Closure Conversion by explicit Lambda Abstraction

Lee Gao (lg342)

April 16, 2013

We want to translate from the nonclosed IL1 language

$$\begin{aligned} v &::= n \mid x \mid \lambda xk.c \mid \mathsf{halt} \\ e &::= v \mid v_0 + v_1 \mid (v_i) \mid \pi_n v \\ c &::= \mathsf{let}\, x = e \,\mathsf{in}\, c \mid v_0\ v_1\ v_2 \end{aligned}$$

into the closed restriction with an explicit abstraction $\bar{\rho} \in \mathsf{var\ list}$ over all of the variables.

$$\begin{aligned} v &::= n \mid x \mid \lambda xk\bar{\rho}.c \mid \mathsf{halt} \\ e &::= v \mid v_0 + v_1 \mid (v_i) \mid \pi_n v \\ c &::= \mathsf{let}\, x = e \,\mathsf{in}\, c \mid v_0\ v_1\ v_2\ \bar{\rho} \end{aligned}$$

Here, we can proceed as natural with the translation functions

$$\begin{aligned} \mathcal{V}[\![v]\!] &: v \\ \mathcal{E}[\![e]\!] &: e \\ \mathcal{C}[\![c]\!] &: c \end{aligned}$$

1 Values

$$\begin{aligned} \mathcal{V}[\![n]\!] &= n \\ \mathcal{V}[\![x]\!] &= \rho_{\langle x \rangle} \\ \mathcal{V}[\![\mathsf{halt}]\!] &= \mathsf{halt} \\ \mathcal{V}[\![\lambda xk.c]\!] &= \lambda xk\bar{\rho}.\,\mathsf{let}\, \rho_{\langle x \rangle} = x, \rho_{\langle k \rangle} = k \,\mathsf{in}\, \mathcal{C}[\![c]\!] \end{aligned}$$

2 Expressions

$$\begin{aligned} \mathcal{E}[\![v]\!] &= \mathcal{V}[\![v]\!] \\ \mathcal{E}[\![v_0 + v_1]\!] &= \mathcal{V}[\![v_0]\!] + \mathcal{V}[\![v_1]\!] \\ \mathcal{E}[\![(v_i)]\!] &= (\mathcal{V}[\![v_i]\!]) \\ \mathcal{E}[\![\pi_n v]\!] &= \pi_n \mathcal{V}[\![v]\!] \end{aligned}$$

3 Commands

$$\mathcal{C}[\![v_0\ v_1\ v_2]\!] = \mathcal{V}[\![v_0]\!]\ \mathcal{V}[\![v_1]\!]\ \mathcal{V}[\![v_2]\!]\ \bar{\rho}$$
$$\mathcal{C}[\![\mathsf{let}\, x = e \,\mathsf{in}\, c]\!] = \mathsf{let}\, \rho_{\langle x \rangle} = \mathcal{E}[\![e]\!] \,\mathsf{in}\, \mathcal{C}[\![c]\!]$$

For example, suppose I have the command

$$c \triangleq \mathsf{let}\, f = \lambda xk.k\ (x+1)\ k \,\mathsf{in}\, f\ 4\ \mathsf{halt}$$

there are three variables here, x, k, f, so we can construct the injection:

$$\langle\cdot\rangle = \{x \mapsto 1, k \mapsto 2, f \mapsto 3\}$$

so

$$\bar{\rho} = \rho_1, \rho_2, \rho_3$$

then the above would translate into

$$\mathcal{C}[\![c]\!] = \mathsf{let}\, \rho_{\langle f \rangle} = \mathcal{E}[\![\lambda xk.k(x+1)k]\!] \,\mathsf{in}\, \mathcal{C}[\![f\ 4\ \mathsf{halt}]\!]$$

expand the $\mathcal{E}$ first

$$\begin{aligned}
&= \mathsf{let}\, \rho_{\langle f \rangle} = \mathcal{V}[\![\lambda xk.k(x+1)k]\!] \,\mathsf{in}\, \mathcal{C}[\![f\ 4\ \mathsf{halt}]\!] \\
&= \mathsf{let}\, \rho_{\langle f \rangle} = \lambda xk\rho_1\rho_2\rho_3.\, \mathsf{let}\, \rho_{\langle x \rangle} = x, \rho_{\langle k \rangle} = k \,\mathsf{in}\, \mathcal{C}[\![k\ (x+1)\ k]\!] \,\mathsf{in}\, \mathcal{C}[\![f\ 4\ \mathsf{halt}]\!] \\
&\vdots \\
&= \mathsf{let}\, \rho_{\langle f \rangle} = \lambda xk\rho_1\rho_2\rho_3.\, \mathsf{let}\, \rho_{\langle x \rangle} = x, \rho_{\langle k \rangle} = k \,\mathsf{in}\, \rho_{\langle k \rangle}(\rho_{\langle x \rangle}+1)\rho_{\langle k \rangle}\rho_1\rho_2\rho_3 \,\mathsf{in}\, \rho_{\langle f \rangle}\ 4\ \mathsf{halt}\ \rho_1\rho_2\rho_3 \\
&= \mathsf{let}\, \rho_3 = \lambda xk\rho_1\rho_2\rho_3.\, \mathsf{let}\, \rho_1 = x, \rho_2 = k \,\mathsf{in}\, \rho_2(\rho_1+1)\rho_{\langle k \rangle}\rho_1\rho_2\rho_3 \,\mathsf{in}\, \rho_3\ 4\ \mathsf{halt}\ \rho_1\rho_2\rho_3
\end{aligned}$$