

Seminarium: Programowanie w teorii typów

Teoria typów

Wojciech Jedynek, Paweł Wieczorek

Instytut Informatyki Uniwersytetu Wrocławskiego

25 września 2011

1 Matematyka konstruktywna

Matematyka konstruktywna

- powstały na początku poprzedniego wieku pogląd na temat fundamentów matematyki
- Brouwer, twórca ideologii
- empiryczna zawartość twierdzeń matematycznych
- co znaczy orzeczenie o istnieniu pewnego obiektu?
- odrzucenie dowodów przez sprowadzenie do sprzeczności
- odrzucenie idealistycznego podejścia do prawdziwości orzeczeń
- Bishop, konstruktywna analiza matematyczna

Książkowy przykład twierdzenia niekonstruktywnego

Twierdzenie

Istnieją takie dwie liczby niewymierne a oraz b , że a^b jest liczbą wymierną.

Dowód.

Orzeczenie, że $\sqrt{2}^{\sqrt{2}}$ musi być prawdziwe lub musi być fałszywe.

- jeżeli jest prawdziwe to mamy szukane a oraz b
- jeżeli jest fałszywe to niech $a = \sqrt{2}^{\sqrt{2}}$ oraz $b = \sqrt{2}$, wtedy $a^b = 2$



Jedyne co wiemy, to to że muszą istnieć takie liczby.

Kolejny przykład.

Twierdzenie (klasycznie)

Jeżeli funkcja f jest ciągła na przedziale $[0, 1]$ oraz wartości funkcji na krańcach przedziału mają różne znaki to istnieje punkt w tym przedziale na którym funkcja się zeruje.

Twierdzenie (konstruktywnie)

Jeżeli funkcja f jest ciągła na przedziale $[0, 1]$ oraz wartości funkcji na krańcach przedziału mają różne znaki to dla każdego $\epsilon > 0$ istnieje punkt w tym przedziale na którym bezwzględna wartość funkcji jest mniejsza od ϵ .

Interpretacja Brouwer-Heyting-Kołmogorow

- $A \wedge B$ to konstrukcja składająca się z dwóch pod-konstrukcji
- $A \vee B$ to konstrukcja składająca się z lewej lub prawej pod-konstrukcji
- $A \rightarrow B$ to metoda przekształcająca konstrukcję B mając do dyspozycji A
- \perp absurd, konstrukcja której nie można zrealizować
- $\forall x.P(x)$ to metoda przekształcająca wartość a w konstrukcję $P(a)$
- $\exists x.P(x)$ to konstrukcja mająca składać się ze świadka a oraz z konstrukcji $P(a)$
- $\neg A$ to skrót od $A \rightarrow \perp$
- Czy przy tej interpretacji wszystkie klasyczne prawa mają sens?
 - ▶ $\exists x P(x) \vee \neg \exists x P(x)$
 - ▶ $(\neg \forall x \neg P(x)) \rightarrow \exists x P(x)$

System naturalnej dedukcji

- system dowodzenia
- posługujemy się sędami $\Gamma \vdash \varphi$
- dowód to wyprowadzenie o strukturze drzewa
 - ▶ korzeń - wniosek (sąd)
 - ▶ węzeł - reguła wnioskowania
 - ▶ liść - aksjomat
- reguły wprowadzania i eliminacji spójników logicznych

System naturalnej dedukcji

$$I_{\wedge} \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \quad E_{\wedge 1} \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}$$

$$I_{\vee 1} \frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \quad E_{\vee} \frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C}$$

$$I_{\rightarrow} \frac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B} \quad E_{\rightarrow} \frac{\Gamma \vdash A \rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B}$$

$$AX \frac{}{\Gamma, A \vdash A}$$

2 Teoria typów Martina-Löfa

Historia, idee, początki

- Teoria typów jako logika matematyczna - B. Russel

$$A = \{w \mid w \notin w\}$$

- λ -rachunek , funkcja jako pojęcie pierwotne - A.Church
- System typów, likwidacja paradoksu Kleene'go

