Uniwersytet Warszawski
Wydział Filozofii i Socjologii

Marek Czarnecki
Nr albumu: 208417

Definicje prawdy w ubogich FM–dziedzinach

Truth definitions in poor FM–domains

Praca magisterska
na kierunku Filozofia
w zakresie Logika

Praca wykonana pod kierunkiem
prof. dr hab. Marcina Mostowskiego
Instytut Filozofii, Uniwersytet Warszawski

Warszawa, czerwiec 2008

Oświadczenie kierującego pracą

Oświadczam, że niniejsza praca została przygotowana pod moim kierunkiem i stwierdzam, że spełnia ona warunki do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

Data　　　　　　　　　　　　　　　　　　　Podpis kierującego pracą

Oświadczenie autora pracy

Świadom odpowiedzialności prawnej oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie i nie zawiera treści uzyskanych w sposób niezgodny z obowiązującymi przepisami.

Oświadczam również, że przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem tytułu zawodowego w wyższej uczelni.

Oświadczam ponadto, że niniejsza wersja pracy jest identyczna z załączoną wersją elektroniczną.

Data　　　　　　　　　　　　　　　　　　　Podpis autora pracy

Streszczenie

W niniejszej pracy zajmujemy się rekonstrukcją semantyki w skończonych modelach arytmetyki względnej pierwszości. Arytmetyka względnej pierwszości jest najsłabszą znaną, spośród ubogich arytmetyk, w której można zinterpretować w modelach skończonych arytmetykę dodawania i mnożenia. W szczególności dowodzimy FM–wersji twierdzenia Tarskiego o niedefiniowalności prawdy dla tej arytmetyki. Praca zawiera aparat logiczny niezbędny do dowodu tego twierdzenia.

Abstract

In this work we reconstruct semantics in finite models of coprimality. This arithmetic is the weakest known poor arithmetic in which arithmetic of addition and multiplication can be interpreted in finite models. In particular we prove FM–version of Tarski's undefinability of truth theorem for arithmetic of coprimality. This work contains all necessary logic apparatus for obtaining this result.

Słowa kluczowe

definicja prawdy, model skończony, względna pierwszość, uboga arytmetyka, FM-reprezentowalność, interpretowalność, FM-dziedzina, twierdzenie Tarskiego o niedefiniowalności prawdy

Dziedzina pracy

(kody według programu Socrates Erasmus)

08100

Spis treści

1 Wstęp

1.1 Motywacje

Rozważania nad modelami skończonymi wpisują się w szerszy kontekst filozoficzny badań dotyczących posiadanych przez nas środków poznawczych, jeżeli świat, w którym żyjemy, jest skończony, a dokładniej — jedynie potencjalnie nieskończony. Wydaje się prawdopodobnym, że nasz świat nie jest nieskończony w sensie aktualnym — zatem jest skończony, jednak w każdym momencie możemy rozszerzyć go o nowe elementy na przykład koncepcje, konstrukcje. Jan Mycielski w swojej pracy *Analysis without actual infinity* [15] pokazał jak można podstawowe pojęcia analizy matematycznej zrekonstruować na początkowych segmentach liczb naturalnych. W ten sposób pojęcia takie jak granica ciągu, ciągłość, czy pochodna funkcji uzyskują naturalną interpretację bez aktualnej nieskończoności. Marcin Mostowski w swojej pracy *On representing concepts in finite models* [9] pokazał, w jaki sposób można na takich segmentach początkowych reprezentować semantykę. Wykorzystał on też w miarę bogaty zestaw pojęć arytmetycznych, z których — jak łatwo zauważyć — wystarczające są już dodawanie i mnożenie. W późniejszych badaniach nad arytmetykami skończonymi wynika, że wystarczają: samo mnożenie (Krynicki, Zdanowski [6]), ewentualnie sama relacja podzielności (Mostowski, Wasilewska [12]), lub nawet sama względna pierwszość (Mostowski, Zdanowski [14]). Jak zauważają autorzy tej ostatniej pracy, z podanych przez nich rezultatów wynika, że pewne elementy semantyki można wyrazić w skończonych modelach w języku względnej pierwszości. Celem niniejszej pracy jest rozwinięcie tej idei oraz pokazanie FM–wersji twierdzenia Tarskiego o niedefiniowalności prawdy dla arytmetyki względnej pierwszości. Chcielibyśmy pokazać, że w skończonych modelach, już arytmetyka wyposażona jedynie w predykat $\perp$ — względnej pierwszości wystarcza, by zdefiniować prawdę arytmetyczną. W tym celu zbadamy jakie własności formuł reprezentować można w modelach skończonych, w tym celu posłużymy się wprowadzonym przez Marcina Mostowskiego w [13] pojęciem FM–reprezentowalności. Podamy dokładną charakterystykę relacji FM–reprezentowalnych w skończonych modelach arytmetyki dodawania i mnożenia w postaci Twierdzenia o FM–reprezentowalności, a następnie sprawdzimy jakie wypływają z niego konsekwencje dla FM–reprezentowalności relacji w modelach względnej pierwszości.

W badaniach nad własnościami skończonych modeli nie zauważano dotychczas (pierwszy raz w [6]) wartości informacji, jaką uzyskujemy wiedząc, że niektóre wyniki pewnych działań nie mieszczą się już w konkretnym modelu skończonym — na przykład w modelu $(\{0, ..., 5\}, +, \times)$ prawdziwe jest następujące zdanie: $\neg\exists c\, 2 \times 3 = c$. Zauważmy, że w modelach skończonych formuły postaci $a \times b = c$ traktujemy jako trójargumentowe predykaty z argumentami a, b, c. Jak dotąd nieistnienie elementów modelu spełniających rozmaite relacje traktowano jako przeszkodę i usuwano uznając na przykład, że interesują nas jedynie większe modele, takie w których istnieje wynik działania 2×3. Marcin Mostowski zwrócił uwagę, że to co postrzegane było jako mało interesująca i raczej dokuczliwa cecha modeli skończonych, w istocie w znacznym stopniu zwiększa siłę semantyczną języka pierwszego rzędu w tych modelach. Fakt ten wykorzystany został po raz pierwszy w pracach [6, 12]. W standardowym — nieskończonym modelu arytmetyki funkcje takie jak na przykład dodawanie, czy mnożenie są funkcjami całkowitymi, w przeciwieństwie do dodawania i mnożenia obciętego do pewnego odcinka początkowego liczb naturalnych. Ponieważ w skończonych modelach są to funkcje częściowe, w naturalny sposób będziemy traktować je jako trójargumentowe relacje. Przy czym odpowiednie predykaty z argumentami a, b, c zapiszemy jako $a + b = c$ oraz $a \times b = c$.

W naszej pracy pokażemy, w jaki sposób można wykorzystać fakt, że niektóre funkcje całkowite w modelu standardowym, są częściowe w modelach skończonych. Pokażemy, że względna pierwszość jest bardzo słabą relacją arytmetyczną w modelu standardowym oraz jak bardzo zmienia się jej siła semantyczna, gdy ograniczymy ją do modeli skończonych.

1.2 Ustalenia notacyjne

W niniejszej pracy używamy standardowych oznaczeń i terminów dla opisu pojęć arytmetycznie definiowalnych. Ponieważ jednak w różnych zakresach badawczych funkcjonują pewne rozbieżności notacyjne, ustalamy więc naszą terminologię następująco.

Δ_0 (również Σ_0 i Π_0) są to formuły arytmetyki dodawania i mnożenia z kwantyfikatorami ograniczonymi. Formuły Σ_n są następującej postaci: $\exists x_1 \forall x_2 ... Q x_n \, \varphi(x_1, ..., x_n, y_1, ..., y_k)$, gdzie Q jest $\forall$ dla n parzystych, a $\exists$ dla n nieparzystych oraz $\varphi(x_1, ..., x_n, y_1, ..., y_k)$ jest formułą Δ_0. Podobnie definiujemy klasę Π_n, z tą różnicą, że pierwszy kwantyfikator jest ogólny.

Na mocy twierdzenia o preneksowej postaci normalnej oraz możliwości wyrażania rzutowań dla par uporządkowanych za pomocą formuł Δ_0 wiemy, że każda formuła arytmetyczna równoważna jest w modelu standardowym pewnej formule Σ_n.

Podobnych oznaczeń używamy dla klasyfikowania relacji arytmetycznie definiowalnych (definiowalnych za pomocą formuł pierwszego rzędu z dodawaniem i mnożeniem). Mówimy, że relacja jest Σ_n (Π_n), dla $n \geq 1$, jeżeli jest definiowalna w modelu standardowym przez pewną formułę Σ_n (Π_n). Dla $n \geq 1$ relacja jest Δ_n dokładnie wtedy, gdy jest zarazem Σ_n oraz Π_n. W szczególności wiadomo, że Σ_1 są to dokładnie relacje rekurencyjnie przeliczalne. Relacje Δ_1 są to dokładnie relacje rekurencyjne. Relacje Δ_2 określa się też jako relacje rekurencyjne w granicy[1]. W szczególności wiadomo, że zbiór relacji Δ_1 zawarty jest w zbiorze relacji Σ_1 (każda relacja rekurencyjna jest rekurencyjnie przeliczalna), wszystkie relacje Σ_1 lub Π_1 są również Δ_2.

1.3 Twierdzenie Tarskiego o niedefiniowalności prawdy

W 1933 roku Alfred Tarski w swojej pracy *Pojęcie prawdy w językach nauk dedukcyjnych* [19] pokazał, jak można w naukowy sposób uprawiać semantykę. Był to wielki przełom w sposobie patrzenia na język nauki, zdominowany wówczas propagowanym przez Rudolfa Carnapa podejściem, polegającym na badaniu wyłącznie składni języka. Wynikało to w dużej mierze z powszechnego przekonania, że badania nad semantyką w istotny sposób wykraczają poza nasze możliwości poznawcze, i że jedynym aspektem języka dającym się sensownie opisywać jest składnia, ponieważ podczas rozważań na poziomie semantycznym wikłamy się w antynomie — takie jak na przykład słynne antynomialne zdanie Eubulidesa: *Ja teraz kłamię*. Praca Tarskiego doprowadziła do rewizji sposobu patrzenia na badania nad językiem, kładąc fundament dla badań z dziedziny semantyki, których pierwszym ważnym wynikiem jest właśnie twierdzenie Tarskiego o niedefiniowalności prawdy.

[1] Przegląd interesujących metodologicznie, równoważnych charakterystyk relacji Δ_2 można znaleźć w [11].

W 1931 roku Kurt Gödel podał przykład prawdziwego zdania arytmetycznego, które nie może być dowiedzione na gruncie systemu z Principia Mathematica, oraz którego negacja również jest w nim niedowodliwa [3]. Oczywiście twierdzenie o niezupełności nie odnosi się jedynie do tego jednego systemu formalnego. Analizując klasyczny dowód twierdzenia o niezupełności arytmetyki można wyabstrahować warunki wystarczające, by móc przeprowadzić ten dowód. Jednym z ważniejszych lematów, z których Gödel korzysta jest — wyodrębniony po raz pierwszy przez Carnapa [2] — lemat przekątniowy, pozwalający konstruować zdania samoodnoszące, dla dowolnej formuły z co najmniej jedną zmienną wolną. To właśnie możliwość wykorzystania samoodniesienia jest jednym z kluczowych składników potrzebnych do dowodu obu omawianych tutaj twierdzeń. Gödel pokazał, że w standardowej arytmetyce można udowodnić lemat przekątniowy. Podamy teraz pewną wersję tego lematu.

Twierdzenie 1. *(Lemat przekątniowy) Niech $\mathcal{K}$ będzie wystarczająco silną klasą formuł arytmetycznych oraz niech $\varphi(x) \in \mathcal{K}$ będzie formułą z jedną zmienną wolną x. Wtedy istnieje zdanie $\psi \in \mathcal{K}$ takie, że $\mathcal{N} \models \psi \equiv \varphi(\ulcorner \psi \urcorner)$.*

Powyższe twierdzenie niewiele mówi, dopóki nie wyjaśnimy co znaczy *wystarczająco silna klasa formuł arytmetycznych*, oraz w jaki sposób można nazywać zdania wewnątrz języka (klasy formuł), co oznaczaliśmy powyżej za pomocą symboli $\ulcorner\,\urcorner$. Gödel wymyślił sposób w jaki można zarytmetyzować język, czyli kodować (nazywać) formuły za pomocą liczb oraz pokazał, że gdy możemy skonstruować w języku formuły $\mathrm{Name}(x)$ oraz $\mathrm{Subst}(x, y)$ takie, że $\mathrm{Name}(x) = y$ wyraża fakt, że y jest nazwą liczby x oraz $\mathrm{Subst}(x, y) = z$ gdy z jest nazwą formuły powstającej z formuły o nazwie x przez podstawienie w miejsce pierwszej zmiennej wolnej termu o nazwie y, wtedy też możemy udowodnić lemat przekątniowy. Idea zawarta w dowodzie polega właśnie na próbie rekonstrukcji samoodnoszenia wyrażeń języka — tak jak w przypadku, gdy rozważamy zdanie: *Ja teraz kłamię* — wypowiadając to zdanie twierdzimy coś o naszej aktualnej wypowiedzi.

