Typing Rules and Evaluation rules

L

June 4, 2023

1 Syntax

$t ::=$		*terms*
	v	
	$if\ term\ then\ term\ else\ term$	
	$succ\ number$	
	$pred\ number$	
	$iszero\ number$	
	$refterm$	
	$t\ t$	
	$wait\ t$	
	$fork\{t\}$	
	$mutex < X1, X2, ..., Xn >$	
	$abstraction$	
	$tagterm$	
	$recordsterm$	
$abstraction ::=$		*abstraction term*
	$\lambda x : T.t$	
	$\lambda < X_1, X_2, ..., X_n > x : T.t$	
	$\lambda < X_1, X_2, ..., X_n > [Y]x : T.t$	

$$
\begin{array}{rll}
refterm ::= & & \texttt{terms about ref} \\
& !t & \\
& ref\ t & \\
& ref < X1, X2, ..., Xn > \ t & \\
& t := t & \\
thterm ::= & & \texttt{terms about thread} \\
& wait\ t & \\
& fork\{t\} & \\
tagterm ::= & & \texttt{terms about tags} \\
& < l = t > \ as\ T & \\
& case\ t\ of\ \ < l_i = x_i > \Longrightarrow\ t_i^{i \in 1..n} & \\
recodesterm ::= & & \texttt{terms about recoders} \\
& l_i = t_i^{i \in 1..n} & \\
& t.l & \\
v = & & \texttt{values} \\
& true & \\
& false & \\
& 0 & \\
& \lambda x : T.t & \\
& string & \\
& unit & \\
& number & \\
& float & \\
& record & \\
& mutex & \\
& loc & \\
& tag & \\
& forkv & \\
& < l = v > \ as\ T & \\
& l_i = v_i^{i \in 1..n} &
\end{array}
$$

2 Typing rules

2.1 Fork

$$\frac{\Gamma|\mathbb{L} \vdash t : \ T}{fork\{\ t\ \} : \ Thread\ T} \quad \text{(T-Fork)}$$

$$\frac{\Gamma|\mathbb{L} \vdash t : \ Thread\ T}{wait\ t \ : \ T} \quad \text{(T-Wait)}$$

2.2 Mutex

$$\frac{}{\Gamma|\mathbb{L} \vdash mutex < X >: \ Mutex\ X} \quad \text{(T-Mutex)}$$

2.3 Acquire

$$\frac{max\{\mathbb{L}\} <_{lex} X \qquad \Gamma|\mathbb{L} \vdash t_1 : Mutex\ X \qquad \Gamma|(\mathbb{L}, X) \vdash t_2 : T}{\Gamma|\mathbb{L} \vdash Aacquire\ t_1\ t_2 : T} \quad \text{(T-Acquire)}$$

2.4 Abstraction

$$\frac{(\Gamma, x : T_1)|(\mathbb{L} \cup \{X_i\}^{i \in 1 \ldots n}) \vdash t : T_2 \qquad mam(\Gamma, T1) = Y}{\Gamma|\mathbb{L} \vdash \lambda < X_i^{i \in 1 \ldots n} > x : T_1.t : \ T_1 < X_i^{i \in 1 \ldots n} > [Y] \to T_2} \quad \text{(T-Abs)}$$

$$\frac{\Gamma|\mathbb{L} \vdash T_1.t : \ T_1 < X_i^{i \in 1 \ldots n} > [Y] \to T_2 \qquad \Gamma|\mathbb{L} \vdash t_2 : T_2}{\Gamma|\mathbb{L} \vdash t_1\ t_2 : \ T_2} \quad \text{(T-App)}$$

2.5 Ref

$$\frac{\Gamma|\mathbb{L} \vdash v : T}{\Gamma|\mathbb{L} \vdash ref < X_i^{i \in 1...n} > v : \ Ref < X_i^{i \in 1...n} > T} \quad \text{(T-Ref)}$$

$$\frac{\Gamma|\mathbb{L} \vdash t_1 : Source < X_i^{i \in 1...n} > T \qquad X_i \in \mathbb{L}^{i \in 1...n}}{\Gamma|\mathbb{L} \vdash !t_1 : T} \quad \text{(T-Deref)}$$

$$\frac{\Gamma|\mathbb{L} \vdash t_1 : Sink < X_i^{i \in 1...n} > T \qquad \Gamma|\mathbb{L} \vdash t_2 : T \qquad X_i \in \mathbb{L}^{i \in 1...n}}{\Gamma|\mathbb{L} \vdash t_1 := t_2 : Unit} \quad \text{(T-Assign)}$$

3 Subtyping rules

3.1 Thread

$$\frac{T_1 <: T_2}{Thread\ T_1 \ <: \ Thread\ T_2} \quad \text{(S-Thread)}$$

3.2 Abstraction

$$\frac{T_1 \ <: \ S_1 \qquad S_2 \ <: \ T_2 \qquad Y_1 \geq_{lex} Y_2 \qquad \{X_i\}^{i \in 1...n} \subseteq \{Z_j\}^{j \in 1...m}}{S_1 < X_i^{i \in 1...n} > [Y_1] \to S_2 \ <: \ T_1 < Z_j^{j \in 1...m} > [Y_2] \to T_2} \quad \text{(S-Arrow)}$$

