Kapitola 6

Státnice - Algoritmicky vyčíslitelné funkce

6.1 Částečně rekurzivní funkce

K. Gödel v 30. letech vynalezl primitivní rekurzivní funkce, později společně s dalšími částečně rekurzivní funkce. Jde o funkcionální přístup k algoritmům. Lze se na ně dívat i jako na logiku 1. řádu: základní funkce jsou axiomy, máme operátory – odvozovací pravidla – a z toho vyrábíme formule – rekurzivní funkce.

Definice (Podmíněná rovnost, konvergence, divergence)

- $\simeq$ značí "podmíněnou rovnost", tj. v případě, že alespoň jedna strana má smysl, tak má smysl i druhá a rovnají se.
- $P_1(D)\downarrow$ značí, že predikát je definován, tj "konverguje" (občas se značí ! místo $\downarrow$)
- $P_1(D)\uparrow$ značí, že predikát není definován, tj. "diverguje"

Značky konvergence, divergence i podmíněné rovnosti se vztahují jak na predikáty, tak na funkce.

Definice (Základní funkce)

- $o(x) \simeq 0 \quad \forall x \in \mathbb{N}$ ("nula")
- $s(x) \simeq x + 1 \quad \forall x \in \mathbb{N}$ ("následník")
- $I_n^j(x_1, \ldots, x_n) \simeq x_j \quad 1 \leq j \leq n$ ("projekce", vybrání jedne ze složek)

Definice (Základní operátory)

- $R_n \quad (n \geq 1)$ – primitivní rekurze
  Funkcím f ($n-1$ proměnných) a g ($n+1$ proměnných) přiřadí $R_n(f,g) = h$, kde $h(0, x_2, \ldots, x_n) \simeq f(x_2, \ldots, x_n)$ a $h(y+1, x_2, \ldots, x_n) \simeq g(y, h(y, x_2, \ldots, x_n), x_2, \ldots, x_n)$ (analogické k for-cyklu).
- S_n^m – substituce
  Funkci f (m proměnných) a m funkcím g_i (všechny n proměnných) přiřadí funkci h (n proměnných) předpisem $h = S_n^m(f, g_1, \ldots, g_m) \equiv h(x_1, \ldots, x_n) \simeq f(g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n))$ (analogické k podprogramu).
- M_n – minimalizace
  Funkci f ($n+1$ proměnných) přiřadí h (n proměnných) tak, že

  $$h(x_1, \ldots, x_n)\downarrow \wedge h(x_1, \ldots, x_n) \simeq y \equiv f(x_1, \ldots, x_n, y)\downarrow, \simeq 0 \wedge f(x_1, \ldots, x_n, j)\downarrow, \not\simeq 0 \ \forall j < y$$

  (analogické k while-cyklu).

Další značení:

- $\mu_y P(x,y)$ je funkce proměnné x, která vrátí nejmenší y takové, aby platil predikát $P(x,y)$. Lze jí sestrojit pomocí operátoru minimalizace.

Definice (Třída primitivně a částečně rekurzivních funkcí)

- Třída primitivně rekurzivních funkcí je nejmenší třída funkcí $f : \mathbb{N}^k \to \mathbb{N}$, která obsahuje základní funkce a je uzavřená na R_n a S_n^m.
- Třída částečně rekurzivních funkcí je nejmenší třída, která obsahuje zákl. funkce a je uzavřená na R_n, S_n^m a M_n.

Poznámka (Vlastnosti zákl. funkcí a operátorů)

- Všechny zákl. funkce jsou všude definované ("totální") a efektivně vyčíslitelné.
- Všechny zákl. operátory zachovávají efektivní vyčíslitelnost.
- R_n, S_n^m zachovávají totálnost.
- PRF jsou efektivně vyčíslitelné a totální.

Definice (Odvození funkce)

Odvození funkce je konečná posloupnost funkcí, z nichž každá je buď funkce základní, nebo vzniká z už odvozených funkcí pomocí nějakého operátoru. Ke každé funkci si pamatujeme, jak vznikla (toto v praxi hraje roli programu).

Definice (Obecně rekurzivní funkce)

Funkce je obecně rekurzivní (ORF), jestliže je ČRF a totální.