Drugą istotną własnością języka wykorzystywaną w dowodzie twierdzenia Tarskiego jest domkniętość na negację. Zauważmy, że łatwo możemy skonstruować zdania samoodnoszące, które nie są antynomialne, jak na przykład: *To zdanie składa się z siedmiu słów.* Zdanie to jest po prostu prawdziwe, a jego wartość logiczna nie budzi naszych wątpliwości, mimo iż odnosi się ono samo do siebie. Konieczność domknięcia na negację rozważanego języka jest istotna — antynomia Eubuidesa opiera się na zdaniu, które orzeka o sobie, że **nie** jest prawdzie, zdanie Gödla orzeka o sobie, że **nie** jest dowodliwe w danym systemie formalnym. Przedstawimy teraz twierdzenie Tarskiego o niedefiniowalności prawdy.

Twierdzenie 2. *(Tarskiego o niedefiniowalności prawdy) Dla dowolnej klasy formuł arytmetycznych $\mathcal{K}$ zawierającej Δ_0 oraz domkniętej na konstrukcje pierwszego rzędu — w tym na negację — nie istnieje formuła $\varphi(x) \in \mathcal{K}$ z jedną zmienną wolną x taka, że dla dowolnego zdania $\psi \in \mathcal{K}$ zachodzi $\mathcal{N} \models \psi \equiv \varphi(\ulcorner \psi \urcorner)$.*

Zauważmy, że to co nazywaliśmy wystarczającą siłą wyrazu teorii zostało wyeksplikowane jako $\Delta_0 \subseteq \mathcal{K}$ — faktycznie zarówno formułę $\mathrm{Name}(x)$ jak i $\mathrm{Subst}(x, y)$ możemy zdefiniować jako Δ_0–formuły. Uwzględniona została też konieczność domknięcia na negację klasy $\mathcal{K}$ — zauważmy, że klasy formuł Σ_n i Π_n zawierają Δ_0 dla dowolnego $n \geq 1$, jednak mieszczą w sobie własne definicje prawdy — nie są zatem domknięte na negację [18].

Twierdzenie Tarskiego dostarcza nam również bardzo użytecznego narzędzia do klasyfikacji języków pod względem ich siły semantycznej. Pokazuje ono, że gdy staramy się określić semantykę, dla pewnego języka bazowego $\mathcal{L}_0$ musimy posługiwać się pewnym silniejszym językiem $\mathcal{L}_1$, oraz że $\mathcal{L}_0$ nie może wystarczać do tego celu. Z drugiej strony, gdy rozważając siłę semantyczną dwóch różnych języków $\mathcal{L}_0$ i $\mathcal{L}_1$ skonstruujemy w $\mathcal{L}_1$ definicję prawdy dla zdań z $\mathcal{L}_0$, natychmiastowo możemy wywnioskować, że siła semantyczna $\mathcal{L}_1$ musi znacząco przewyższać siłę semantyczną $\mathcal{L}_0$.

1.4 Względna pierwszość w modelu standardowym

Pokażemy teraz, jak słabą relacją jest względna pierwszości w standardowym modelu. Łatwo zauważyć, że relacja względnej pierwszości $\perp$ tworzy na liczbach naturalnych dość prostą strukturę algebraiczną — atomową kratę z elementem najmniejszym 1 i elementem największym 0. Chcielibyśmy jednak spojrzeć na arytmetykę względnej pierwszości w trochę szerszym kontekście, rozważając ją jako jedną z ubogich arytmetyk: $\mathrm{Th}((\omega, \perp))$, $\mathrm{Th}((\omega, |))$ i $\mathrm{Th}((\omega, \times))$ odpowiednio: arytmetyka względnej pierwszości, arytmetyka podzielności i arytmetyka mnożenia (Skolema).

Za pomocą mnożenia, bez trudu możemy zdefiniować w logice pierwszego rzędu podzielność, natomiast przy pomocy podzielności możemy wyrazić względną pierwszość. Mamy następujące definicje:

- $x|y \equiv_{df} \exists z\, x \times z = y$
- $x \perp y \equiv_{df} \forall z\, ((z|x \wedge z|y) \Rightarrow \forall w\, z|w)$

Zatem siła semantyczna arytmetyki względnej pierwszości musi być mniejsza niż arytmetyki podzielności, najsilniejsza zaś jest arytmetyka mnożenia.

W 1930 roku Thoralf Skolem ogłosił, że teoria $\mathrm{Th}((\omega, \times))$ jest niesprzeczna, zupełna i rozstrzygalna [17], czyli że istnieje algorytm, który otrzymując na wejściu kod zdania φ w języku arytmetyki Skolema odpowiada, czy $\varphi \in \mathrm{Th}((\omega, \times))$, czy $\neg\varphi \in \mathrm{Th}((\omega, \times))$. W artykule Skolema nie znajdował się jednak bezpośredni dowód tego faktu — dopiero Andrzej Mostowski w 1952 roku podał jego pełny dowód [8]. Polegał on na obserwacji, w jaki sposób możemy zinterpretować każdą z rozważanych tu arytmetyk.

Nie zaprezentujemy teraz pełnego dowodu rozstrzygalności arytmetyki Skolema, przedstawimy jedynie szkic oryginalnego rozumowania z [8] natomiast w Dodatku I. podamy inny, ciekawy dowód rozstrzygalności arytmetyki Skolema, korzystający z pojęć automatu skończonego i języków regularnych oraz podstawowych faktów ich dotyczących. Na mocy zasadniczego twierdzenia arytmetyki każda liczba naturalna większa od 1 posiada jednoznaczny rozkład na czynniki pierwsze: $n = p_0^{s_0}...p_k^{s_k}$. Możemy zatem zinterpretować arytmetykę mnożenia w strukturze algebraicznej będącej przeliczalnym produktem prostym $(\omega, +) \times (\omega, +) \times ...$ i z jej pomocą badać $(\omega, \times)$. Produkt prosty różni się od zwyczajnego iloczynu kartezjańskiego tym, że zawiera jedynie takie elementy tego iloczynu, w których tylko skończenie wiele wyrazów różni się od zera. Gdy $n = p_0^{s_0}...p_k^{s_k}$, $m = p_0^{t_0}...p_l^{t_l}$ (dla prostoty załóżmy, że $k < l$), to $n \times m = p_0^{s_0+t_0}...p_k^{s_k+t_k}p_{k+1}^{t_{k+1}}...p_l^{t_l}$. W taki sposób sprowadzamy mnożenie do prostszego działania — dodawania wykładników odpowiednich liczb pierwszych.

Zwróćmy uwagę jeszcze na podobny sposób interpretacji przez produkty proste arytmetyki podzielności i arytmetyki względnej pierwszości. Tą pierwszą można interpretować jako przeliczalny produkt prosty $(\omega, <) \times (\omega, <) \times ...$ ponieważ podzielność sprowadza się do porównywania wielkości wykładników dwóch liczb w ich rozkładach na czynniki pierwsze. Rozpatrywana przez nas relacja względnej pierwszości w modelu standardowym interpretuje się przez trywialną wręcz relację — identyczność $(\omega, =) \times (\omega, =) \times ...$ ponieważ na to by dwie liczby były względnie pierwsze potrzeba i wystarcza, by nie miały wspólnych dzielników pierwszych. Powyższy sposób rozważania słabych arytmetyk dostarcza nam również intuicji dotyczących ich siły semantycznej — pokazuje też, że już identyczność wystarcza w pewnym sensie do mówienia o względnej pierwszości.

Oczywiście z rozstrzygalności arytmetyki Skolema wynika rozstrzygalność arytmetyki względnej pierwszości. Jak się wkrótce przekonamy w skończonych modelach siła względnej pierwszości rośnie na tyle, by pozwolić na zinterpretowanie pełnej arytmetyki.

2 FM–definicje prawdy

Zajmiemy się teraz definicjami prawdy w modelach skończonych.

Definicja 1. *Niech $\mathcal{K}$ będzie pewną klasą formuł ustalonego słownika σ. Powiemy, że σ–formuła $\varphi(x)$ jest FM–definicją prawdy dla $\mathcal{K}$, gdy dla dowolnego zdania $\psi \in \mathcal{K}$ i dla prawie wszystkich (wszystkich poza skończoną liczbą) skończonych σ–modeli M zachodzi:*

$$M \models \psi \equiv \varphi(\ulcorner \psi \urcorner)$$

Niech $R \subseteq \omega^r$ będzie relacją arytmetyczną, wtedy $R^{(n)} = R \cap \{0, ..., n-1\}^n$.
W naszych rozważaniach ograniczamy się do skończonych modeli (o słownikach relacyjnych), których uniwersa są segmentami początkowymi liczb naturalnych. Dla dowolnego modelu $\mathcal{A}$ ustalonego słownika relacyjnego $\sigma = \{R_1, ..., R_k\}$ definiujemy FM–dziedzinę $\mathrm{FM}(\mathcal{A}) = \{\mathcal{A}_n : n = 0, 1, 2, ...\}$, gdzie $\mathcal{A}_n = (\{0, ..., n-1\}, R_1^{(n)}, ..., R_k^{(n)})$. Możemy zatem zamiast w prawie wszystkich modelach równoważnie powiedzieć — w dostatecznie dużych modelach. Prowadzi to nas do wprowadzenia następującego oznaczenia:
$\mathrm{FM}(\mathcal{A}) \models_{sl} \psi$ wtedy i tylko wtedy, gdy $\exists k \forall n > k\, \mathcal{A}_n \models \psi$.

Omawiając twierdzenie Tarskiego o niedefiniowalności prawdy zwracaliśmy uwagę na konieczność posiadania nazw wyrażeń rozpatrywanego języka. W naszym przypadku chcemy przypisać nazwy formułom arytmetycznym. Niech A będzie zbiorem wszystkich symboli, z których powstają formuły arytmetyczne pierwszego rzędu, czyli $A = \{=, 0, s, +, \times, \perp, \neg \rightarrow, \forall, (,)\} \cup \{v_i : i = 0, 1, 2, ...\}$. Kwantyfikator szczegółowy oraz pozostałe spójniki logiczne są definiowalny za pomocą negacji, implikacji i kwantyfikatora ogólnego, natomiast zbiór $\{v_i : i = 0, 1, 2, ...\}$ to przeliczalny zbiór zmiennych indywiduowych. Mimo iż w słowniku pełnej arytmetyki mamy jedynie dwie trójargumentowe relacje $+$ i $\times$ oraz dwuargumentową identyczność $=$, chcemy rozpatrywać formuły jako ciągi nad większym alfabetem —

zbiorem A zdefiniowanym powyżej. Nie jest to jednak istotne rozszerzenie języka, ponieważ wszystkie dodatkowe symbole, które wprowadziliśmy: 0, s i $\perp$ odpowiadają definiowalnym za pomocą $+$ i $\times$ relacjom.
Wprowadzimy teraz następujące kodowanie składni, za pomocą funkcji arytmetycznej $\ulcorner\ \urcorner : A^* \to \omega$ takiej, że:

- $\ulcorner = \urcorner = 3$
- $\ulcorner 0 \urcorner = 5$
- $\ulcorner s \urcorner = 7$
- $\ulcorner + \urcorner = 9$
- $\ulcorner \times \urcorner = 11$
- $\ulcorner \perp \urcorner = 13$
- $\ulcorner \neg \urcorner = 15$
- $\ulcorner \rightarrow \urcorner = 17$
- $\ulcorner \forall \urcorner = 19$
- $\ulcorner (\urcorner = 21$
- $\ulcorner) \urcorner = 21$
- $\ulcorner v_i \urcorner = 2(i+1)$ dla $i = 0, 1, 2, ...$
- $\ulcorner \alpha \urcorner = p_0^{\ulcorner a_0 \urcorner + 1} ... p_n^{\ulcorner a_n \urcorner + 1}$, dla dowolnego słowa $\alpha = a_0, ..., a_n$ nad alfabetem A.

Dla dowolnej formuły arytmetycznej φ mówimy, że $\ulcorner \varphi \urcorner$ jest numerem (kodem) Gödla formuły φ. Zauważmy, że tak dobrane kodowanie jest różnowartościowe, co gwarantuje nam, że odczytując z kodu formułę otrzymujemy ją w sposób jednoznaczny. Różnowartościowość $\ulcorner\ \urcorner$ jest konsekwencją zasadniczego twierdzenia arytmetyki.

Przez $\mathcal{N}$ oznaczamy standardowy model arytmetyki:
$\mathcal{N} = (\omega, +, \times, s, \perp, 0)$. Chcielibyśmy mieć funkcje $\mathrm{Name}(x)$ i $\mathrm{Subst}(x, z)$, takie że:

- $\mathrm{FM}(\mathcal{N}) \models_{sl} \mathrm{Name}(x) = \ulcorner \underbrace{s(s...s}_{x}(0)) \urcorner$ dla dowolnego $x \in \omega$,
- $\mathrm{FM}(\mathcal{N}) \models_{sl} \mathrm{Subst}(\ulcorner \varphi(x) \urcorner, \ulcorner t \urcorner) = \ulcorner \varphi(t) \urcorner$ dla dowolnej formuły $\varphi(x)$ i termu t.

Mając funkcje $\mathrm{Name}(x)$ i $\mathrm{Subst}(x, z)$ możemy udowodnić FM–wersję lematu przekątniowego oraz FM–wersję twierdzenia Tarskiego dla $\mathrm{FM}(\mathcal{N})$.

