Seminarium: Programowanie w teorii typów

Teoria typów

Wojciech Jedynak, Paweł Wieczorek

Instytut Informatyki Uniwersytetu Wrocławskiego

3 października 2011

Plan

- Przypomnimy sobie ideę konstruktywizmu oraz izomorfizmu Curry'ego-Howard'a.
- Zapoznamy się z podstawami teorii typów Martina-Löf'a:
  - Zapoznamy się z sądami w tym systemie.
  - Sformułujemy podstawowe typy jak liczby naturalne czy *typy wyliczeniowe*.
  - Przebrniemy przez formalne definicje podstawowych typów zależnych: Π-typ, Σ-typ
  - Zobaczymy jak teoria typów koduje logikę pierwszego rzędu.
  - Zapoznamy się z równością propozycyjną (typem identycznościowym).

Matematyka konstruktywna

- powstały na początku poprzedniego wieku pogląd na temat fundamentów matematyki
- L.E.J.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
- E. 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 $\sqrt{2}^{\sqrt{2}} \in \mathbf{Q}$ 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 $a^b = 2$

$\square$

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?
  - $\exists x\ P(x) \vee \neg \exists x\ P(x)$
  - $\exists x\ P(x) \equiv \neg(\forall x \neg P(x))$
  - $A \vee B \equiv \neg(\neg A \wedge \neg B)$

System naturalnej dedukcji

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

System naturalnej dedukcji

$$\mathrm{I}\wedge\,\frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \qquad \mathrm{E}\wedge_1\,\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}$$

$$\mathrm{I}\vee_1\,\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \qquad \mathrm{E}\vee\,\frac{\Gamma \vdash A \vee B \qquad \Gamma, A \vdash C \qquad \Gamma, B \vdash C}{\Gamma \vdash C}$$

$$\mathrm{I}\!\to\,\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B} \qquad \mathrm{E}\!\to\,\frac{\Gamma \vdash A \to B \qquad \Gamma \vdash A}{\Gamma \vdash B}$$

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

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 sądami w systemie typów
- realizacja BHK

$$\mathrm{I}{\to}\,\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x : A.t : A \to B} \qquad \mathrm{E}{\to}\,\frac{\Gamma \vdash f : A \to B \qquad \Gamma \vdash x : A}{\Gamma \vdash f\ x : B}$$

System typów dla λ-rachunku

$$\text{I}\wedge\,\frac{\Gamma \vdash M : A \qquad \Gamma \vdash N : B}{\Gamma \vdash (M, N) : A \wedge B} \qquad \text{E}\wedge_1\,\frac{\Gamma \vdash M : A \wedge B}{\Gamma \vdash \mathit{fst}\ M : A}$$

$$\text{I}\vee_1\,\frac{\Gamma \vdash M : A}{\Gamma \vdash \mathit{inl}\ M : A \vee B}$$

$$\text{E}\vee\,\frac{\Gamma \vdash M : A \vee B \qquad \Gamma, x : A \vdash P : C \qquad \Gamma, x : B \vdash Q : C}{\Gamma \vdash \mathit{when}\ M\ (\lambda x.P)\ (\lambda x.Q) : C}$$

$$\text{I}\!\rightarrow\,\frac{\Gamma, A \vdash M : B}{\Gamma \vdash \lambda x.M : A \rightarrow B} \qquad \text{E}\!\rightarrow\,\frac{\Gamma \vdash M : A \rightarrow B \qquad \Gamma \vdash N : A}{\Gamma \vdash M\ N : B}$$

$$\text{AX}\,\frac{}{\Gamma, x : A \vdash x : A}$$

Teoria typów: 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

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

$$(K\ K) = \neg(K\ K) = \neg\neg(K\ K) = \cdots$$

Teoria typów Martina-Löf'a

- Per Martin-Löf, A Theory of Types, 1972
- Per Martin-Löf, Constructive Mathematics and Computer Programming, 1979
- Per Martin-Löf, **Intuitionistic Type Theory**, 1980
- Per Martin-Löf, Truth of a Proposition, Evidence of a Judgement, Validity of a Proof, 1985
- B. Nordström, K. Petersson, and Jan M. Smith, **Programming in Martin-Löf's Type Theory: An Introduction**, 1990

Wyrażenia jakimi się posługujemy

- język wyrażeń:
  - $e(e')$ - aplikacja ($e(e_1, \cdots, e_n)$ skrócony zapis $e(e_1)\cdots(e_n)$)
  - $(x)e$ - abstrakcja
  - stałe, np λ, Π, $apply, 0, succ$
- wyrażeniom przypisujemy arność:
  - chcemy to napominać?