Operace s PRF, predikáty

Poznámka (Některé PRF)

Pomocí PRF lze popsat např.:

- součet
- součin
- mocninu, faktoriál
- operaci $x \dot{-} y$, kde $x \dot{-} y = x - y$ pro $x \geq y$, jinak 0
- operátory sg a $\overline{\text{sg}}$ (testy na nenulovost, resp. nulovost argumentu)
- minimum, maximum, absolutní hodnotu rozdílu

Definice (Charakteristická funkce)

Mějme predikát P (libovolné tvrzení) o n proměnných. Potom c_P je jeho charakteristická funkce, když je to všude definovaná funkce daná následovně:

$$c_P(x_1, \ldots, x_n) \simeq \begin{cases} 1 & \text{pokud } P(x_1, \ldots, x_n) \\ 0 & \text{jinak} \end{cases}$$

Částečná charakteristická funkce pro nějaký predikát P o n proměnných je funkce f o n proměnných taková, že $f(x_1, \ldots, x_n) \downarrow \Leftrightarrow P(x_1, \ldots, x_n)$ a $f(x_1, \ldots, x_n) \downarrow \Rightarrow f(x_1, \ldots, x_n) = 1$.

Definice (PR, OR, RS Predikáty)

Řekneme, že predikát je primitivně (obecně) rekurzivní, jestliže jeho charakteristická funkce je primitivně (obecně) rekurzivní. Predikát je rekurzivně spočetný, jestliže jeho částečná charakteristická funkce je částečně rekurzivní.

S funkcemi a predikáty se operuje docela nedůsledně, dají se v podstatě ztotožnit.

Poznámka (Jiná možnost nahlížení)

ČRF odpovídají funkcionální logice 1. řádu:

- termy číselné: $0, x, x+1, \ldots$
- termy funkční: $o, I_1^1, s, R_2(I_1^1, S_3^1(s, I_3^2)), \ldots$
- pravidlo aplikace: $Ap(f, x) = \cdots = f(x)$ (kde "..." je proces vyhodnocení termu, potenciálně nekonečný, dává z funkce číselný term)
- pravidlo zobecnění: $\lambda xy(x+y)$ dává z číselného termu $x+y$ funkci

Poznámka (Operace zachovávající PR)

PR jsou:

- Rozšíření počtu proměnných, konstantní funkce
- Permutace a ztotožnění proměnných
- Kódování $\mathbb{N}^k$ do $\mathbb{N}$ – iterace Cantorova diagonálního kódování dvojic ($\langle x, y\rangle_2 = \frac{(x+y)(x+y+1)}{2} + x$)
- Opačná operace – dekódování
- Funkce $p(i)$ – i-té prvočíslo
- Predikát rovnosti a $<, >$
- Logické spojky $\vee, \wedge, \neg$, omezené kvantifikátory (kvantifikace spočetně mnoha prvků)
- Gödelovo prvočíselné kódování: slovo $a_{i_0} \dots a_{i_k}$ do $p(0)^{i_0} \cdot \dots \cdot p(k)^{i_k}$

Ackermannova funkce

Definice (Ackermannova funkce)

Ackermannova funkce je funkce definovaná jako:

$$A(0,x) = \begin{cases} 1 & x = 0 \\ 2 & x = 1 \\ x+2 & x > 1 \end{cases}$$

$$A(y,0) = 1$$

$$A(y+1, x+1) = A(y, A(y+1, x))$$

Definice (Strukturální složitost)

Definujeme strukturální složitost – hloubku rekurze (intuitivně: počet vnořených for-cyklů – syntakticky, ne výpočtem) jako 0 pro základní funkce a

$$h(R_n(P,Q)) = \max(h(P), h(Q)+1),\ h(S_n^m(P, Q_0, \dots, Q_k)) = \max(h(P), h(Q_0), \dots, h(Q_k))$$

Pak $\mathcal{R}_i$ je třída PRF, které lze získat pomocí PR-termů hloubky $\leq i$ a PRF samo je $\cup_{i=1}^{\infty} \mathcal{R}_i$

Věta (O Ackermannově funkci)

Ackermannova funkce není PRF, ale je ORF.