Twierdzenie 3. *Dla dowolnej formuły arytmetycznej $\varphi(x)$ z jedną zmienną wolną x istnieje zdanie w języku arytmetyki zdanie ψ takie, że:*
$\mathrm{FM}(\mathcal{N}) \models_{sl} \psi \equiv \varphi(\ulcorner \psi \urcorner)$.

Powyższą FM–wersję lematu przekątniowego dla $\mathrm{FM}(\mathcal{N})$ pozostawiamy bez dowodu. W dalszej części pracy można będzie znaleźć analogiczny dowód dla $\mathrm{FM}((\omega, \perp))$. Przy pomocy lematu przekątniowego dowodzimy FM–wersji twierdzenia Tarskiego dla $\mathrm{FM}(\mathcal{N})$.

Twierdzenie 4. *Nie istnieje formuła arytmetyczna $\varphi(x)$ z jedną zmienną wolną x taka, że dla dowolnego zdania języka arytmetyki ψ zachodzi:*
$\mathrm{FM}(\mathcal{N}) \models_{sl} \psi \equiv \varphi(\ulcorner \psi \urcorner)$.

Dowód. Załóżmy, że istnieje formuła $\varphi(x)$ o takich własnościach. Stosując lemat przekątniowy do $\neg\varphi(x)$ otrzymujemy zdanie ψ takie, że:
$\mathrm{FM}(\mathcal{N}) \models_{sl} \psi \equiv \neg\varphi(\ulcorner \psi \urcorner)$.
Jednak ponieważ $\varphi(x)$ jest FM–definicją prawdy dla zdań arytmetycznych, więc zachodzi:
$\mathrm{FM}(\mathcal{N}) \models_{sl} \psi \equiv \varphi(\ulcorner \psi \urcorner)$.
Zatem dochodzimy do paradoksalnego wniosku, że:
$\mathrm{FM}(\mathcal{N}) \models_{sl} \psi \equiv \neg\psi$. □

Pokazaliśmy zatem, że o ile możemy reprezentować funkcje $\mathrm{Name}(x)$ i $\mathrm{Subst}(x, z)$ w modelach skończonych — podobnie jak w modelu standardowym — arytmetyka dodawania i mnożenia jest wystarczająco silna, by reprezentować semantykę.

3 Wyrażalność w $\mathrm{FM}((\omega, \perp))$

Termin FM–reprezentowalność wprowadzony został przez Marcina Mostowskiego w [9]. Główną motywacją dla badania jakie relacje są FM–reprezentowalne były rozważania nad tym jakie relacje można opisywać w skończonych modelach.

Wprowadzimy teraz pojęcie FM–reprezentowalności.

Definicja 2. *Niech $R \subseteq \omega^k$ będzie relacją arytmetyczną. Powiemy, że R jest FM–reprezentowana przez formułę $\varphi(x_1, ..., x_k)$ w FM–dziedzinie* $\mathrm{FM}(\mathcal{A})$ *wtedy i tylko wtedy, gdy dla dowolnych $a_1, ..., a_k \in \omega$ spełnione są następujące warunki:*

1. $\mathrm{FM}(\mathcal{A}) \models_{sl} \varphi[a_1, ..., a_k]$ *wtedy i tylko wtedy, gdy $R(a_1, ..., a_k)$,*

2. $\mathrm{FM}(\mathcal{A}) \models_{sl} \neg\varphi[a_1, ..., a_k]$ *wtedy i tylko wtedy, gdy $\neg R(a_1, ..., a_k)$.*

Powiemy, że R jest FM–reprezentowalna w $\mathrm{FM}(\mathcal{A})$*, jeżeli istnieje formuła φ, która FM–reprezentuje R w* $\mathrm{FM}(\mathcal{A})$*.*

W szczególności, aby móc udowodnić wersję twierdzenia Tarskiego dla pewnej FM–dziedziny $\mathrm{FM}(\mathcal{A})$, musimy dowieść FM–reprezentowalności relacji $\mathrm{Name}(x) = y$ oraz $\mathrm{Subst}(x, z) = w$ w $\mathrm{FM}(\mathcal{A})$.

3.1 FM–reprezentowalność w $\mathrm{FM}(\mathcal{N})$

Zaprezentujemy teraz wynik z pracy [9] charakteryzujący formuły FM–reprezentowalne w $\mathrm{FM}(\mathcal{N})$. Jego dowód można znaleźć w Dodatku II.

Twierdzenie 5. *(Twierdzenie o FM–reprezentowalności dla* $\mathrm{FM}(\mathcal{N})$*) Niech R będzie relacją arytmetyczną. R jest FM–reprezentowalna wtedy i tylko wtedy, gdy $R \in \Delta_2$.*

Δ_2 to bardzo duża klasa relacji arytmetycznych zawierająca wszystkie relacje rekurencyjne. Aby przekonać się, że FM–wersja twierdzenia Tarskiego może być udowodniona dla $\mathrm{FM}(\mathcal{N})$, wystarczy teraz pokazać, że funkcje $\mathrm{Name}(x)$ oraz $\mathrm{Subst}(x, z)$ są FM–reprezentowalne w $\mathrm{FM}(\mathcal{N})$. Poniżej podamy algorytmy obliczające te funkcje, dowodząc tym samym, że są one rekurencyjne.

3.1.1 Obliczanie $\mathrm{Name}(x)$

Powyższy algorytm działa w następujący sposób:
Na wejściu otrzymuje liczbę naturalną x.
Sprawdza, *czy* $x = 0$?

Wejście: Liczba naturalna x

Wyjście: Numer Gödla termu $\underbrace{s(s(...s(}_{x}0\underbrace{))...)}_{x}$

if $x = 0$ **then**
    `return`($\ulcorner 0 \urcorner$)
else
    $k := 1$;
    for $i := 1$ **to** x **do**
        $k := k \times p_{2i-2}^{\ulcorner s \urcorner + 1} \times p_{2i-1}^{\ulcorner (\urcorner + 1}$;
    end
    $k := k \times p_{2x}^{\ulcorner 0 \urcorner + 1}$;
    for $i := 1$ **to** x **do**
        $k := k \times p_{2x+i}^{\ulcorner) \urcorner + 1}$;
    end
    `return`(k);
end

Algorytm 1: Obliczanie funkcji Name(x)

- Jeżeli tak jest to kończy pracę zwracając wynik $\ulcorner 0 \urcorner$,
- W przeciwnym przypadku:
  - Ustawia wartość pomocniczą k na 1,
  - Liczy $\ulcorner \underbrace{s(...s(}_{x} \urcorner$ i zapisuje tę wartość w k,
  - Liczy $\ulcorner \underbrace{s(...s(}_{x} 0 \urcorner$ i zapisuje tę wartość w k,
  - Liczy $\ulcorner \underbrace{s(...s(}_{x} 0 \underbrace{)...)}_{x} \urcorner$, zapisuje tę wartość w k i zwraca k jako wynik.

Otrzymujemy zatem poprawny wynik Name(x) $= \ulcorner \underbrace{s(...s(}_{x} 0 \underbrace{)...)}_{x} \urcorner$ dla dowolnego $x \in \omega$.

3.1.2 Obliczanie Subst(x, z)

Wynik obliczenia, opisywanego w tym punkcie, algorytmu, na wejściu (x, z), jeżeli x jest kodem formuły, a z kodem termu, to kod formuły powstałej z formuły o nazwie x przez podstawienie w miejsce wszystkich wolnych wystąpień zmiennej v_0 termu o nazwie z, lub 0 w przeciwnym przypadku.

```
Wejście: Liczby naturalne x, z
Wyjście: Liczba naturalna y
if x jest kodem formuły, a z kodem termu then
    Buduj drzewo syntaktyczne formuły o nazwie x, korzeń drzewa
    nazwij x0;
    Buduj drzewo syntaktyczne termu o nazwie z, korzeń drzewa
    nazwij z0;
    Przeszukuj drzewo o korzeniu w wierzchołku x0;
    if Bieżący wierzchołek nie jest liściem then
        if Bieżący wierzchołek ma etykietę ∀v0 lub ∃v0 then
            Nie przeszukuj potomków bieżącego wierzchołka
        else
            Oznacz bieżący wierzchołek jako sprawdzony i przejdź
            do jego potomków
        end
    else
        if Bieżący wierzchołek ma etykietę v0 then
            Zastąp bieżący wierzchołek wierzchołkiem z0;
        end
    end
    Konstruuj formułę z powstałego drzewa syntaktycznego;
    k := numer Gödla otrzymanej formuły;
    return (k);
else
    return (0)
end
```

Algorytm 2: Obliczanie funkcji Subst(x, z)

Wiadomo, że z kodu formuły możemy odczytać wszystkie podformuły formuły φ. Algorytm ma tworzyć drzewo syntaktyczne zadanej formuły oraz drzewo syntaktyczne zadanego termu, o ile na wejściu otrzymał poprawne kody — w przeciwnym wypadku zwraca wartość 0. Następnie algorytm przeszukuje drzewo formuły dowolną metodą w następujący sposób:

- Jeżeli algorytm sprawdza wierzchołek w i w nie jest liściem, to:
  - Jeżeli etykieta wierzchołka w to $\forall v_0$ lub $\exists v_0$, to algorytm nie przeszukuje potomków w, ponieważ wszystkie tego typu wystąpienia v_0 są związane przez kwantyfikator,
  - W przeciwnym wypadku algorytm przeszukuje wszystkich potomków w,
- Jeżeli w jest liściem, to:
  - Jeżeli etykieta w to v_0, to algorytm w wstawia w miejsce w korzeń drzewa termu t, co odpowiada wstawieniu w zadanej formule w miejsce pewnego wystąpienia zmiennej wolnej v_0 zadanego termu,
  - W przeciwnym przypadku algorytm nie robi nic.

Następnie algorytm musi skonstruować formułę z powstałego drzewa syntaktycznego i jako wynik zwrócić numer Gödla otrzymanej formuły.

Wiemy zatem, istnieją algorytmy obliczające wartości funkcji $\mathrm{Name}(x)$ oraz $\mathrm{Subst}(x, z)$ na dowolnych argumentach, zatem są to funkcje rekurencyjne, więc tym bardziej są FM–reprezentowalne w $\mathrm{FM}(\mathcal{N})$. Dowiedliśmy tym samym, że można udowodnić FM–wersję lematu przekątniowego dla $\mathrm{FM}(\mathcal{N})$, a tym samym uzupełniliśmy ostatni brakujący fragment dowodu Twierdzenia 4.

3.2 FM–reprezentowalność w $\mathrm{FM}((\omega, \perp))$

Chcielibyśmy móc FM–reprezentować funkcje $\mathrm{Name}(x)$ oraz $\mathrm{Subst}(x, z)$ w $\mathrm{FM}((\omega, \perp))$. Niestety nie da się bezpośrednio przełożyć idei stosowanych w FM–dziedzinie pełnej arytmetyki. Problemem jest istnienie nietrywialnych automorfizmów modeli względnej pierwszości, czego konsekwencją jest niedefiniowalność wielu konceptów, których definicje mieliśmy w pełnej arytmetyce, wynika to z faktu, że relacja względnej pierwszości jest bardzo słaba.

Przy pomocy jedynie względnej pierwszości nie możemy rozróżniać elementów o tych samych nośnikach $\mathrm{Supp}(a) = \{p : p|a \text{ p–liczba pierwsza}\}$. Nie można zatem FM–reprezentować na przykład zbioru $\{2\}$, ponieważ jego elementy (2) spełniają dokładnie te same formuły, co na przykład elementy zbioru $\{4\}$. Nierozróżnialność elementów o tych samych nośnikach sprawia, że kodowanie składni podane w poprzednim ustępie traci różnowartościowość, stając się tym samym bezużyteczne. Gdyby więcej niż jedna formuła miała taki sam kod, nie moglibyśmy jednoznacznie odkodowywać informacji. Musimy zatem zmienić kodowanie składni w taki sposób, aby odzyskać różnowartościowość.

W dalszej części pracy pokażemy w jaki sposób możemy zinterpretować $\mathrm{FM}(\mathcal{N})$ w $\mathrm{FM}((\omega, \perp))$ — wprowadzimy w $\mathrm{FM}((\omega, \perp))$ arytmetykę na indeksach liczb pierwszych. Okaże się również, że kodowanie zdefiniowane w następujący sposób $\mathrm{GN}(\varphi) = p_{\ulcorner\varphi\urcorner}$ pozwoli nam otrzymać tłumaczenia funkcji $\mathrm{Name}(x)$ i $\mathrm{Subst}(x, z)$ na arytmetykę indeksów liczb pierwszych.

4 Interpretowalność $\mathrm{FM}(\mathcal{N})$ w $\mathrm{FM}((\omega, \perp))$

Pokażemy teraz, w jaki sposób zinterpretować FM–dziedzinę $\mathrm{FM}(\mathcal{N})$ w $\mathrm{FM}((\omega, \perp))$. Będzie to arytmetyka zbudowana na indeksach liczb pierwszych modeli względnej pierwszości. Najpierw wyjaśnimy czym jest i na czym polega interpretacja.