Wyrażenia jakimi się posługujemy

- Pozwalamy na definiowanie nowych stałych jako makra, np

$$double \equiv (n)(n+n)$$

$$double(n) \equiv n+n$$

- Równość definicyjna
  - relacja równoważności, kongruencja
  - surowa równość na wyrażeniach
  - nic nie mówi o znaczeniu
  - elementy równe definicyjne traktujemy jako synonimy

$$\begin{array}{rcll} ((x)b)(x) & \equiv & b & x \text{ nie występuje w } b \\ ((x)b)(a) & \equiv & b[x := a] & \\ ((x)b) & \equiv & (y)(b[x := y]) & \end{array}$$

Wzbogacony system typów o więcej sądów

- Teoria typów Martina Löf'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
- sformułowanie zbioru to
  - określenie kanonicznych elementów jakie ten zbiór zawiera
  - określenie co znaczy że dwa elementy są równe w tym zbiorze
  - określenie obliczeń

Sądy mają więcej interpretacji

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

Reguły wnioskowania

- reguły formułowania
- reguły wprowadzania
- reguły eliminacji
- reguły równościowe

$$\frac{A\ set \qquad B(x) set\ [x \in A]}{(\Pi x \in A)B(x)\ set} \qquad \frac{A\ set\ [\Gamma] \qquad B(x) set\ [x \in A, \Delta]}{(\Pi x \in A)B(x)\ set[\Gamma, \Delta]}$$

$$\frac{b(x) \in B\ [x \in A]}{\lambda b \in A \rightarrow B}$$

$$\frac{A\ set\ [\Gamma] \qquad B\ set\ [\Delta] \quad b(x) \in B\ [\Theta, x \in A]}{\lambda b \in A \rightarrow B\ [\Gamma, \Delta, \Theta]}$$

Zbiór liczb naturalnych

$$\frac{}{Nat\ set} \qquad \frac{}{0 \in Nat} \qquad \frac{n \in Nat}{succ(n) \in Nat}$$

$$\frac{n = n' \in Nat}{succ(n) = succ(n') \in Nat}$$

$$\frac{a \in N \qquad C(v)\ set\ [v \in Nat] \qquad d \in C(0) \qquad e(x,y) \in C(succ(x))\ [x \in Nat, y \in C(x)]}{natrec(a,d,e) \in C(a)}$$

$$\frac{C(v)\ set\ [v \in Nat] \qquad C(0) \qquad e(x,y) \in C(succ(x))\ [x \in Nat, y \in C(x)]}{natrec(0,d,e) = d \in C(a)}$$

$$\frac{a \in N \qquad C(v)\ set\ [v \in Nat] \qquad d \in C(0) \qquad e(x,y) \in C(succ(x))\ [x \in Nat, y \in C(x)]}{natrec(succ(a),d,e) = e(a, natrec(a,d,e)) \in C(succ(a))}$$

Zbiór liczb naturalnych

$$\frac{a \in N \qquad C(v)\ set\ [v \in Nat] \qquad d \in C(0) \qquad e(x,y) \in C(succ(x))\ [x \in Nat, y \in C(x)]}{natrec(a,d,e) \in C(a)}$$

$$\frac{a \in N \qquad C(v)\ prop\ [v \in Nat] \qquad C(0)\ true \qquad C(succ(x))\ true\ [x \in Nat, C(x)\ true]}{C(a)\ true}$$

Zbiór liczb naturalnych

- przykłady

Produkt indeksowanej rodziny zbiorów (w matematyce)

- matematyczna definicja

$$\prod_{x \in A} B_x = \left\{ f : A \to \bigcup_{x \in A} B_x \mid \forall x \in A.\ f(x) \in B_x \right\}$$

