1 Typed $\lambda_{\textbf{lvar}}$ calculus

1.1 Syntax

Types and environments

$$
\begin{array}{rcll}
T, U & := & \mathbf{1} & \text{unit} \\
& | & T \times U & \text{product} \\
& | & T \to U & \lambda \text{ abstraction} \\
& | & \mathcal{D}^d & D \text{ indexed by } \bigsqcup d \\
& | & \mathcal{L}^J & \text{locations indexed by } J
\end{array}
$$

$$
\begin{array}{rcll}
\Gamma & := & \cdot & \text{empty environment} \\
& | & x : T & \text{environment extension}
\end{array}
$$

Terms and Status bits

$$
\begin{array}{rcll}
L, M, N & := & x & \text{variables} \\
& | & V & \text{values} \\
& | & B & \text{status bits} \\
& | & K & \text{constants} \\
& | & M\ N & \text{parallel application} \\
& | & () & \text{unit introduction} \\
& | & \textbf{let } () = M \textbf{ in } N & \text{unit elimination} \\
& | & (M, N) & \text{product introduction} \\
& | & \textbf{let } (x, y) = M \textbf{ in } N & \text{product elimination}
\end{array}
$$

$$
\begin{array}{rcll}
B & := & 1 & \text{frozen LVar} \\
& | & 0 & \text{unfrozen LVar}
\end{array}
$$

Stores and Configurations

$$
\begin{array}{rcll}
S & := & \cdot & \text{empty store} \\
& | & S, l \mapsto (B, V) & \text{store extension}
\end{array}
$$

$$
\begin{array}{rcll}
C & := & \langle M | S \rangle & \text{programs are pairs of terms and store}
\end{array}
$$

Values

$$
\begin{array}{rcll}
V, W & := & l & \text{locations} \\
& | & J & \text{threshold set} \\
& | & D & \text{distinguished set} \\
& | & \lambda x.M & \lambda \text{ abstraction} \\
& | & () & \text{unit} \\
& | & (M, N) & \text{product}
\end{array}
$$

Constants

$$\begin{array}{rcll} K & \coloneqq & \textbf{new} & \text{allocate new LVar} \\ & \mid & \textbf{freeze} & \text{freeze LVar} \\ & \mid & \textbf{get} & \text{read threshold from LVar} \\ & \mid & \textbf{put} & \text{add value to LVar} \end{array}$$

Evaluation Context

$$\begin{array}{rcl} E & \coloneqq & V\ E \\ & \mid & E\ V \\ & \mid & (V, E) \\ & \mid & (E, V) \\ & \mid & \textbf{let}\ () = E\ \textbf{in}\ M \\ & \mid & \textbf{let}\ (x, y) = E\ \textbf{in}\ M \end{array}$$

1.2 Type System

$$\textsc{T-Var}\ \frac{}{x : T \vdash x : T} \qquad \textsc{T-Lam}\ \frac{\Gamma, x : T \vdash M : U}{\Gamma \vdash \lambda x.M : T \to U} \qquad \textsc{T-App}\ \frac{\Gamma \vdash M : T \to U \qquad \Gamma \vdash N : T}{\Gamma \vdash M\ N : U}$$

$$\textsc{T-Unit}\ \frac{}{\cdot \vdash () : \mathbf{1}} \qquad \textsc{T-LetUnit}\ \frac{\Gamma \vdash M : \mathbf{1} \qquad \Gamma \vdash N : T}{\Gamma \vdash \textbf{let}\ () = M\ \textbf{in}\ N : T}$$

$$\textsc{T-Pair}\ \frac{\Gamma \vdash M : T \qquad \Gamma \vdash N : U}{\Gamma \vdash (M, N) : T \times U} \qquad \textsc{T-LetPair}\ \frac{\Gamma \vdash M : T \times T' \qquad \Gamma, x : T, y : T' \vdash N : U}{\Gamma \vdash \textbf{let}\ (x, y) = M\ \textbf{in}\ N : U}$$

1.3 Operational semantics

Configuration-independent reductions

$$\begin{array}{llll} \textsc{E-Lam} & (\lambda x.M)\ V & \longrightarrow_M & M\{V/x\} \\ \textsc{E-Unit} & \textbf{let}\ () = ()\ \textbf{in}\ M & \longrightarrow_M & M \\ \textsc{E-Pair} & \textbf{let}\ (x, y) = (V, W)\ \textbf{in}\ M & \longrightarrow_M & M\{V/x\}\{W/y\} \end{array}$$

$$\textsc{E-Lift}\ \frac{M \longrightarrow_M M'}{E[M] \longrightarrow_M E[M']}$$

Configuration-dependent reductions

E-New	$\langle E[\mathbf{new}] \mid S\rangle$	$\longrightarrow$	$\langle E[l] \mid S, l \mapsto (0, \bot)\rangle$
E-Freeze	$\langle E[\mathbf{freeze}] \mid S, l \mapsto (b, d)\rangle$	$\longrightarrow$	$\langle E[d] \mid S, l \mapsto (1, d)\rangle$
E-Put	$\langle E[\mathbf{put}\ l\ d] \mid S, l \mapsto (0, d')\rangle$	$\longrightarrow$	$\langle E[()] \mid S, l \mapsto (0, d' \sqcup d)\rangle$
E-RunLVar	$\langle E[\mathbf{runLVar}\ M] \mid S\rangle$	$\longrightarrow$	$\langle E[(\lambda l.\mathbf{let}\ () = M\ l\ \mathbf{in}\ \mathbf{freeze}\ l)\ \mathbf{new}] \mid S\rangle$

E-Get
$$\frac{incomp(J) \quad d' \in J \quad d' \sqsubseteq d}{\langle E[\mathbf{get}\ l\ J] \mid S, l \mapsto (b, d)\rangle \longrightarrow \langle E[d'] \mid S, l \mapsto (b, d)\rangle}$$

E-Pair'
$$\frac{\langle E[M] \mid S\rangle \longrightarrow \langle E[M'] \mid S'\rangle \quad \langle E[N] \mid S\rangle \longrightarrow \langle E[N'] \mid S''\rangle}{\langle E[(M, N)] \mid S\rangle \longrightarrow \langle E[(M', N')] \mid S' \sqcup S''\rangle}$$

E-App
$$\frac{\langle E[M] \mid S\rangle \longrightarrow \langle E[M'] \mid S'\rangle \quad \langle E[N] \mid S\rangle \longrightarrow \langle E[N'] \mid S''\rangle}{\langle E[M\ N] \mid S\rangle \longrightarrow \langle E[M'\ N'] \mid S' \sqcup S''\rangle}$$

E-Step
$$\frac{M \longrightarrow_M M'}{\langle E[M] \mid S\rangle \longrightarrow \langle E[M'] \mid S\rangle}$$