Niech σ i τ będą słownikami relacyjnymi. Niech σ zawiera predykaty $R_1, ..., R_k$ o arnościach $s_1, ..., s_k$. Ciąg $\overline{\varphi} = (\varphi_U, \varphi_{\approx}, \varphi_{R_1}, ..., \varphi_{R_k})$ formuł słownika τ jest prostą interpretacją pierwszego rzędu modeli słownika σ, gdy zmienną wolną φ_U jest x_1, zmiennymi wolnymi $\varphi_{\approx}$ są x_1, x_2 oraz zmiennymi wolnymi każdej spośród formuł φ_{R_i} są $x_1, ..., x_{s_i}$ dla $i = 1, ..., k$. Ciąg $\overline{\varphi}$ definiuje w modelu $\mathcal{A}$ słownika τ model słownika σ w następującym sensie:

Określamy najpierw zbiór U jest definiowany przez φ_U:

$$U = \{a : \mathcal{A} \models \varphi_U[a]\}$$

Relacja $\approx$ jest zadana przez formułę $\varphi_{\approx}$ — musi ona definiować kongruencję na U, czyli jest równoważnością na U, oraz dla dowolnego $i \leq k$, dowolnych $a_1, ..., a_{s_i}$ i $b_1, ..., b_{s_i}$ jeżeli dla każdego $j \leq s_i$ $a_j \approx b_j$, to $R_i(a_1, ..., a_{s_i}) \equiv R_i(b_1, ..., b_{s_i})$ (można powiedzieć, że elementy z tych samych klas abstrakcji relacji $\approx$ wchodzą, w dokładnie te same relacje na U).

Dla $i = 1, ..., k$ interpretacja R_i dana jest przez:
$\mathbf{R}_i(\mathbf{a}_1, ..., \mathbf{a}_{s_i})$ wtedy i tylko wtedy, gdy:

$\exists a_1 \in \mathbf{a}_1 ... \exists a_{s_i} \in \mathbf{a}_{s_i} \mathcal{A} \models \varphi_{R_i}[a_1, ..., a_{s_i}]$, gdzie $\mathbf{a}_1, ..., \mathbf{a}_{s_i}$ są klasami abstrakcji relacji definiowanej przez $\varphi_{\approx}$ na U.

Tak więc uniwersum modelu $I_{\overline{\varphi}}(B_n)$, to zbiór klas abstrakcji relacji $\approx$ dla elementów z U, czyli $U/_{\approx}$. Odpowiednie relacje $\mathbf{R}_1, ..., \mathbf{R}_k$ interpretujące predykaty $R_1, ..., R_k$ określamy jak wyżej.

Definicja 3. *Powiemy, że $\overline{\varphi}$ jest prostą interpretacją* $\mathrm{FM}(\mathcal{A})$ *w* $\mathrm{FM}(\mathcal{B})$ *jeżeli istnieje nieograniczona funkcja monotoniczna $f : \omega \longrightarrow \omega$ taka, że dla dowolnego $n \geq 1$:*

$$I_{\overline{\varphi}}(\mathcal{B}_n) \cong \mathcal{A}_{f(n)}$$

Pokażemy, że istnieje prosta interpretacja $\mathrm{FM}(\mathcal{N})$ w $\mathrm{FM}((\omega, \perp))$, czyli ciąg formuł $\overline{\varphi} = (\varphi_U, \varphi_{\approx}, \varphi_+, \varphi_\times)$ w języku względnej pierwszości spełniający wyżej wspomniane warunki.

Bardzo istotne podczas badania relacji względnej pierwszości jest pojęcie nośnika liczby naturalnej. Dla dowolnej liczby $a \in \omega$ definiujemy $\mathrm{Supp}(a) = \{p_i \ : \ p_i | a\}$, gdzie p_i jest i–tą liczbą pierwszą. Nośniki liczb dostarczają nam całej wiedzy, jaką możemy uzyskać za pomocą predykatu $\perp$, to znaczy nie możemy w $\mathrm{FM}((\omega, \perp))$ rozróżniać elementów o tych samych nośnikach. Definiujemy następującą relację $\approx$:

$$a \approx b \equiv_{df} Supp(a) = Supp(b)$$

$\approx$ jest oczywiście relacją równoważności na liczbach naturalnych, co wynika wprost z jej definicji i z faktu, iż identyczność jest relacją równoważności. Klasę abstrakcji liczby a przy relacji $\approx$ będziemy oznaczać standardowo przez $[a]$.

4.1 Twierdzenia pomocnicze

Na początku podamy trzy znane wyniki z teorii liczb, które będą potrzebne do uzyskania głównego twierdzenia tego rozdziału.

Twierdzenie 6. *(O liczbach pierwszych) Zdefiniujmy funkcję*

$$\pi(x) = \max_i \{p_i \leq x\} + 1.$$

Wtedy $\lim_{x \to \infty} \frac{\pi(x)}{\frac{x}{\ln(x)}} = 1.$

Jest to klasyczny wynik, którego dowód można znaleźć na przykład w [4].

Fakt 1. *Dla dowolnego $b \in \omega$ istnieje $K \in \omega$ takie, że dla dowolnego $n \geq K$ oraz dowolnych $i < b$ istnieje liczba pierwsza q taka, że $in \leq q < (i+1)n$.*

Sierpiński w [16] zauważył, że wystarcza $K = e^b$.

Fakt 2. *Niech $0 < \varepsilon < 1$. Istnieje $N \in \omega$ takie, że dla dowolnego $x \geq N$ przedział $(x, x(1+\varepsilon))$ zawiera liczbę pierwszą.*

Fakt 2 jest jednym z wniosków z twierdzenia o liczbach pierwszych, wspomnianego w [4].

Dowiedziemy teraz kilku lematów, za pomocą których będziemy budować interpretację $\mathrm{FM}(\mathcal{N})$ w $\mathrm{FM}((\omega, \perp))$.

Lemat 1. *Następujące formuły są definiowalne w języku względnej pierwszości:*

- $P(x) \equiv_{df} \forall y \forall z\, (\neg z \perp x \wedge \neg y \perp x \Rightarrow \neg z \perp y)$ *(x jest potęgą liczby pierwszej),*
- $\{p, q\} \equiv_{df} \forall z\, (z \perp a \equiv (z \perp p \wedge z \perp q))$ *funkcja zdefiniowana dla par liczb p i q, zwracająca jako wartość klasę elementu $a \approx pq$.*

Następujący lemat pokazuje wyrażalność w języku z samym predykatem $\perp$ następujących predykatów, pozwalających na operowanie na klasach równoważności relacji $\approx$.

Lemat 2. *Istnieją formuły $\varphi_{\cup}(x, y, z)$, $\varphi_{\cap}(x, y, z)$ oraz $\varphi_{-}(x, y, z)$ spełniające w dowolnym modelu względnej pierwszości M następujące warunki:*

- $M \models \varphi_{\cup}[a, b, c]$ *wtedy i tylko wtedy, gdy $Supp(a) \cup Supp(b) = Supp(c)$,*
- $M \models \varphi_{-}[a, b, c]$ *wtedy i tylko wtedy, gdy $Supp(a) - Supp(b) = Supp(c)$,*
- $M \models \varphi_{\cap}[a, b, c]$ *wtedy i tylko wtedy, gdy $Supp(a) \cap Supp(b) = Supp(c)$,*

dla dowolnych $a, b, c \in |M|$.

Dowód. Podamy explicite definicje tych formuł:

- $\varphi_{\cup}(x, y, z) \equiv_{df} \forall w\, (w \perp z \equiv (w \perp x \wedge w \perp y))$
- $\varphi_{-}(x, y, z) \equiv_{df} \forall w\, (P(w) \Rightarrow (\neg w \perp z \equiv (\neg w \perp x \wedge w \perp y)))$
- $\varphi_{\cap}(x, y, z) \equiv_{df} \exists w\, (\varphi_{-}(x, w, z) \wedge \varphi_{-}(x, y, w))$

$\square$

Pokażemy teraz jak wykorzystać powyższe wyniki w dowodzie twierdzenia o istnieniu interpretacji $\mathrm{FM}(\mathcal{N})$ w $\mathrm{FM}((\omega, \perp))$.

Twierdzenie 7. *Istnieje interpretacja prosta* $\overline{\varphi}$ $\mathrm{FM}(\mathcal{N})$ *w* $\mathrm{FM}((\omega, \perp))$ *taka, że dla dowolnego* $k \in \omega$ *istnieje* $n \in \omega$ *takie, że* $\overline{\varphi}$ *definiuje w modelu* $(\{0, ..., n-1\}, \perp)$ *relacje* $R_+(x, y, z)$ *i* $R_\times(x, y, z)$ *oraz*
$\mathrm{card}(\{[a]_\approx : (\{0, ..., n-1\}, \perp) \models \varphi_U[a]\}) \geq k$.

4.2 Dowód istnienia interpretacji $\mathrm{FM}(\mathcal{N})$ w $\mathrm{FM}((\omega, \perp))$

Naszym pierwszym celem jest zdefiniowanie relacji R_+ oraz $R_\times$ spełniających warunki:

- $R_+(x, y, z)$ wtedy i tylko wtedy, gdy $x \approx p_i$, $y \approx p_j$, $z \approx p_k$ oraz $i + j = k$
- $R_\times(x, y, z)$ wtedy i tylko wtedy, gdy $x \approx p_i$, $y \approx p_j$, $z \approx p_k$ oraz $ij = k$

R_+ i $R_\times$ mają zatem definiować relacje odpowiednio dodawania i mnożenia na indeksach liczb pierwszych, czyli mają być elementami poszukiwanej przez nas interpretacji.

Rozważmy formułę $\varphi_\approx(x, y) \equiv_{df} \forall z\,(z \perp x \equiv z \perp y)$. Zauważmy, że definiuje ona relację $\approx$ oraz jest kongruencją w dowolnych modelach względnej pierwszości. Znaczy to, że nie możemy rozróżnić elementów należących do tych samych klas abstrakcji relacji $\approx$, które na mocy poczynionych wyżej obserwacji można identyfikować z nośnikami poszczególnych elementów modelu.

Będziemy stosować oznaczenie ΠX dla oznaczenia iloczynu elementów zbioru X, dla dowolnego skończonego zbioru X.
Na mocy Lematu 2 posiadamy formuły, za pomocą których możemy zdefiniować przydatne funkcje podobne do wyżej zdefiniowanej $\{p, q\}$. Dla dowolnego $n \geq 2$ możemy zdefiniować określoną na liczbach naturalnych funkcję (mającą za wartości klasy abstrakcji relacji $\approx$) n–argumentową $\{p_1, ..., p_n\}_n$ indukcyjnie:
$\{p_1, p_2\}_2 = \{p_1, p_2\} \approx p_1 p_2$,
$\{p_1, ..., p_n, p_{n+1}\}_{n+1} = \Pi(\mathrm{Supp}(\{p_1, ..., p_n\}_n) \cup \mathrm{Supp}(p_{n+1})) \approx p_1 ... p_{n+1}$.
Dla dowolnego n funkcję $\{p_1, ..., p_n\}_n$ będziemy oznaczać po prostu przez $\{p_1, ..., p_n\}$.

Możemy teraz zdefiniować bardzo ważną dla naszych potrzeb relację $\prec$, którą wyrażać będzie następująca formuła:

$$\varphi_{\prec}(x,y) \equiv_{df} \exists z\,(P(z) \wedge x \perp z \wedge y \perp z \wedge \exists w\, \varphi_{\cup}(x,z,w) \wedge \neg \exists w\, \varphi_{\cup}(y,z,w))$$

Relacja $\prec$ jest oczywiście inwariantna ze względu na względną pierwszość. Porządkuje ona nośniki liczb naturalnych. Następujący lemat pokazuje, że na pewnym odcinku początkowym relacja zdefiniowana przez $\varphi_{\prec}(x,y)$ pokrywa się ze standardowym porządkiem na ω.

Lemat 3. *Dla dowolnego $c \in \omega$ istnieje $N \in \omega$ takie, że dla dowolnych $n > N$, dowolnych a, b spełniających $1 \leq a, b \leq n$ oraz $\max\{\Pi\mathrm{Supp}(a), \Pi\mathrm{Supp}(b)\} \leq c$ zachodzi:*
$(\{0,...,n-1\},\perp) \models \varphi_{\prec}[a,b]$ *wtedy i tylko wtedy, gdy* $\Pi\mathrm{Supp}(a) < \Pi\mathrm{Supp}(b)$.