Důkaz

- Určitě je ORF – důkaz se provádí transfinitní indukcí typu ω^2; pro výpočet každé hodnoty potřebuji jen konečně mnoho předchozích hodnot – stačí mi μ_z, kde z je nejmenší kus $\mathbb{N}^2$, který stačí k výpočtu $A(y,x)$ (dá práci dokázat, že je konečný, potřeba ordinálů, lexikografického uspořádání).
- A roste rychleji než každá PRF: $\forall \varphi$ PRF (jedné proměnné) $\exists x_0 : \forall x \geq x_0 : \varphi(x) < A(x,x)$.
- Uvažujme $A(y,x)$ jako matici funkcí $f_y(x)$. Potom určitě $f_i \in \mathcal{R}_i \backslash \mathcal{R}_{i-1}$ a $f_y(x)$ je (až na konečně mnoho x) rostoucí. Navíc pro libovolnou $\varphi \in \mathcal{R}_i$ existuje x_0 takové, že $\forall x \geq x_0 : \varphi(x) < f_{i+1}(x)$, tedy f_{i+1} majorizuje všechny funkce z $\mathcal{R}_i$
- Nechť pro spor má $A(x,x)$ hloubku i. Potom $A(x,x) = \varphi(x) < f_{i+1}(x)$ pro nějaké pevné i. Potom ale $A(x,x) < f_{i+1}(x) < f_x(x) = A(x,x)$, tj. pro $x > i+1$ máme spor.

Věta (O vztahu PRF, ORF a ČRF)

Platí PRF$\subset$ ORF $\subset$ ČRF a inkluze jsou ostré.

Důkaz

Pro ORF$\subset$ ČRF mám funkci $g(x,y) \simeq y+1$ a $h(x) \simeq \mu_y(g(x,y) \simeq 0)$, ta není nikde definovaná. Pro PRF$\subset$ ORF mám Ackermannovu funkci.

6.2 Univerzální funkce

Definice (Univerzální funkce)

Mějme $\mathcal{T}$ – spočetnou množinu ČRF jedné proměnné. Potom $\mathcal{U}(i,x)$ je univerzální funkce třídy $\mathcal{T}$, jestliže:

- $\forall$ přirozené $i : \lambda x\mathcal{U}(i,x) \in \mathcal{T}$
- $\forall \varphi \in \mathcal{T} : \exists i_0 : \varphi = \lambda x\mathcal{U}(i_0,x)$

A $\mathcal{U}$ tedy indexuje všechny funkce třídy $\mathcal{T}$. Podobně se definují i univerzální funkce pro ČRF více proměnných. Platí, že $\{\lambda x\mathcal{U}(i,x)\}_{y\geq 0}$ je posloupnost všech funkcí z $\mathcal{T}$, takže $\mathcal{U}$ určuje numeraci prvků $\mathcal{T}$

Věta (O univerzální funkci PRF)

Existuje ORF, která je univerzální pro třídu PRF (jedné proměnné). Taková funkce pak nemůže být PRF.

Důkaz

- Seřadím všechny PR-termy (PR-programy) do posloupnosti (máme 3 axiomy a 3 odvozovací pravidla, seřazení je možné).
- Potom $U(x,y) := h_x(y)$, kde h_x vyčísluje x-tý program.
- Sporem nechť $U(x,y)$ je PRF. Pak i $U(x,x)$ je PRF, $1\dot{-}U(x,x)$ je PRF, z toho $1\dot{-}U(x,x) = U(x_0,x)$. Dosadím $x = x_0$ a mám spor, neboť obě strany jsou definovány (toto je příklad použití Cantorovy diagonální metody).

Definice (Turingovsky vyčíslitelná funkce)

Vezmeme Turingovy stroje s vnější abecedou, jejíž prvním znakem je "|". Čísla $0, 1, \ldots$ zapisujme na pásku jako $|, ||, |||, \ldots$, n-tice oddělujme znakem λ. Potom:

- Řekneme, že stroj M je n-aritmetický, pokud pro každou n-tici přir. čísel $x_1, \ldots, x_n$ reprezentovanou počáteční konfigurací S platí: je-li M použitelný k S (zastaví-li se výpočet nad ní) a je-li výsledná konfigurace T, pak v T je na pásce nějaké jedno přirozené číslo a hlava stroje M stojí nad jeho posledním znakem |.
- Stroj je dále n-aritmetický typu 0/1, pokud má abecedu $\{|, \lambda\}$ a jediný koncový stav.
- Řekneme, že M vyčísluje funkci f o n proměnných, pokud M je n-aritmetický a pro každou n-tici přir. čísel $x_1, \ldots, x_n$ v poč. konfiguraci S platí: M je použitelný, právě když je f pro $x_1, \ldots, x_n$ definovaná a je-li f definovaná, pak ve výsledné konfiguraci T je na pásce stroje číslo $f(x_1, \ldots, x_n)$ a hlava stojí nad jeho posledním znakem.
- Řekneme, že funkce je turingovsky vyčíslitelná, pokud existuje nějaký n-aritmetický TS (typu 0/1), který ji vyčísluje.

