Задача за минимизация

В тази задача ще използваме наготово аритметиката с нумералите на Church като ще приемем, че имаме работещи основни аритметични функции за числата, както и за тяхното представяне. Също и ще приемем, че имаме работещи основните неща от булевата логика. За момента ще покажем минимизация само за обикновени функции от вида $f : \mathbb{N} \to \mathbb{N}$. Тоест няма да имаме "константни аргументи".

Нека разгледаме терма $\Gamma = \lambda_F \lambda_f \lambda_y (c_=(f\ y)c_0)y(Ff(c_s\ y))$. Тук функцията c_s дава следващото число, тоест $c_s c_x \stackrel{\beta}{=} c_{x+1}$. От сега нататък приемаме, че искаме да минимизираме функцията $f_1 \in \Lambda$. Приемаме, че тя има следната семантика $f_1 c_x \stackrel{\beta}{=} c_y \iff f_2(x) = y$. Термът който ще минимизира f_1 ще бъде $s_1 := Y\Gamma f_1$. Ще докажем, че нашият терм s_1 има семантиката на функцията $s_2(x) = \begin{cases} x & f_2(x) = 0 \\ s_2(x+1) & \exists y > x : f_2(y) = 0 \\ \text{недефинирана} & \nexists y > x : f_2(y) = 0 \end{cases}$.
(тоест, че $s_1 c_x \stackrel{\beta}{=} c_y \iff s_2(x) = y$). Също и ще покажем, че функцията s_2 всъщност извършва минимизацията на f_2.

Случай 1: $f_1 c_x \stackrel{\beta}{=} c_0 \iff f_2(x) = 0$

Трябва да покажем, че $s_1 c_x \stackrel{\beta}{=} c_x$.
$s_1 c_x \equiv (Y\Gamma f_1)c_x \equiv ((\lambda_f(\lambda_x f(xx))(\lambda_x f(xx)))\Gamma f_1)c_x \stackrel{\beta}{=} (((\lambda_x \Gamma(xx))(\lambda_x \Gamma(xx)))f_1)c_x \stackrel{\beta}{=} \Gamma((\lambda_x \Gamma(xx))(\lambda_x \Gamma(xx)))f_1 c_x$. Нека $\Delta := ((\lambda_x \Gamma(xx))(\lambda_x \Gamma(xx)))$.
Тогава $\Gamma((\lambda_x \Gamma(xx))(\lambda_x \Gamma(xx)))f_1 c_x \equiv \Gamma \Delta f_1 c_x \equiv \lambda_F \lambda_f \lambda_y (c_=(f\ y)c_0)y(Ff(c_s\ y))\Delta f_1 c_x \stackrel{\beta}{=} (c_=(f_1\ c_x)c_0)c_x(\Delta f_1(c_s\ c_x))$. От допускането можем да презапишем този израз като $(c_= c_0 c_0)c_x(\Delta f_1(c_s\ c_x)) \stackrel{\beta}{=} c_t c_x(\Delta f_1(c_s\ c_x)) \stackrel{\beta}{=} c_x$. По този начин довършваме случая.

Случай 2: $f_1 c_x \stackrel{\beta}{\neq} c_0 \iff f_2(x) \neq 0$ и $s_2(x)$ е дефинирано

В този случай трябва да покажем, че $s_1 c_x \stackrel{\beta}{=} s_1 c_{x+1}$. По-късно ще видим като доказваме коректността на функцията s_2, че $s_1 c_{x+1}$ ще бъде λ-определимо. Подобно на миналия случай започваме да правим β-редукции $s_1 c_x \equiv (Y\Gamma f_1)c_x \stackrel{\beta}{=} ... \stackrel{\beta}{=} (c_=(f_1\ c_x)c_0)c_x(\Delta f_1(c_s\ c_x))$. От предположението знаем, че този израз е β-еквивалентен на $c_f c_x(\Delta f_1(c_s\ c_x)) \stackrel{\beta}{=} \Delta f_1(c_s\ c_x) \equiv ((\lambda_x \Gamma(xx))(\lambda_x \Gamma(xx)))f_1(c_s\ c_x) \stackrel{\beta}{=} Y\Gamma f_1(c_s\ c_x) \stackrel{\beta}{=} Y\Gamma f_1 c_{x+1} \equiv s_1 c_{x+1}$. Така доказахме случая.

Случай 3: $s_2(x)$ не е дефинирано

Тук трябва да покажем, че $s_1 c_x$ няма нормална форма, но за жалост не знам как :(.

Знаейки вече, че $s_1 c_x$ има същата семантика като $s_2(x)$, то остава смао да направим доказателство, че $s_2(0)$ наистина намира най-малката стойност x, за която $f_2(x) = 0$, ако изобщо съществува такава. Ще докажем това твърдение със силна индукция наобратно. По-конкретно ще докажем следното твърдение:

$$s_2(x) = \begin{cases} y & \exists y \in \mathbb{N}, y \geq x : f_2(y) = 0 \land \nexists z \in \mathbb{N}, x \leq z < y : f_2(z) = 0 \\ \text{недефинирана} & \nexists y \in \mathbb{N}, y \geq x : f_2(y) = 0 \end{cases}$$

Тоест, ще доказваме, че $s_2(x)$ намира най-близкото не по-малко число от x, което е корен на f_2.

База

Нека $y \in \mathbb{N}$, $f_2(y) = 0$ и y е минималното такова число. Нека $x = y$, тогава е ясно, че $s_2(x) = x = y$. Ако такова y не съществува, то от дефиницията на s_2 става ясно, че $s_2(x)$ е недефинирано $\forall x \in \mathbb{N}$, така че вече ще разглеждаме само случая, в който y съществува. Тогава също и термът $s_1 c_y$ ще е ламбда определим и β-еквивалентен на c_y.

Индуктивно предположение

Допускаме, че функцията s_2 се държи както е описано по-горе за всички стойности в интервала $[x+1, y]$, където y е минималният корен на функцията f_2. А пък, ако такова y не съществува, то $s_2(x)$ не е дефинирано. Също и че термът $s_1 c_a$, $a \in [x+1, y]$ е λ-определеим и има семантиката на $s_2(a)$.

Индуктивна стъпка

Ще докажем, че $s_2(x)$ има очакваното поведение използвайки ИП. От минималността на y знаем, че $f_2(x) \neq 0$. Тогава по определението за s_2 знаем, че $s_2(x) = s_2(x+1)$. От индуктивното предположение става ясно, че $s_2(x+1) = y$, което е очаквания резултат. Също и е ясно, че ако y не съществува, то $s_2(x)$ няма да е дефинирано. Също така, тъй като $s_1 c_x$ ще бъде β-еквивалентно на $s_1 c_{x+1}$ в този случай, то $s_1 c_x$ ще има семантиката на $s_2(x)$.

Така вече знаем, че s_2 върши това, което очакваме да върши и можем да заключим, че $s_1 c_0$ наистина намира най малкия корен на функцията f_1, ако той съществува, а иначе няма нормална форма.