Dowód. Niech $\mathcal{M} = (\{0,...,n-1\},\perp)$.
$(\rightarrow)$ Załóżmy, że $\mathcal{M} \models \varphi_{\prec}[a,b]$. Istnieje wtedy liczba pierwsza $d \in |\mathcal{M}|$ taka, że $d\Pi\mathrm{Supp}(a) \leq n-1$ oraz $d\Pi\mathrm{Supp}(b) > n-1$. Mamy zatem tezę: $\Pi\mathrm{Supp}(a) < \Pi\mathrm{Supp}(b)$.
$(\leftarrow)$ Niech $A_1 = \Pi\mathrm{Supp}(a)$ oraz $B_1 = \Pi\mathrm{Supp}(b)$. Załóżmy, że $A_1 < B_1$. $\varphi_{\prec}(a,b)$ jest spełniona dokładnie wtedy, gdy przedział $(\frac{n-1}{B_1}, \frac{n-1}{A_1}]$ zawiera pewną liczbę pierwszą. Najmniej optymistycznym przypadkiem jest, gdy $B_1 = A_1+1$ — wtedy aby otrzymać tezę musimy móc wskazać liczbę pierwszą w przedziale $(\frac{n-1}{B_1}, \frac{n-1}{B_1}(1+\frac{1}{A_1})]$. Na mocy Faktu 2 wiadomo, że wystarczy wziąć N odpowiednie dla $\varepsilon = \frac{1}{A_1}$. $\square$

Przedstawimy teraz narzędzie, za pomocą którego będziemy mogli kodować za pomocą liczb pierwszych ich pary. Posiadając takie kodowanie będziemy mogli kodować obliczenia. Kodowanie jest właściwe, jeżeli jest funkcyjne i różnowartościowe — wtedy kod jest wyznaczony jednoznacznie i można jednoznacznie odczytać z niego zakodowaną informację.
Zdefiniujemy kod w następujący sposób:

$$\begin{aligned}\mathrm{Code}(p,x,y,q) \equiv_{df}\ & P(p) \wedge P(q) \wedge P(x) \wedge P(y) \wedge \\ & \wedge \forall z \forall w\, [(\varphi_{\cup}(x,y,z) \wedge \varphi_{\cup}(p,z,w)) \Rightarrow \varphi_{\prec}(q,w)] \wedge \\ & \wedge \forall r\, [(P(r) \wedge \varphi_{\prec}(q,r)) \Rightarrow \exists z\, \exists w\, (\varphi_{\cup}(x,y,z) \wedge \\ & \wedge \varphi_{\cup}(p,z,w) \wedge \varphi_{\prec}(w,r))]\end{aligned}$$

Powyższa formuła wyraża fakt, że wszystkie jej argumenty są potęgami liczb pierwszych oraz, że q jest $\prec$–najmniejszym elementem większym niż $\{p, x, y\}$ (zmienna w pełni rolę wartości ostatniego wyrażenia). Znaczeniem $\mathrm{Code}(p, x, y, q)$ jest to, że q koduje zbiór $\{x, y\}$ jednoznacznie z dokładnością do $\approx$. p pełni rolę bazy kodowania — czyli przeskalowania, którego zastosowanie gwarantuje nam różnowartościowość kodowania (z dokładnością do $\approx$).

Poniżej zdefiniujemy formułę wyrażającą fakt, że p jest dobrą bazą kodowania, czyli że dla q_1, q_2 poniżej x w sensie $\prec$ kodowanie $\mathrm{Code}(p, q_1, q_2, c)$ jest różnowartościowe z dokładnością do $\approx$.

$$\begin{aligned}\mathrm{GoodBase}(p, x) \equiv_{df} P(p) \wedge \forall q_1 ... \forall q_4 \, \{[\bigwedge_{i \geq 4} (P(q_i) \wedge \varphi_{\preceq}(q_i, x)) \wedge \\ \wedge \neg \varphi_{\approx}(\{q_1, q_2\}, \{q_3, q_4\})] \Rightarrow \\ \Rightarrow \exists c_1 \exists c_2 \, (Code(p, q_1, q_2, c_1) \wedge \\ \wedge \, Code(p, q_3, q_4, c_2) \wedge \neg \varphi_{\approx}(c_1, c_2))\}\end{aligned}$$

Istnienie dobrej bazy kodowania obrazuje następujący lemat korzystający z Faktu 1.

Lemat 4. *Dla dowolnego $k \in \omega$ istnieje $N \in \omega$ oraz liczba pierwsza $p \leq N$ taka, że dla dowolnego $n \geq N$,* $\mathrm{Code}(p, x_1, x_2, z)$ *definiuje różnowartościowe (z dokładnością do $\approx$) kodowanie par liczb pierwszych mniejszych niż k w dowolnym modelu $(\{0, ..., n-1\}, \perp)$.*

Dowód. Ustalmy $k \in \omega$ oraz wybierzmy K przy pomocy Faktu 1 dla $b = k^2$. Niech teraz p będzie liczbą pierwszą większą niż K. Na mocy Faktu 1 p jest dobrą bazą kodowania we wszystkich modelach $(\{0, ..., n-1\}, \perp)$, dla $n \geq N = k^2 p$. □

Parę uporządkowaną liczb pierwszych definiujemy w sposób standardowy. Mając nieuporządkowaną parę liczb pierwszych definiujemy:

$$< x, y > =_{df} \{\{x\}, \{x, y\}\}$$

Lemat 4 gwarantuje nam istnienie dobrej bazy kodowania na pewnym początkowym segmencie dostatecznie dużych modeli. Wystarcza to jednak, ponieważ interesują nas jedynie asymptotyczne własności skończonych modeli względnej pierwszości. Wolno nam zatem oznaczać parę uporządkowaną przez $< x, y >$, pomijając informację o bazie kodowania.

Mając formułę $\varphi_{\prec}(x,y)$ oraz wiedząc, że definiuje ona liniowy porządek na pewnym początkowym odcinku indeksów klas abstrakcji relacji $\approx$ liczb pierwszych możemy w łatwy sposób zdefiniować relację następnika taką, że $S_{\prec}(p_i) \approx p_{i+1}$ (o ile $S(p_i)$ jest zdefiniowana w danym modelu):

$$S_{\prec}(x) \approx y \equiv_{df} \varphi_{\prec}(x,y) \wedge P(x) \wedge P(y) \wedge \forall z\,(P(z) \Rightarrow \neg(\varphi_{\prec}(x,z) \wedge \varphi_{\prec}(z,y))).$$

Następujące twierdzenie pozwoli nam na kodowanie obliczeń.

Twierdzenie 8. *Funkcje częściowe zdefiniowane na indeksach liczb naturalnych FM–reprezentowalne w modelach z $\perp$ i $\prec$ domknięte są na schemat rekursji prostej.*

Dowód. Niech $g : \omega^n \longrightarrow \omega$ oraz $h : \omega^{n+2} \longrightarrow \omega$ będą funkcjami na indeksach liczb pierwszych FM–reprezentowalnymi w modelach względnej pierwszości z $\prec$. Musimy pokazać, że funkcja $f : \omega^{n+1} \longrightarrow \omega$ zdefiniowana następująco:

$$f(0,\overline{x}) = g(\overline{x})$$

$$f(i+1,\overline{x}) = h(i+1,\overline{x},f(i,\overline{x}))$$

również jest FM–reprezentowalna w modelach względnej pierwszości z $\prec$. Dla przejrzystości dowodu załóżmy, że $n = 1$. Mając w języku predykaty $\perp$ i $\prec$ możemy, na mocy Lematu 4 zdefiniować kodowanie par liczb pierwszych $< x, y >$ za pomocą liczb pierwszych. Formuła definiująca $f(p_i, p_x) = p_t$ oznacza, że istnieje zbiór opisujący obliczenie funkcji f na argumentach p_i, p_x zwracające wartość p_t. Można je zapisać następująco:

$$\exists X\,\{< p_0, g(p_x) >\in X \wedge \forall p_z \forall p_w\,[\varphi_{\prec}(p_z,p_i) \Rightarrow (< p_{z+1}, p_w >\in X \Leftrightarrow$$
$$\Leftrightarrow \exists p_v\,(< p_z, p_v >\in X \wedge p_w \approx h(p_{z+1}, p_x, p_v)))] \wedge < p_i, p_t >\in X\}$$

Kwantyfikację po zbiorze X możemy łatwo zastąpić przez kwantyfikację pierwszego rzędu, ponieważ za X możemy wziąć a takie, że $X = Supp(a)$. Zatem funkcja f jest funkcją FM–reprezentowalną w modelach względnej pierwszości z $\prec$. $\square$

Korzystając z Twierdzenia 8 z formuł, które FM–reprezentują 0 i następnik $S_{\prec}(x)$ otrzymujemy poszukiwane przez nas formuły $\varphi_{+}(x,y,z)$ oraz $\varphi_{\times}(x,y,z)$ definiujące dodawanie i mnożenie na indeksach liczb pierwszych.

Brakuje nam już jedynie formuły $\varphi_U(x)$ definiującej uniwersum naszej interpretacji. Chcemy, żeby $\varphi_U(x)$ była prawdziwa w modelach względnej pierwszości o elementach x takich, że $P(x)$ oraz:

- Dla dowolnych y, z jeżeli $\varphi_U(y)$ i $\varphi_U(z)$, to istnieje w takie, że $P(w)$ i $\varphi_+(y, z, w)$ oraz poniżej w relacja $\prec$ obcięta do elementów będących potęgą liczby pierwszej jest liniowym porządkiem na klasach abstrakcji liczb pierwszych.
- Dla dowolnych y, z jeżeli $\varphi_U(y)$ i $\varphi_U(z)$, to istnieje w takie, że $P(w)$ i $\varphi_\times(y, z, w)$ oraz poniżej w relacja $\prec$ obcięta do elementów będących potęgą liczby pierwszej jest liniowym porządkiem na klasach abstrakcji liczb pierwszych.

Chcemy pokazać, że $\overline{\varphi} = (\varphi_U, \varphi_\approx, \varphi_+, \varphi_\times)$ jest interpretacją $\mathrm{FM}(\mathcal{N})$ w $\mathrm{FM}((\omega, \perp))$. Wystarczy pokazać, że zdefiniowana w następujący sposób funkcja $f(n) = \mathrm{card}(\{[a]_\approx : (\{0, ..., n\}, \perp) \models \varphi_U[a]\})$ rośnie monotonicznie i nieograniczenie wraz ze wzrostem mocy modelu względnej pierwszości.

- Monotoniczność jest oczywista. Jeżeli $n \geq m$, to: $\mathrm{card}(\{x : (\{0, ..., n\}, \perp) \models \varphi_U(x)\}) \geq \mathrm{card}(\{x : (\{0, ..., m\}, \perp) \models \varphi_U(x)\})$.
- Nieograniczoność. Ponieważ wiadomo, że istnieje nieskończenie wiele liczb pierwszych, to pozostaje pokazać, że $\prec$ obcięte do klas abstrakcji potęg liczb pierwszych jest liniowym porządkiem na pewnym segmencie początkowym, którego wielkość rośnie nieograniczenie. Niech dla pewnego modelu względnej pierwszości o mocy n relacja $\prec$ obcięta to klas abstrakcji potęg liczb pierwszych będzie liniowym porządkiem na zbiorze $\{[p_0], ..., [p_k]\}$ i niech $\neg p_k \prec p_{k+1}$ $(k > 0)$, czyli w szczególności nie istnieje reprezentant klasy abstrakcji $[\{2, p_{k+1}\}]$. Wystarczy, że weźmiemy większy model względnej pierwszości, taki by wspomniany reprezentant istniał, lecz nie istniał reprezentant klasy $[\{2, p_{k+1}\}]$, by otrzymać liniowość $\prec$ na $\{[p_0], ..., [p_{k+1}]\}$. Posiadając dowolnie długi liniowy porządek na klasach liczb pierwszych możemy zawsze wybrać model względnej pierwszości w taki sposób, by liczba klas abstrakcji relacji $\approx$ elementów tego modelu spełniających $\varphi_U(x)$ była większa, niż dowolnie z góry ustalona liczba naturalna.

Ponieważ mamy $\varphi_+(x, y, z)$, $\varphi_\times(x, y, x)$ definiują relacje dodawania i mnożenia na elementach z U oraz $\varphi_\approx(x, y)$ jest kongruencją na U, zatem:

$I_{\overline{\varphi}}(\{0, ..., n\}, \perp) \cong \mathcal{N}_{f(n)}$.

Zatem $\overline{\varphi}$ jest szukaną przez nas interpretacją FM($\mathcal{N}$) w FM($(\omega, \perp)$).

4.3 Tłumaczenia relacji na indeksy liczb pierwszych

Podamy teraz definicję tłumaczenia relacji arytmetycznej na relację na indeksach liczb pierwszych.

Definicja 4. *Niech $R \subseteq \omega^r$ będzie relacją arytmetyczną.*

$$R^* = \{(x_1, ..., x_r) : \exists a_1 ... \exists a_r \, (\bigwedge_{i \leq r} (x_i \approx p_{a_i}) \wedge R(a_1, ..., a_r))\}$$

Na mocy Twierdzenia 7 otrzymujemy następujący wniosek.

Wniosek 1. *Niech $R \subseteq \omega^r$ będzie relacją arytmetyczną. Jeżeli R jest FM–reprezentowalna w* FM($\mathcal{N}$)*, to R^* jest FM–reprezentowalna w* FM($(\omega, \perp)$).