$$K = \lambda x. \neg(x\ x)$$

$$(K\ K) = \neg(K\ K) = \neg\neg(K\ K) = \dots$$

Wzbogacony system typów o więcej sądów

- Teoria typów Martina Lőf'a - system typów dla λ -rachunku, w którym możemy wydawać różne sądy:
 - ▶ A set - jest zbiorem
 - ▶ $a \in A$ - a jest elementem zbioru
 - ▶ $a =_A b \in A$ - a oraz b są równymi elementami w zbiorze A
 - ▶ $A = B$ - A oraz B są równymi zbiorami
- elementy zbiorów dzielimy na
 - ▶ kanoniczne - wartości (postać normalna)
 - ▶ niekanoniczne - obliczenia (nie znormalizowane)
- sformułowanie zbioru to
 - ▶ określenie kanonicznych elementów jakie ten zbiór zawiera
 - ▶ określenie co znaczy że dwa elementy są równe w tym zbiorze
 - ▶ określenie obliczeń

Przykład iloczynu kartezjańskiego w uproszczonej formie

$$\frac{A \text{ set} \quad B \text{ set}}{A \times B \text{ set}} \quad \frac{A = A' \quad B = B'}{A \times B = A' \times B'} \quad \frac{a \in A \quad b \in B}{(a, b) \in A \times B}$$

$$\frac{a = a' \in A \quad b = b' \in B}{(a, b) = (a', b') \in A \times B}$$

$$\frac{C \text{ set} \quad p \in A \times B \quad f(x, y) \in C [x \in A, y \in B]}{\text{split}(p, f) \in C}$$

Przykład funkcje w uproszczonej formie

$$\frac{A \text{ set} \quad B \text{ set}}{A \rightarrow B \text{ set}} \quad \frac{A = A' \quad B = B'}{A \rightarrow B = A' \rightarrow B'} \quad \frac{t(x) \in B [x \in A]}{\lambda t \in A \rightarrow B}$$

$$\frac{t(x) = t'(x) \in B [x \in A]}{\lambda t = \lambda t' \in A \rightarrow B} \quad \frac{f \in A \rightarrow B \quad a \in A}{f \cdot a \in B}$$

$$\frac{t(x) \in B [x \in A] \quad a \in A}{\lambda t \cdot a = t(a) \in B}$$

Izomorfizm Curry-Howard (proposition as types)

- Martin-Löf, Curry, Howard, deBruijn i wiele innych
- typy oznaczają formuły
- otypowane termy oznaczają dowody dla swoich typów (formuł)
- izomorfizm pomiędzy wyprowadzeniami formuł w logice intuicjonistycznej a sędami w systemie typów
- realizacja BHK

$$\begin{array}{c} \text{I} \rightarrow \frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x : A. t : A \rightarrow B} \quad \text{E} \rightarrow \frac{\Gamma \vdash f : A \rightarrow B \quad \Gamma \vdash x : A}{\Gamma \vdash f x : B} \end{array}$$

Izomorfizm Curry-Howard (proposition as types)

- fundamentalna teoria według kryteriów matematyki konstruktywnej
- pojęciem pierwotnym jest funkcja, nie zbiór
- nie używamy klasycznych definicji pojęć, mogą być one nie konstruktywne, nie dające się zrealizować
- funkcje które definiujemy są obliczalne i totalne
- teoria nie wyrażona jako logika pierwszego rzędu, lecz kodująca ją w sobie

Sądy mają więcej interpretacji

- A set - jest zbiorem
- A set - jest problemem, zagadnieniem, zadaniem
- A prop - jest formułą logiczną
- A true - umiemy zrealizować A , istnieje dowód A
- $a \in A$ - a jest elementem zbioru
- $a \in A$ - a jest dowodem propozycji A
- $a \in A$ - a jest programem spełniającym specyfikację A
- $a \in A$ - a jest rozwiązaniem problemu A