Věta (O ekvivalenci TS a ČRF)

Funkce n proměnných je částečně rekurzivní, právě když existuje n-aritmetický Turingův stroj typu 0/1, který ji vyčísluje.

Důkaz

"$\Leftarrow$": Každá ČRF je T-vyčíslitelná.

Důkaz indukcí podle složitosti funkce – pro základní funkce to jistě platí, R_n, M_n, S_n^m toto zachovávají (S_n^m znamená použití více pásek, vyčíslení a složení, R_n znamená vyčíslení f a y-krát "otočení" g, M_n je vyčíslování f na vstupu a zvětšujícím se počítadle cyklů, dokud nedostanu 0 – pak vrátím hodnotu počítadla).

"$\Rightarrow$": Pro každý TS M existuje ČRF, která dává stejný výsledek

Je nutné zavést kódy konfigurací; TS navíc nemají žádné podtřídy, tedy nelze postupovat induktivně. Platí:

- $\text{step}_M(X)$ – jeden krok stroje je PR záležitost (pracuje se nad konfiguracemi TS $UqsV$, z obou stran obalenými spec. znakem h, pak slovo není nekonečné a lok. změna se dá spočítat). Existuje určitě PR funkce, která popisuje lokální změnu (jde vlastně o rozhodovací strom, který se staví pomocí R_n).
- $\text{comp}_M(X,i)$ – výsledek stroje po i krocích práce je stále PR (for-cyklus – R_n)
- $\mu_i(\text{comp}_M(X,i)$ obsahuje $q_0)$ – q_0 je koncový stav (pracuj, dokud neskončíš – while)
- Potom výsledná ČRF g je dána jako $g(\text{kód}(S)) \simeq \text{result}(\mu_i(\text{comp}_M(X,i)$ obsahuje $q_0))$, kde result je jednoduchá funkce smazání okrajů atp. BÚNO je takový stroj úplný a q_0 jeho jediný koncový stav. Operátor minimalizace se vyskytuje jen jednou, proto je vhodné ho vysunout co nejvíce "ven" v uzávorkování.

Pak také platí, že mám-li nějakou částečnou funkci (tj. nemusí být totální), která je turingovsky vyčíslitelná, pak je ČR.

Kleenova věta

Věta (Kleenova o normální formě)

Pro každé $k \geq 1$ existují

- ČRF Ψ_k $k+1$ proměnných
- PRP T_k $k+2$ proměnných (Kleeneův predikát)
- PRF U jedné proměnné
- PRF s_k $k+1$ proměnných

takové, že:

1. Ψ_k je univerzální funkcí pro třídu všech ČRF k proměnných. $\Psi_k(e, x_1, \ldots, x_k)$ vyčísluje e-tou ČRF k proměnných. Navíc z odvození ČRF lze efektivně získat e a naopak z e lze efektivně získat odvození příslušné ČRF.
2. $\Psi_k(e, x_1, \ldots, x_k) \simeq U(\mu_y T_k(e, x_1, \ldots, x_k, y))$, kde T_k odpovídá výpočtu Turingova stroje, $y = \langle y_0, y_1 \rangle$, y_0 je doba výpočtu, y_1 výsledek a U vydělí z $\langle y_0, y_1 \rangle$ druhou složku.
3. s_k je prostá funkce rostoucí ve všech proměnných, o které platí (tato část Věty o normální formě se nazývá S-m-n věta):
   $\Psi_{m+n}(e, z_1, \ldots, z_m, x_1, \ldots, x_n) \simeq \Psi_n(s_m(e, z_1, \ldots, z_m), x_1, \ldots, x_n)$
   $T_{m+n}(e, \vec{z}, \vec{x}) \equiv T_n(s_m(e, \vec{z}), \vec{x})$
4. $T_k(e, x_1, \ldots, x_k, y) \wedge T_k(e, x_1, \ldots, x_k, z) \Rightarrow y = z$

Díky tomu lze ČRF efektivně očíslovat. $\varphi_e(x_1, \ldots, x_k)$ pak značí e-tou funkci k proměnných. Indexu e se říká Gödelovo číslo funkce.