Dowód. Niech $R \subseteq \omega^r$ będzie relacją arytmetyczną FM–reprezentowalną w FM($\mathcal{N}$). Pokażemy, że R^* jest FM–reprezentowalna w FM($(\omega, \perp)$). Niech $\varphi^*(x_1, ..., x_r)$ będzie formułą definiującą R^* w FM($(\omega, \perp)$). Przypomnijmy definicję R^*:

$$R^* = \{(x_1, ..., x_r) : \exists a_1 ... \exists a_r \, (\bigwedge_{i \leq r} (x_i \approx p_{a_i}) \wedge R(a_1, ..., a_r))\}$$

Ustalmy $b_1, ..., b_r \in \omega$. Jeżeli dla pewnego $j \leq r$ nie istnieje a_j takie, jak w powyższej definicji, czyli nie jest prawdą, że $b_j \approx p_{a_j}$, to $\varphi^*(b_1, ..., b_j)$ jest fałszywa we wszystkich modelach względnej pierwszości.
W przeciwnym wypadku dla wszystkich $i \leq r$ zachodzi $b_i \approx p_{a_i}$.
Jeżeli $\neg R(a_1, ..., a_r)$, to niech $\varphi(x_1, ..., x_r)$ formuła FM–reprezentująca R w FM($\mathcal{N}$), wtedy FM($\mathcal{N}$) $\models_{sl} \neg\varphi[a_1, ..., a_r]$, zatem na mocy Twierdzenia 7 FM($(\omega, \perp)$) $\models_{sl} \neg\varphi^*[b_1, ..., b_r]$.
Jeżeli $R(a_1, ..., a_r)$, to FM($\mathcal{N}$) $\models_{sl} \varphi[a_1, ..., a_r]$, zatem na mocy Twierdzenia 7 FM($(\omega, \perp)$) $\models_{sl} \varphi^*[b_1, ..., b_r]$.
Pokazaliśmy zatem, że $\varphi^*(x_1, ..., x_r)$ FM–reprezentuje R^* w FM($(\omega, \perp)$). □

W poprzednim punkcie niniejszej pracy wykazaliśmy, że funkcje Name(x) oraz Subst(x, z) są FM–reprezentowalne w FM($\mathcal{N}$). Na mocy powyższego wniosku istnieją relacje Name$^*(x) \approx y$ oraz Subst$^*(x, z) \approx w$, które są FM–reprezentowalne w FM($(\omega, \perp)$). Wykorzystamy to w kolejnym punkcie pracy, w dowodzie FM–wersji twierdzenia Tarskiego dla FM($(\omega, \perp)$).

5 FM–definicje prawdy w $\mathrm{FM}((\omega, \perp))$

Mamy już wszystkie narzędzia potrzebne do dowodu FM–wersji twierdzenia Tarskiego o niedefiniowalności prawdy dla $\mathrm{FM}((\omega, \perp))$.

Twierdzenie 9. *(FM–wersja lematu przekątniowego dla* $\mathrm{FM}((\omega, \perp))$*) Dla dowolnej formuły w języku względnej pierwszości $\varphi(x)$ z jedną zmienną wolną x istnieje zdanie ψ spełniające następujący warunek:*

$$\mathrm{FM}((\omega, \perp)) \models_{sl} \psi \equiv \varphi(\mathrm{GN}(\psi)).$$

Dowód. Pokazaliśmy, że w $\mathrm{FM}(\mathcal{N})$ FM–reprezentowalne są funkcje $\mathrm{Name}(x)$ oraz $\mathrm{Subst}(x, z)$. Wiemy również, że dzięki istnieniu interpretacji skończonych modeli arytmetyki dodawania i mnożenia w skończonych modelach względnej pierwszości, w $\mathrm{FM}((\omega, \perp))$ FM–reprezentowalne są $\mathrm{Name}^*(x)$ oraz $\mathrm{Subst}^*(x, z)$ — tłumaczenia tych funkcji na klasy równoważności relacji $\approx$ liczb pierwszych.
Jeżeli $\mathrm{GN}(\psi)$ jest kodem formuły ψ, takim jaki zdefiniowaliśmy w jednym z wcześniejszych punktów niniejszej pracy: $\mathrm{GN}(\psi) = p_{\ulcorner\psi\urcorner}$, to $\mathrm{Name}^*(x)$ oraz $\mathrm{Subst}^*(x, z)$ spełniają następujące warunki:

- $\mathrm{FM}((\omega, \perp)) \models_{sl} \mathrm{Subst}^*(\mathrm{GN}(\varphi(x)), \mathrm{GN}(t)) = \mathrm{GN}(\varphi(t))$
- $\mathrm{FM}((\omega, \perp)) \models_{sl} \mathrm{Name}^*(x) = \mathrm{GN}(\underbrace{s(s...s}_{x}(0)))$

Przedstawimy teraz następujące pomocnicze definicje:
Niech $\zeta(x) \equiv_{df} \varphi(\mathrm{Subst}^*(x, \mathrm{Name}^*(x)))$,
Niech $m =_{df} \mathrm{GN}(\zeta(x))$,
Niech $\psi \equiv_{df} \zeta(m)$.
Chcemy pokazać, że:

$$\mathrm{FM}((\omega, \perp)) \models_{sl} (\psi \equiv \varphi(\mathrm{GN}(\psi))).$$

W dostatecznie dużych modelach z $\mathrm{FM}((\omega, \perp))$ równoważne są następujące formuły:

- ψ
- $\zeta(m)$
- $\zeta(\mathrm{GN}(\zeta(x)))$
- $\varphi(\mathrm{GN}(\mathrm{Subst}^*(\mathrm{GN}(\zeta(x)), \mathrm{Name}^*(\mathrm{GN}(\zeta(x))))))$

- $\varphi(\mathrm{GN}(\zeta(\mathrm{Name}^*(\mathrm{GN}(\zeta(x))))))$
- $\varphi(\mathrm{GN}(\zeta(m)))$
- $\varphi(\mathrm{GN}(\psi))$

□

Twierdzenie 10. *(FM–wersja twierdzenia Tarskiego dla* $\mathrm{FM}((\omega, \perp))$*) Nie istnieje formuła* $\varphi(x)$ *z jedną zmienną wolną* x *w języku względnej pierwszości taki, że dla dowolnego zdania* ψ *w języku względnej pierwszości zachodzi:*

$$\mathrm{FM}((\omega, \perp)) \models_{sl} (\varphi(\mathrm{GN}(\psi)) \equiv \psi)$$

Dowód. Załóżmy nie wprost, że $\varphi(x)$ istnieje. Stosujemy FM–wersję lematu przekątniowego dla $\mathrm{FM}((\omega, \perp))$ i formuły $\neg\varphi(x)$ i otrzymujemy istnienie takiego zdania ψ, że:

$$\mathrm{FM}((\omega, \perp)) \models_{sl} \psi \equiv \neg\varphi(\mathrm{GN}(\psi))$$

Ponieważ φ jest FM–definicją prawdy, więc zachodzi:

$$\mathrm{FM}((\omega, \perp)) \models_{sl} \psi \equiv \varphi(\mathrm{GN}(\psi))$$

Zatem:

$$\mathrm{FM}((\omega, \perp)) \models_{sl} \psi \equiv \neg\psi$$

Sprzeczność — $\varphi(x)$, będące FM–definicją prawdy dla $\mathrm{FM}((\omega, \perp))$ nie może istnieć.

□

Wraz z końcem powyższego twierdzenia udało się osiągnąć cel naszej pracy, czyli pokazać, że do reprezentacji semantyki w skończonych modelach wystarcza już relacja względnej pierwszości.

Dodatek I: Twierdzenie o rozstrzygalności arytmetyki Skolema $\mathrm{Th}((\omega, \times))$

Korzystając z podstawowych faktów dotyczących języków regularnych i automatów skończonych pokażemy zaczerpnięty z [1] dowód następującego twierdzenia.

Twierdzenie 11. *Teoria pierwszego rzędu* $\mathrm{Th}((\omega, \times))$ *jest rozstrzygalna.*

Poniżej przedstawimy dowód rozstrzygalności teorii $\mathrm{Th}((\omega - \{0\}, \times))$, to jednak wystarczy dla rozstrzygalności teorii mnożenia o dziedzinie ω. Fakt ten zilustrujemy lematem.

Lemat 5. *Niech* $M = (\omega, \times, 0)$ *będzie modelem arytmetyki mnożenia oraz niech* $N = (\omega - \{0\}, \times')$ *będzie modelem arytmetyki mnożenia bez zera. Istnieje efektywny przekład* $\varphi \mapsto \varphi^*$ *formuł arytmetyki mnożenia na formuły arytmetyki mnożenia bez zera, taki że dla dowolnego wartościowania* $\overline{a}$ *w* $\omega - \{0\}$ *zachodzi:* $M \models \varphi[\overline{a}] \Leftrightarrow N \models \varphi^*[\overline{a}]$.

Dowód. Najpierw skonstruujemy przekład $\varphi \mapsto \varphi^*$.

- Jeżeli $\varphi(\overline{x}, \overline{y})$ jest formułą atomową, czyli φ jest postaci: $t_1(\overline{x}) = t_2(\overline{y})$, to:
  - Jeżeli 0 występuje tylko w jednym z termów $t_1(\overline{x}), t_2(\overline{x})$, to $\varphi^* =_{df} \exists x\, x \neq x$,
  - Jeżeli 0 występuje zarówno w termie $t_1(\overline{x})$ jak i w $t_2(\overline{x})$, to $\varphi^* =_{df} \exists x\, x = x$,
  - W przeciwnym wypadku $\varphi^* =_{df} \varphi$.
- Jeżeli $\varphi(\overline{x})$ jest postaci $\psi(\overline{x}) \to \vartheta(\overline{x})$ dla pewnych formuł $\psi(\overline{x})$ i $\vartheta(\overline{x})$, to $\varphi^*(\overline{x}) =_{df} \psi^*(\overline{x}) \to \vartheta^*(\overline{x})$,
- Jeżeli $\varphi(\overline{x})$ jest postaci $\neg\psi(\overline{x})$ dla pewnej formuły $\psi(\overline{x})$, to $\varphi^*(\overline{x}) =_{df} \neg\psi^*(\overline{x})$,
- Jeżeli $\varphi(\overline{x})$ jest postaci $\forall y\, \psi(y, \overline{x})$ dla pewnej formuły $\psi(y, \overline{x})$, to ponieważ $\forall y\, \psi(\overline{x}) \equiv \psi(0, \overline{x}) \wedge \forall y\, (y \neq 0 \Rightarrow \varphi(y, \overline{x}))$ więc ustalamy $\varphi^*(\overline{x}) =_{df} \psi^*(0, \overline{x}) \wedge \forall y\, \psi^*(y, \overline{x})$.

Pozostaje sprawdzić, że zachodzi $M \models \varphi[\overline{a}] \Leftrightarrow N \models \varphi^*[\overline{a}]$ dla dowolnego wartościowania $\overline{a}$ w $\omega - \{0\}$. Pokażemy to stosując indukcję po budowie formuły.

Jeżeli $\varphi(\overline{x})$ jest formułą atomową, to jest postaci: $a_1 \times ... \times a_n = b_1 \times ... \times b_m$ dla termów atomowych $a_1, ..., a_n, b_1, ..., b_m$. Zatem jeżeli jednym z a_i lub b_j dla $i \leq n$, $j \leq m$ jest 0, to wartość odpowiednio lewej lub prawej strony równości wynosi 0 przy dowolnym wartościowaniu $\overline{a}$ w $\omega - \{0\}$. Zatem jeżeli tylko po jednej stronie wystąpi term 0, to dla dowolnego wartościowania w $\omega - \{0\}$ równość będzie fałszywa w modelu M, podobnie jak w N fałszywa będzie $\varphi^* = \exists x\, x \neq x$. Jeżeli zero wystąpi po obu stronach, to równość oczywiście będzie prawdziwa — dla dowolnego wartościowania będzie równoważna $0 = 0$, ale $\varphi^* = \exists x\, x = x$ również jest prawdziwa dla dowolnego wartościowania. Jeżeli zero nie występowało jako term atomowy, to oczywiście również mamy równoważność.
Dowód dla negacji, implikacji oraz kwantyfikatora wynika wprost z definicji spełniania.

$\square$

Mając efektywny przekład, o którym mowa w Lemacie 5 wiemy jak przenieść wynik o rozstrzygalności teorii $\mathrm{Th}((\omega - \{0\}, \times))$, na rozstrzygalność $\mathrm{Th}((\omega, \times))$. Pokażemy teraz rozstrzygalność teorii $\mathrm{Th}((\omega - \{0\}, \times))$.

Dowód. Podamy algorytm rozstrzygający, czy zadane zdanie φ należy do teorii $\mathrm{Th}((\omega - \{0\}, \times))$, czy należy do niej jego zaprzeczenie $\neg\varphi$. W tym celu potrzebujemy odpowiedniego kodowania, za pomocą którego będziemy zadawać wejście dla algorytmu.
Niech $a \in \omega - \{0\}$ i niech $a = p_0^{n_0} ... p_k^{n_k}$ będzie rozkładem na liczby pierwsze liczby a. Kod liczby a definiujemy następująco:
$c(a) = \{\#0^{r_k}\mathrm{bin}(n_k)\#...\#0^{r_0}\mathrm{bin}(n_0)\#, \text{ dla } r_k, ..., r_0 \in \omega\}$ gdzie dla $s \in \omega$ $\mathrm{bin}(s)$ to zapis binarny liczby s. W szczególności $\varepsilon \in c(1)$ (ε oznacza słowo puste).
Zauważmy, że kod $c(a)$ liczby $a \in \omega - \{0\}$ nie jest wyznaczony jednoznacznie, z uwagi na dowolność liczb $r_k, ..., r_0$. Niejednoznaczność ta jest jednak zamierzona i zostanie wykorzystana przy kodowaniu n–tek dodatnich liczb naturalnych. Kody liczb możemy traktować jako słowa nad alfabetem $\mathcal{L}_1 = \{0, 1, \#\}$. Dla zakodowania n–tki liczb naturalnych $w = (a_1, ..., a_n)$ wprowadzamy alfabet $\mathcal{L}_n = \{\{\#\}_n, \{0, 1\}_n\}$ składający się z kolumn złożonych z n symboli # lub kolumn zer i jedynek (w dowolnej konfiguracji). Niech $r_i(w)$ będzie i–tym wierszem kodu n–tki w — możemy $r_i(w)$ traktować jako słowo nad alfabetem $\mathcal{L}_1$. Eksplicite możemy napisać, że $c(a_1, ..., a_n) = \{w \in \mathcal{L}_n^* : \forall i \leq n\, r_i(w) \in c(a_i)\}$.

