
+----+         +----+     |
| E  |         |  F |     |
|    +----+----+    |     |
|         |         |     |
|         |         |     |
+---------+---------+-----+
```

Program w jednym z dialektów języka Prolog wygląda tak:

```
domains
   kolor=symbol
predicates
   mapa(kolor,kolor,kolor,kolor,kolor,kolor)
   ok(kolor,kolor)
   inne(kolor,kolor)
clauses
   mapa(A,B,C,D,E,F) :- ok(A,B),ok(A,D),ok(A,E),ok(B,C),ok(B,D),
                        ok(B,F),ok(C,F),ok(D,E),ok(D,F),ok(E,F).
   inne(czerwone,zielone).
   inne(czerwone,niebieskie).
```

```
   inne(zielone,niebieskie).
   ok(X,Y) :-  inne(X,Y).
   ok(X,Y) :- inne(Y,X).

Goal: mapa(A,B,C,D,E,F)
```

Rozwiązania znalezione przez program:

```
A=czerwone, B=zielone, C=niebieskie, D=niebieskie, E=zielone, F=czerwone
A=czerwone, B=niebieskie, C=zielone, D=zielone, E=niebieskie, F=czerwone
A=zielone, B=niebieskie, C=czerwone, D=czerwone, E=niebieskie, F=zielone
A=zielone, B=czerwone, C=niebieskie, D=niebieskie, E=czerwone, F=zielone
A=niebieskie, B=czerwone, C=zielone, D=zielone, E=czerwone, F=niebieskie
A=niebieskie, B=zielone, C=czerwone, D=czerwone, E=zielone, F=niebieskie
6 Solutions
```


Podziękowania

Za liczne uwagi, które pomogły usunąć z tych notatek rozmaite błędy, dziękuję Paniom Eli Krępskiej i Magdzie Michalskiej oraz Panom Michałowi Brzozowskiemu, Piotrowi Dittwaldowi, Markowi Dopierze, Aleksandrowi Jankowskiemu, Michałowi Jaszczykowi, Łukaszowi Kalbarczykowi, Adamowi Kawie, Pawłowi Kępce, Krzysztofowi Kulewskiemu, Olkowi Nałęczyńskiemu, Filipowi Noworycie, Krzysztofowi Nozderko, Mikołajowi Radwanowi, Sławomirowi Sadziakowi, Przemkowi Strzelczykowi, Bartoszowi Sułkowskiemu, Jankowi Urbańskiemu, Piotrowi Wojtalewiczowi, a także prof. Piotrowi Zakrzewskiemu.

Ostatnia zmiana 25 sierpnia 2005 o godzinie 12:18.