Důkaz

- Oklikou přes Univerzální Turingův stroj: ke každé ČRF máme TS a jeho kód e. Vezmeme si proto UTS, který s kódy umí počítat, a hledáme jeho ČRF.
- Páska univerzálního stroje vypadá v obecném případě následovně:

  $$Y \text{ blok1 } Y \text{ blok2 } \Delta \text{ blok3 } \times O_1 \times O_2 \ldots Y$$

  První blok je aktuální konfigurace, druhý číslo stavu a třetí aktuální políčko, zbytek je program. Čísla kódujeme unárně (x jako $x+1$ čar).
- Základní idea – bez proměnných $x_1, \ldots x_k$ páska UTS vypadá takto: $Y\ M\ Y$ blok2 Δ blok3 $\times O_1 \times O_2 \ldots Y$ (M je kód programu).
- Konstrukce $\Psi_m(e, x_1, \ldots, x_m)$:
  - Zkontrolujeme, zda e po rozkódování obsahuje nějaký kód programu M.
  - Jestliže ne, je výsledkem nulová funkce (syntax error).
  - Jestliže ano, nejlevější výskyt M nahradíme $||\ldots|\lambda||\ldots|\lambda\ldots\lambda||\ldots|M$ (kódování vstupních dat $x_1, \ldots, x_n$; substituce) a spustíme program e na UTS, podle toho získáme výsledek – Ψ_k
- $s_k(e, y_1, \ldots, y_k)$ odpovídá: čekej na $x_1, \ldots, x_j$, přidej k nim $y_1, \ldots, y_k$ a spusť program e.

Věta (Vlastnosti predikátu Ψ_k)

1. Predikát $\Psi_k(e, x_1, \ldots, x_k)\downarrow$ je rekurzivně spočetný, není rekurzivní.
2. jeho negace $\Psi_k(e, x_1, \ldots, x_k)\uparrow$ není rekurzivně spočetná.
3. Dále Ψ_k nelze rozšířit do ORF. Dokonce pokud α je ČRF, která je rozšířením Ψ_k, potom lze efektivně nalézt vstup $\vec{z}$ takový, že $\alpha(\vec{z})\uparrow$.

Univerzální funkce pro danou třídu funkcí tedy buď nemůže patřit do této třídy, nebo nemůže být totální.

Důkaz

- Z definice je zřejmé, že $\Psi_k(\ldots)\downarrow$ je rekurzivně spočetný predikát. Stačí ukázat, že $\Psi_k(\ldots)\uparrow$ není rekurzivně spočetný. Z toho přímo plyne, že $\Psi_k(\ldots)\downarrow$ není rekurzivní.
- Bez újmy na obecnosti uvažujme k=1. Použijeme Cantorovu diagonální metodu.
- Kdyby $\Psi_1(\ldots)\downarrow$ byl rekurzivní, potom by $\Psi_1(x,x)\uparrow$ byl také rekurzivní, tím spíše rekurzivně spočetný. Tedy pro nějakou ČRF φ by platilo $\Psi_1(x,x)\uparrow\Leftrightarrow \varphi(x)\downarrow$. Vezmeme-li index funkce φ (označme jej x_0), dostáváme $\Psi_1(x,x)\uparrow\Leftrightarrow \Psi_1(x_0,x)\downarrow$, po dosazení $x = x_0$ dostáváme $\Psi_1(x_0,x_0)\uparrow\Leftrightarrow \Psi_1(x_0,x_0)\downarrow$, což je spor.
- Pro důkaz zbytku tvrzení předpokládejme, že $h(e,x)$ je ORF rozšířením $\Psi_1(e,x)$. Potom $1\dot{-}h(x,x)$ je ORF g. Nechť g má index x_0, tj. $g(x) \simeq \Psi_1(x_0,x)$. Protože g je ORF, pro všechna x platí $\Psi_1(x_0,x)\downarrow$, tedy $\Psi_1(x_0,x_0)\downarrow$. Dostáváme $h(x_0,x_0) = \Psi_1(x_0,x_0)$, což ovšem vede ke sporu: $1\dot{-}\Psi_1(x_0,x_0) \simeq h(x_0,x_0) \simeq \Psi_1(x_0,x_0)$.
- Pokud nějaká ČRF β je rozšířením Ψ_1, umím pro β (podle předch. důkazu) najít e takové, že $\beta(e,e)\uparrow$.
- Myšlenka obsažená v předchozím důkazu je založená na Cantorově diagonální metodě. Spor na diagonále si vynutí divergenci, neboť rovnost funkcí je jenom podmíněná, tedy v případě divergence je vše v pořádku.