Kod n–tki $(a_1, ..., a_n)$ — $c(a_1, ..., a_n)$ to dowolne słowo v nad alfabetem $\mathcal{L}_n$ takie, że dla $1 \leq i \leq n$ wiersz $r_i(v)$ jest kodem a_i. Widać teraz, że niejednoznaczność w kodowaniu pojedynczych liczb pozostawiamy po to, aby móc znaleźć odpowiednio długi kod liczby w taki sposób, by móc zsynchronizować długości kodów $r_i(v)$, otrzymując tym samym kod będący słowem nad $\mathcal{L}_n$. Dla dowolnej relacji $R \subseteq (\omega - \{0\})^r$ definiujemy $C(R) = \{v \in \mathcal{L}_n^* : \exists w \in R\,(v \in c(w))\}$. Pokażemy teraz w jaki sposób skonstruować dla dowolnej formuły arytmetyki mnożenia $\varphi(x_1, ..., x_r)$ automat skończony A_φ taki, że $\mathcal{L}(A_\varphi) = C(\varphi^{(\omega-\{0\},\times),x_1,...,x_r})$, czyli język akceptowany przez A_φ jest równy zbiorowi kodów relacji definiowanej w $(\omega-\{0\}, \times)$ przez formułę $\varphi(x_1, ..., x_r)$. Musimy jeszcze rozwiązać problem formuł bez zmiennych wolnych, ponieważ nie zdefiniowaliśmy języka $\mathcal{L}_0$ przy pomocy, którego uzyskiwalibyśmy kody relacji 0 argumentowych. Dla zdania ψ rozważamy formułę $\psi^*(x) \equiv_{df} \psi \wedge x = x$. Bez trudu zauważamy, że:

- Jeżeli $(\omega - \{0\}, \times) \models \psi$, to $\mathcal{L}(A_{\psi^*}) = \bigcup_{i \in (\omega-\{0\})} c(i)$
- Jeżeli $(\omega - \{0\}, \times) \not\models \psi$, to $\mathcal{L}(A_{\psi^*}) = \varnothing$

Niech ψ zdanie arytmetyki Skolema. Wtedy:

- $(\omega - \{0\}, \times) \models \psi$ wtedy i tylko wtedy, gdy $\varepsilon \in \mathcal{L}(A_{\psi^*})$.

$(\rightarrow)$ Jeżeli $(\omega-\{0\}, \times) \models \psi$, to $(\omega-\{0\}, \times) \models \psi \wedge x = x[1]$, czyli $\varepsilon \in c(1) \subset \mathcal{L}(A_{\psi^*})$.
$(\leftarrow)$ Jeżeli $(\omega - \{0\}, \times) \not\models \psi$, to dla dowolnego $i = 1, 2, ...$ $(\omega - \{0\}, \times) \not\models \psi \wedge x = x[i]$, czyli $\mathcal{L}(A_{\psi^*}) = \varnothing$, a w szczególności nie prawda, że $\varepsilon \in \mathcal{L}(A_{\psi^*})$.

Wykorzystamy indukcyjną definicję formuły, aby pokazać, że dla każdej formuły z n zmiennymi wolnymi możemy skonstruować wyrażenie regularne, a więc również automat skończony, który akceptuje dokładnie kody n–tek $(a_1, ..., a_n)$ takich, że $(\omega - \{0\}, \times) \models \varphi[a_1, ..., a_n]$.

- Dla $x = y$ język regularny to:

$$\left(\begin{pmatrix}\#\\\#\end{pmatrix}\begin{pmatrix}0\\0\end{pmatrix}^*\left(\begin{pmatrix}0\\0\end{pmatrix}+\begin{pmatrix}1\\1\end{pmatrix}\right)^+\right)^+\begin{pmatrix}\#\\\#\end{pmatrix}$$

- Dla $x \times y = z$ język regularny definiujemy za pomocą definicji języka dla równości. Jeżeli $x = p_0^{x_0}...p_k^{x_k}$ oraz $y = p_0^{y_0}...p_k^{y_k}$ wtedy musi zachodzić $z = p_0^{x_0+y_0}...p_k^{x_k+y_k}$. Biorąc pod uwagę fakt, że istnieje automat skończony dla dodawania binarnego, co pozwala nam obliczać efektywnie $x_i + y_i$ dla $i = 1, ..., k$.

Dla kombinacji Boole'owskich formuł atomowych konstruujemy automaty skończone korzystając z faktów, że dopełnienia i przecięcia języków regularnych są językami regularnymi — w ten sposób skonstruujemy automaty skończone dla koniunkcji i negacji. Pozostaje przeprowadzenie konstrukcji dla jednego z kwantyfikatorów. Niech $\varphi(x_1, ..., x_n) = \exists x{+}1\psi(x_1, ..., x_{n+1})$. Z założenia indukcjnego mamy automat skończony A_ψ. Na bazie A_ψ budujemy automat skończony A_φ następująco:

- $Q_{A_\varphi} = Q_{A_\psi}$
- $F_{A_\varphi} = F_{A_\psi}$
- $q_0^{A_\varphi} = q_0^{A_\psi}$
- $$\delta_{A_\varphi}(q, \begin{pmatrix} d_1 \\ \vdots \\ d_n \end{pmatrix}) = \begin{cases} \delta_{A_\psi}(q, \#_{n+1}) & gdy \begin{pmatrix} d_1 \\ \vdots \\ d_n \end{pmatrix} = \#_n \\ \delta_{A_\psi}(q, \begin{pmatrix} d_1 \\ \vdots \\ d_n \\ 0 \end{pmatrix}) \cup \delta_{A_\psi}(q, \begin{pmatrix} d_1 \\ \vdots \\ d_n \\ 1 \end{pmatrix}) & gdy \begin{pmatrix} d_1 \\ \vdots \\ d_n \end{pmatrix} \in \{0,1\}_n \end{cases}$$

Pozostaje pokazać, że powyższa konstrukcja jest właściwa, czyli, że dla dowolnej formuły $\varphi(x_1, ..., x_r)$ zachodzi $\mathcal{L}(A_{\varphi(x_1,...,x_r)}) = C(\varphi^{(\omega-\{0\},\times),x_1,...,x_r})$. Przeprowadzimy dowód indukcyjny po budowie formuły.
Dla formuły atomowej postaci $x = y$ jest jasne, że język akceptowany przez $A_{x=y}$ to zbiór kodów par (i, i) dla $i = 1, 2, ...$.
Dla formuły atomowej postaci $x \times y = z$ również zachodzi równość, ponieważ istnieje automat skończony wykonujący dodawanie binarne.
Krok indukcyjny dla kombinacji Boole'owskich formuł przebiega następująco. Ponieważ klasa języków akceptowalnych przez automaty skończone domknięta jest na operacje typu: suma, przecięcie, czy dopełnienie, oraz dla dowolnego $n > 0$ mamy automaty skończone A_n akceptujące n–argumentowe relacje pełne (wyznaczające zbiór wszystkich kodów n–tek), mamy zatem:

- Krok dla negacji. $\varphi(\overline{x}) = \neg\psi(\overline{x})$, oraz A_ψ automat akceptujący dokładnie kody n–tek $\overline{a}$ takich, że $\psi[\overline{a}] \in \mathrm{Th}((\omega - \{0\}, \times))$. Wtedy

$A_n \cap \overline{A_\psi}$ automat akceptujący dokładnie kody n–tek $\bar{b}$ takich, że $\varphi[\bar{b}] \in \mathrm{Th}((\omega - \{0\}, \times))$.

- Krok dla alternatywy. Mamy automaty dla $\psi(\overline{x})$ oraz $\vartheta(\overline{y})$ i chcemy skonstruować automat dla $\varphi(\overline{x}, \overline{y}) = \psi(\overline{x}) \vee \vartheta(\overline{y})$. W tym celu budujemy automaty dla $\psi'(\overline{x}, \overline{y})$ oraz $\vartheta(\overline{x}, \overline{y})$, a językiem dla $\varphi(\overline{x}, \overline{y})$ będzie suma języków dla $\psi'(\overline{x}, \overline{y})$ i $\vartheta(\overline{x}, \overline{y})$.

- Krok dla kwantyfikatora. Mamy automat A_ψ akceptujący kody $(n+1)$–tek $(\overline{a}, b)$ takich, że $(\omega - \{0\}, \times) \models \psi[\overline{a}, b]$. Z konstrukcji automatu $A_\varphi(\overline{x})$ wynika, że akceptuje on n–tki $\overline{a}$ dokładnie wtedy, gdy istnieje b takie, że $(\omega - \{0\}, \times) \models \psi[\overline{a}, b]$.

Zatem dla każdej formuły $\varphi(\overline{x})$ arytmetyki Skolema skonstruujemy indukcyjnie, korzystając z konstrukcji formuły φ automat skończony, który rozpoznaje dokładnie kody n–tek spełniających tę formułę. Ponieważ dla dowolnego języka regularnego $\mathcal{L}$ rozstrzygalne jest pytanie: *Czy* $\varepsilon \in \mathcal{L}$*?*, zatem rozstrzygalna jest również teoria $\mathrm{Th}((\omega - \{0\}, \times))$. $\square$

Dodatek II: Twierdzenie o FM–reprezentowalności w FM($\mathcal{N}$)

Twierdzenie 12. *(O FM–reprezentowalności) Niech $R \subseteq \omega^r$ relacja arytmetyczna. R jest FM–reprezentowalna w* FM($\mathcal{N}$) *wtedy i tylko wtedy, gdy $R \in \Delta_2$.*

Dla dowodu twierdzenia o FM–reprezentowalności najpierw pokażemy, że FM–reprezentowalne są wszystkie rekurencyjnie przeliczalne relacje arytmetyczne.

Twierdzenie 13. *Niech $R \subseteq \omega^r$ relacja arytmetyczna i $R \in \Sigma_1$. Wtedy R jest FM–reprezentowalna w* FM($\mathcal{N}$).

Dowód. Skonstruujemy najpierw predykat Kleenego $T(e, c, n)$ dla maszyny RAM wyrażający fakt, że c jest kodem zakończonego obliczenia maszyny RAM o kodzie e na wejściu o kodzie n. Z konstrukcji wynikać będzie, że w $T(e, c, n)$ wszystkie kwantyfikatory są ograniczone przez c zatem, że jest to predykat rekurencyjny. Dowolna relacja rekurencyjnie przeliczalna może zostać wyrażona formułą $\exists c\, T(e, c, n)$.
Potrzebujemy kodów dla wszystkich pojedynczych instrukcji maszyny RAM. Niech 2^k będzie kodem dla $Z(k)$ — instrukcji zerowania k–tego rejestru (i zwiększenia licznika rozkazów o 1), 3^k będzie kodem dla $S(k)$ — instrukcji zwiększenia k–tego rejestru o 1 (i zwiększenia licznika rozkazów o 1), $5^k 7^l$ będzie kodem dla $C(k, l)$ — instrukcji zapisania w l–tym rejestrze wartości k–tego rejestru (i zwiększenia licznika rozkazów o 1) oraz niech $11^k 13^l 17^q$ będzie kodem dla $I(k, l, q)$ — instrukcji skoku do q–tego rozkazu w przypadku, gdy wartości k–tego i l–tego rejestru są równe lub przejścia do następnego rozkazu w przeciwnym przypadku. Kod i–tego rozkazu oznaczymy przez $r(i)$. Maszyna RAM kończy pracę, gdy wskaźnik rozkazów znajdzie się na pierwszym pustym polu.
Możemy teraz zakodować maszynę RAM, a dokładniej ciąg poleceń składających się na algorytm, jaki maszyna ta ma wykonywać. Jeżeli maszyna RAM ma s rozkazów, to jej kodem będzie kod ciągu $e(0), ..., e(s-1)$, czyli liczba $e = p_0^{r(0)+1} ... p_{s-1}^{r(s-1)+1}$.
Wejściem maszyny RAM jest pewna konfiguracja wartości rejestrów, możemy zatem ją również zakodować za pomocą kodowania ciągów. Jeżeli maszyna ma t rejestrów oraz jej wejście to $n(0), ..., n(t-1)$, to kodem wejścia jest kod tego ciągu, czyli $n = p_0^{e(1)+1} ... p_{t-1}^{e(t-1)+1}$.