- zależność przeciwdziedziny od argumentu

$$\text{sort} \in \prod_{xs \in Lists} \{ ys \in Lists \mid perm(ys, xs) \wedge sorted(ys) \}$$

$$perm(sort(xs_0), xs_0) \wedge sorted(sort(xs_0))$$

- możemy też wyrazić „zwykły” zbiór funkcji, jeżeli $B_x = B$ to

$$\prod_{x \in A} B_x = A \to B$$

Produkt indeksowanej rodziny zbiorów

$$\frac{A \ set \qquad B(x) \ set \ [x \in A]}{(\Pi x \in A)B(x) \ set} \qquad \frac{b(x) \in B(x) \ [x \in A]}{\lambda x.b \in (\Pi x \in A)B(x) \ set}$$

$$\frac{A = A' \qquad B(x) = B'(x) \ [x \in A]}{(\Pi x \in A)B(x) = (\Pi x \in A')B'(x)}$$

$$\frac{b(x) = b'(x) \in B(x) \ [x \in A]}{\lambda x.b = \lambda x.b' \in (\Pi x \in A)B(x) \ set}$$

Produkt indeksowanej rodziny zbiorów

$$\frac{f \in (\Pi x \in A)B(x) \qquad a \in A}{apply(f, a) \in B(a)} \qquad \frac{b(x) \in B(x)\ [x \in A] \qquad a \in A}{apply(\lambda x.b, a) = b(a) \in B(a)}$$

$$\frac{a = a' \in A \qquad f = f' \in (\Pi x \in A)B(x)}{apply(f, a) = apply(f', a') \in B(a)}$$

Produkt indeksowanej rodziny zbiorów

$$(\forall x \in A)B(x) \equiv (\Pi x \in A)B(x)$$

$$\frac{A\ set \qquad B(x)\ set\ [x \in A]}{(\Pi x \in A)B(x)\ set} \quad \Rightarrow \quad \frac{A\ set \qquad B(x)\ prop\ [x \in A]}{(\forall x \in A)B(x)\ prop}$$

$$\frac{b(x) \in B(x)\ [x \in A]}{\lambda x.b \in (\Pi x \in A)B(x)\ set} \quad \Rightarrow \quad \frac{B(x)\ true\ [x \in A]}{(\forall x \in A)B(x)\ true}$$

$$\frac{f \in (\Pi x \in A)B(x) \qquad a \in A}{apply(f, a) \in B(a)} \quad \Rightarrow \quad \frac{(\forall x \in A)B(x)\ true \qquad a \in A}{B(a)\ true}$$

Produkt indeksowanej rodziny zbiorów

- Załóżmy że $f \in (\Pi X \in A)(\Pi y \in B)P(x,y)$, zaprogramujmy *flip f*

$$\underbrace{\frac{f \in (\Pi X \in A)(\Pi y \in B)P(x,y) \qquad x \in A\ [x \in A]}{apply(f,x) \in (\Pi y \in B)P(x,y)\ [x \in A]}}_{D}$$

$$\cfrac{\cfrac{\cfrac{\overbrace{apply(f,x) \in (\Pi y \in B)P(x,y)\ [x \in A]}^{D} \qquad y \in B\ [y \in B]}{apply(apply(f,x),y) \in P(x,y)\ [y \in B, x \in A]}}{\lambda x.\ apply(apply(f,x),y) \in (\Pi x \in B)P(x,y)\ [y \in B]}}{\lambda y.\lambda x.\ apply(apply(f,x),y) \in (\Pi y \in B)(\Pi x \in A)P(x,y)}$$

$$\frac{(\forall x \in A)(\forall y \in B)P(x,y)\ true}{(\forall y \in B)(\forall x \in A)P(x,y)\ true}$$

Produkt indeksowanej rodziny zbiorów

- Jeżeli x nie występuje w B to

$$A \to B \equiv (\Pi x \in A)B$$

$$\frac{A\ set \qquad B\ set\ [x \in A]}{A \to B\ set} \qquad \frac{b(x) \in B\ [x \in A]}{\lambda x.b \in A \to B}$$

$$\frac{A\ prop \qquad B\ prop\ [A\ true]}{A \to B\ prop} \qquad \frac{B\ true\ [A\ true]}{A \to B\ true}$$

Suma rozłączna indeksowanej rodziny zbiorów

$$\frac{A \ set \qquad B(x) \ set \ [x \in A]}{(\Sigma x \in A)B(x) \ set}$$

$$\frac{a \in A \qquad B(x) \ set \ [x \in A] \qquad p \in B(a)}{\langle a, p \rangle \in (\Sigma x \in A)B(x)}$$

$$\frac{A = A' \qquad B(x) = B'(x) \ [x \in A]}{(\Sigma x \in A)B(x) = (\Sigma x \in A')B'(x)}$$

$$\frac{a = a' \in A \qquad b = b' \in B(a)}{\langle a, b \rangle = \langle a'.b' \rangle \in (\Sigma x \in A)B(x) \ set}$$