Aby wyrazić fakt, że pewien ciąg jest obliczeniem pewnej maszyny RAM na jakimś wejściu musimy móc wyrażać konfigurację maszyny na każdym kroku jej obliczenia. Potrzebujemy do tego wiedzy na temat: który rozkaz maszyna musi aktualnie wykonać oraz jakie wartości zawierają rejestry. Konfigurację maszyny RAM zakodujemy zatem jako następujący ciąg długości $t+1$: $u, w(0), ..., w(t-1)$, gdzie u jest numerem rozkazu, który maszyna ma aktualnie wykonać, a $w(i)$ są wartościami rejestrów dla $i = 1, ..., t-1$. Kod konfiguracji powstaje jako kod powyższego ciągu w analogiczny sposób.
Mamy już wszystkie narzędzia do skonstruowania predykatu Kleenego — $T(e, c, n)$. $T(e, c, n)$ jest prawdziwe, gdy e jest kodem ciągu maszyny RAM, n jest kodem ciągu początkowej zawartości rejestrów oraz c kodem obliczenia maszyny RAM, czyli kodem ciągu konfiguracji postaci $(u, w(0), ..., w(t-1))$ takim, że:

- $c_0 = (0, n(0), ..., n(t-1))$
- $c_{lh(c)-1} = \mathrm{lh}(e)$
- Dla $i < \mathrm{lh}(c) - 1$ c_{i+1} powstaje z c_i w następującym sensie:
  - Jeżeli dla pewnego $k < \mathrm{lh}(n)$ $e((c_i)_0) = 2^k$ (czyli w $(i+1)$–szym kroku należy wykonać wyzerowanie k–tego rejestru), to $(c_{i+1})_p = (c_i)_p$ dla $0 < p \neq k+1$ oraz $(c_{i+1})_{k+1} = 0$. Dodatkowo $(c_{i+1})_0 = (c_i)_0 + 1$ (Po wykonaniu $(i+1)$–szego kroku wyzerowany zostaje k–ty rejestr oraz wskaźnik rozkazów został przesunięty na kolejny rozkaz),
  - Jeżeli dla pewnego $k < \mathrm{lh}(n)$ $e((c_i)_0) = 3^k$ (czyli w $(i+1)$–szym kroku należy zwiększyć o 1 wartość k–tego rejestru), to $(c_{i+1})_p = (c_i)_p$ dla $0 < p \neq k+1$ oraz $(c_{i+1})_{k+1} = (c_i)_{k+1} + 1$. Dodatkowo $(c_{i+1})_0 = (c_i)_0 + 1$ (Wartość k–tego rejestru została zwiększona o 1 oraz wskaźnik rozkazów został przesunięty na kolejny rozkaz),
  - Jeżeli dla pewnych $k, l < \mathrm{lh}(n)$ $e((c_i)_0) = 5^k 7^l$ (czyli w $(i+1)$–szym kroku należy skopiować wartość k–tego rejestru do l–tego rejestru), to $(c_{i+1})_p = (c_i)_p$ dla $0 < p \neq l+1$ oraz $(c_{i+1})_{l+1} = (c_i)_{k+1} + 1$. Dodatkowo $(c_{i+1})_0 = (c_i)_0 + 1$ (Wartość k–tego rejestru została zapisana w l–tym rejestrze oraz wskaźnik rozkazów został przesunięty na kolejny rozkaz),
  - Jeżeli dla pewnych $k, l < \mathrm{lh}(n)$ oraz $q \leq \mathrm{lh}(e)$ $e((c_i)_0) = 11^k 13^l 17^q$ (czyli $(i+1)$–szym kroku należy porównać wartości k–tego oraz

l–tego rejestru i jeżeli są one równe, to ustawić wskaźnik rozkazów na q–ty rozkaz, a w przeciwnym przypadku przejść do następnego rozkazu), to jeżeli $(c_i)_{k+1} = (c_i)_{l+1}$, to $(c_{i+1})_0 = q$ i $(c_{i+1})_p = (c_i)_p$ dla $0 < p$, oraz jeżeli $(c_i)_{k+1} \neq (c_i)_{l+1}$, to $(c_{i+1})_0 = (c_i)_0 + 1$ i $(c_{i+1})_p = (c_i)_p$ dla $0 < p$.

c_0 jest konfiguracją początkową obliczenia: licznik rozkazów znajduje się na 0, a rejestry zawierają wejście algorytmu. Drugi warunek wyraża fakt zatrzymania się obliczenia po $(\mathrm{lh}(c) - 1)$–szym kroku, czyli ustawienie wskaźnika rozkazów na wartość, pod którą nie znajduje się już żaden rozkaz. Wskaźnik rozkazów w $(i + 1)$–szym kroku obliczenia ustawiony jest na $(c_i)_0$ i w zależności od zapisanego tam rozkazu, którego kodem jest $e((c_i)_0)$ wykonywana jest odpowiednia instrukcja.

Otrzymaliśmy rekurencyjny predykat $T(e, c, n)$. Zauważmy, że wszystkie kwantyfikatory w jego definicji są ograniczone przez termy o wartościach nie większych niż c.
Jeżeli w modelu standardowym prawdziwe jest zdanie $\neg\exists c\, T(e, c, n)$ dla ustalonych e i n, to we wszystkich skończonych modelach również będziemy mieć $\neg\exists c\, T(e, c, n)$ — zatem $\mathrm{FM}(\mathcal{N}) \models_{sl} \neg\exists c\, T(e, c, n)$. Jeżeli dla ustalonych e i n w modelu standardowym prawdziwe jest zdanie $\exists c\, T(e, c, n)$, to ponieważ wszystkie kwantyfikatory w $T(e, c, n)$ są ograniczone, zatem wartość logiczna tego zdania będzie taka sama w modelach większych niż c — zachodzi wtedy $\mathrm{FM}(\mathcal{N}) \models_{sl} \exists c\, T(e, c, n)$. Czyli $\exists c\, T(e, c, n)$ FM–reprezentuje dowolną (w zależności od wyboru e) relację rekurencyjnie przeliczalną. □

Możemy teraz przejść do dowodu głównego twierdzenia.

Dowód. Niech $R \subseteq \omega^r$ będzie relacją arytmetyczną.
$(\Rightarrow)$ Załóżmy, że R jest FM–reprezentowana w $\mathrm{FM}(\mathcal{N})$ przez formułę arytmetyczną $\varphi(x_1, ..., x_r)$. Z definicji FM–reprezentowalności mamy dla dowolnych $a_1, ..., a_r \in \omega$:

- $R(a_1, ..., a_r) \equiv \mathrm{FM}(\mathcal{N}) \models_{sl} \varphi[a_1, ..., a_r]$
- $\neg R(a_1, ..., a_r) \equiv \mathrm{FM}(\mathcal{N}) \models_{sl} \neg\varphi[a_1, ..., a_r]$

Czyli z definicji $\models_{sl}$:

1. $R(a_1, ..., a_r) \equiv \exists k \forall n > k\, \mathcal{N}_n \models \varphi[a_1, ..., a_r]$
2. $\neg R(a_1, ..., a_r) \equiv \exists k \forall n > k\, \mathcal{N}_n \models \neg\varphi[a_1, ..., a_r]$

Ponieważ relacja spełniania w skończonych modelach jest rekurencyjna, to z warunku 1. relacja $R(x_1, ..., x_r)$ definiowana jest Σ_2 formułą. Z warunku 2. wiemy, że dopełnienie relacji R $\neg R(x_1, ..., x_r)$ również definiowana jest Σ_2 formułą. Zatem $R \in \Delta_2$.
($\Leftarrow$) Załóżmy teraz, że $R \in \Delta_2$. Skorzystamy z faktu, że klasa relacji Δ_2 jest dokładnie równa klasie relacji rekurencyjnych z rekurencyjnie przeliczalną wyrocznią, czyli obliczalnych na przykład przez maszynę RAM o dodatkowej instrukcji: dla pewnego ustalonego zbioru rekurencyjnie przeliczalnego A, maszyna w jednym kroku odpowiada na pytania: *Czy* $x \in A$? dla pewnego $x \in \omega$.
Niech M będzie maszyną RAM z rekurencyjnie przeliczalną wyrocznią A rozpoznającą relację R. Ponieważ A jest rekurencyjnie przeliczalna więc istnieje maszyna RAM N, która zatrzymuje się dokładnie wtedy, gdy na wejściu otrzymuje $x \in A$. Na mocy Twierdzenia 13 wiemy, że zbiór A jest FM–reprezentowalny — niech $\psi(x)$ będzie formułą FM–reprezentującą A w $\mathrm{FM}(\mathcal{N})$.
Formuła $\varphi(x_1, ..., x_r)$ FM–reprezentująca relację R można wyrazić następująco:
Istnieje obliczenie maszyny M na wejściu $x_1, ..., x_r$ takie, że dla każdego pytania do wyroczni: *Czy* $y \in A$*?*, odpowiedź jest pozytywna wtedy i tylko wtedy, gdy $\psi(y)$.
Ponieważ obliczenie jest skończone, to w jego toku zostaje wykonanych jedynie skończenie wiele pytań do wyroczni: czy $v_1, ..., v_s \in A$? Jeżeli $\varphi(a_1, ..., a_r)$ jest prawdziwe w modelu standardowym, to w odpowiednio dużych modelach istnieją świadkowie dla pozytywnego rozstrzygnięcia wszystkich tych pytań oraz kod zakończonego obliczenia maszyny M. W przeciwnym przypadku $\varphi(a_1, ..., a_r)$ jest fałszywe we wszystkich modelach skończonych. Mamy zatem dla dowolnych $a_1, ..., a_r \in \omega$:

- $R(a_1, ..., a_r) \Leftrightarrow \mathrm{FM}(\mathcal{N}) \models_{sl} \varphi[a_1, ..., a_r]$,
- $\neg R(a_1, ..., a_r) \Leftrightarrow \mathrm{FM}(\mathcal{N}) \models_{sl} \neg\varphi[a_1, ..., a_r]$.

Zatem R jest FM–reprezentowana w $\mathrm{FM}(\mathcal{N})$ przez formułę $\varphi(x_1, ..., x_r)$.

$\square$

Literatura

[1] Bès A. *A survey of Arithmetical Definability.* A tribute to Maurice Boffa, Special Issue of Belg. Math. Soc., srt. 1–54, 2002.

[2] Carnap R. *Logische Syntax der Sprache.* Springer, Wiedeń, 1934. Polski przekład: *Logiczna składnia języka.* PWN, Warszawa, 1995.

[3] Gödel K. *Über formal unentscheidbare Sätze der "Principia Mathematica" und verwandter Systeme I.* Monatshefte für Mathematik und Physik, 38:173–198, 1931.

[4] Jameson G.J.O *The prime number theorem.* Cambridge University Press, 2003.

[5] Kołodziejczyk L.A. *Truth definitions in finite models.* The Journal of Symbolic Logic, 69:183–200, 2004.

[6] Krynicki M., Zdanowski K. *Theories of arithmetics in finite models.* The Journal of Symbolic Logic, 70(1):1–28, 2005.

[7] Krynicki M., Mostowski M., Zdanowski K. *Finite Arithmetics.* Fundamenta Informaticae 81(1–3):183–202, 2007.

[8] Mostowski A. *On Direct Products of Theories.* The Journal of Symbolic Logic 17:1–31, 1952.

[9] Mostowski M. *On representing concepts in finite models.* Mathematical Logic Quarterly, 47:513–523, 2001.

[10] Mostowski M. *On representing semantics in finite models.* W: A. Rojszczak, J. Cachro, G. Kurczewski *Philosophical Dimensions of Logic and Science*, str. 15–28, Kluwer Academic Publishers, 2003.

[11] Mostowski M. *Limiting recursion, FM–representability, and hypercomputations.* Manuskrypt, 2008.

[12] Mostowski M., Wasilewska A.E. *Arithmetic of divisibility in finite models.* Mathematical Logic Quarterly 50(2):169–174, 2004.

[13] Mostowski M., Zdanowski K. *FM–representability and Beyond.* W: Cooper B., Loewe B., Torenvliet L. Proceedings of the conference Computability in Europe, vol. 3526 of Lecture Notes in Computer Science, str. 358–367, Springer, 2005.

[14] Mostowski M., Zdanowski K. *Coprimality in finite models.* W: Computer Science Logic: 19th International Workshop, CSL 2005, vol. 3634 of Lecture Notes in Computer Science, str. 263–275, Springer, 2005.

[15] Mycielski J. *Analysis without actual infinity.* The Journal of Symbolic Logic, 46:625-633, 1981.

[16] Sierpiński W. *Elementary Theory of Numbers.* PWN, Warszawa, 1964.

[17] Skolem T. *Über gewisse Satzfunktionen in der Arithmetik.* Skr. Norske Videnskaps-Akademi i Oslo, 7:154–180, 1930. Przedrukowano w Fenstad J.E. *Selected Works in Logic*, Universitetsforlaget, Oslo, str. 281–306, 1970.

[18] Smoryński C. *The incompleteness theorems.* W: Handbook of Mathematical Logic, str. 821–865, North Holland Publ. Comp., Amsterdam, 1977.

[19] Tarski A. *Pojęcie prawdy w językach nauk dedukcyjnych.* Nakładem Towarzystwa Naukowego Warszawskiego, Warszawa 1933.

[20] Zdanowski K. *Arithmetics in finite but potentially infinite worlds.* Rozprawa doktorska, Uniwersytet Warszawski, 2005.