Suma rozłączna indeksowanej rodziny zbiorów

$$\frac{c \in (\Sigma x \in A)B(x) \qquad C(v)\ set\ [v \in (\Sigma x \in A)B(x)] \quad d(x,y) \in C(\langle x,y\rangle)\ [x \in A, y \in B(a)]}{split(c,d) \in C(c)}$$

$$\frac{a \in A \qquad b \in B(a) \quad C(v)\ set\ [v \in (\Sigma x \in A)B(x)] \quad d(x,y) \in C(\langle x,y\rangle)\ [x \in A, y \in B(a)]}{split(\langle a,b\rangle, d) = d(a,b) \in C(\langle a,b\rangle)}$$

$$\frac{c = c' \in (\Sigma x \in A)B(x) \qquad C(v)\ set\ [v \in (\Sigma x \in A)B(x)] \quad d(x,y) = d'(x,y) \in C(\langle x,y\rangle)\ [x \in A, y \in B(a)]}{split(c,d) = split(c',d') \in C(c)}$$

Suma rozłączna indeksowanej rodziny zbiorów

$$(\exists x \in A)B(x) \equiv (\Sigma x \in A)B(x)$$

$$\frac{A\ set \qquad B(x)\ set\ [x \in A]}{(\Sigma x \in A)B(x)\ set} \quad \Rightarrow \quad \frac{A\ set \qquad B(x)\ prop\ [x \in A]}{(\exists x \in A)B(x)\ prop}$$

$$\frac{a \in A \qquad p \in B(a)}{\langle a, p \rangle \in (\Sigma x \in A)B(x)} \quad \Rightarrow \quad \frac{a \in A \qquad B(a)\ true}{(\exists x \in A)B(x)\ true}$$

$$\frac{\begin{array}{c} c \in (\Sigma x \in A)B(x) \qquad C(v)\ set\ [v \in (\Sigma x \in A)B(x)] \\ d(x,y) \in C(\langle x, y \rangle)\ [x \in A, y \in B(a)] \end{array}}{split(c,d) \in C(c)}$$

$$\Downarrow$$

$$\frac{(\exists x \in A)B(x)\ true \qquad C\ prop \qquad C\ true\ [x \in A, B(a)\ true]}{C\ true}$$

Suma rozłączna indeksowanej rodziny zbiorów

- Jeżeli x nie występuje w B to

$$A \times B \equiv A \wedge B \equiv (\Sigma x \in A)B$$

$$\frac{A \ set \qquad B \ set \ [x \in A]}{A \times B \ set} \qquad \frac{a \in A \qquad b \in B}{\langle a, b \rangle \in A \times B}$$

$$\frac{A \ prop \qquad B \ prop \ [A \ true]}{A \wedge B \ prop} \qquad \frac{A \ true \qquad B \ true}{A \wedge B \ true}$$

Równość propozycyjna

- Nie mamy odpowiednika atomowej formuły $a = b$
- $a = b \in C$ to sąd, nie propozycja.
- Czego oczekujemy od równości propozycyjnej?
  - By był zamieszkany wtedy i tylko wtedy gdy formuła atomowa $a = b$ jest prawdziwa.
  - Abyśmy mogli prowadzić wnioskowanie bazujące na równościach.

Równość propozycyjna

$$\frac{A\ set \qquad a \in A \qquad b \in A}{[a =_A b]\ set}$$

$$\frac{a \in A}{id(a) \in [a =_A a]}$$

$$\frac{A = A' \qquad a = a' \in A \qquad b = b' \in A}{[a =_A b] = [a' =_{A'} b']}$$

$$\frac{a = a' \in A}{id(a) = id(a') \in [a =_A a]}$$

Równość propozycyjna

$$\frac{a \in A \qquad b \in A \qquad c \in [a =_A b] \quad C(x,y,z)\ set\ [x \in A, y \in A, z \in [x =_A y] \quad d(x) \in C(x,x,id(x))\ [x \in A]}{idpeel(c,d) \in C(a,b,c)}$$

$$\frac{a \in A \quad C(x,y,z)\ set\ [x \in A, y \in A, z \in [x =_A y] \quad d(x) \in C(x,x,id(x))\ [x \in A]}{idpeel(id(a),d) = d(a) \in C(a,a,id(a))}$$

- Zbyt silna eliminacja:

$$\frac{[a =_A b]\ true}{a = b \in A}$$