3.3 Ref

$$\frac{T_1 <: T_2 \quad \{X_i\}^{i\in 1\ldots n} \subseteq \{Z_j\}^{j\in 1\ldots m}}{Source < X_i^{i\in 1\ldots n} > T_1 <: Source < Z_j^{j\in 1\ldots m} > T_2} \quad \text{(S-Source)}$$

$$\frac{T_2 <: T_1 \quad \{X_i\}^{i\in 1\ldots n} \subseteq \{Z_j\}^{j\in 1\ldots m}}{Sink < X_i^{i\in 1\ldots n} > T_1 <: Sink < X_j^{j\in 1\ldots m} > T_2} \quad \text{(S-Sink)}$$

$$\frac{T_1 <: T_2 \quad \{X_i\}^{i\in 1\ldots n} \subseteq \{Z_j\}^{j\in 1\ldots m}}{Ref < X_i^{i\in 1\ldots n} > T_1 <: Source < X_i^{i\in 1\ldots n} > T_1} \quad \text{(S-RefSource)}$$

$$\frac{T_2 <: T_1 \quad \{X_i\}^{i\in 1\ldots n} \subseteq \{Z_j\}^{j\in 1\ldots m}}{Ref < X_i^{i\in 1\ldots n} > T_1 <: Sink < X_i^{i\in 1\ldots n} > T_1} \quad \text{(S-RefSink)}$$

3.4 Thread

$$\frac{T_1 <: T_2}{Thread\ T_1\ <:\ Thread\ T_2} \quad \text{(S-Thread)}$$

4 Algorithmic Typing Rules

5 Evaluation Rules

5.1 Threads

$$\frac{threads = \{t_i\}^{i\in 1\ldots n} \quad < t_{id}, \mu, Th, L > \rightarrow < t'_{id}, \mu', Th', L' > \quad id' = next(id, threads')}{[Th, \mu, id, L] \rightarrow [Th', \mu', id', L']} \quad \text{(E-Thread)}$$

5.2 Wait

$$\frac{threads(p) = v}{[wait\ p, \mu, Th, L] \to [v, \mu, Th \backslash \{p\}, L]} \quad \text{(E-Wait)}$$

$$[fork\ t, \mu, Th, L] \to [p, \mu, Th, L] \quad \text{(E-Fork)}$$

5.3 Abstraction

$$\frac{[t_1, \mu, Th, L] \to [t_1', \mu', Th', L']}{[t_1\ t_2, \mu, Th, L] \to [t_1'\ t_2, \mu', Th', L']} \quad \text{(E-App1)}$$

$$\frac{[t_2, \mu, Th, L] \to [t_2', \mu', Th', L']}{[v_1\ t_2, \mu, Th, L] \to [v_1\ t_2', \mu', Th', L']} \quad \text{(E-App2)}$$

$$[(\lambda x : T.t_{12})\ v_2, \mu, Th, L] \to [t_{12}[x \mapsto v_2], \mu, Th, L] \quad \text{(E-AppAbs)}$$

5.4 Reference

$$\frac{[t_1, \mu, Th, L] \to [t_1', \mu', Th', L']}{[ref < x > \ t_1, \mu, Th, L] \to [ref < x > \ t_1', \mu']} \quad \text{(E-Ref)}$$

$$\frac{l \notin dom(\mu)}{[ref < x > \ v, \mu, Th, L] \to [l, (\mu, l \mapsto v), Th', L']} \quad \text{(E-RefV)}$$

$$\frac{[t_1, \mu, Th, L] \to [t_1', \mu', Th', L']}{[!t_1, \mu, Th, L] \to [!t_1', \mu', Th', L']} \quad \text{(E-Deref)}$$

$$\frac{\mu(l) = v}{[!l, \mu, Th, L] \to [v, \mu, Th, L]} \quad \text{(E-DerefLoc)}$$

$$\frac{[t_1, \mu, Th, L] \to [t_1', \mu', Th', L']}{[t_1 := t_2, \mu, Th, L] \to [t_1' := t_2, \mu', Th', L']} \quad \text{(E-Assign1)}$$

$$\frac{[t_2, \mu, Th, L] \to [t_2', \mu', Th', L']}{[v := t_2, \mu, Th, L] \to [v := t_2', \mu, Th', L']} \quad \text{(E-Assign2)}$$

$$\frac{\mu(l) = v}{[l := v, \mu, Th, L] \to [unit, (\mu, l \mapsto v), Th', L']} \quad \text{(E-AssignV)}$$

5.5 Acquire

$$\frac{[t_1, \mu, Th, L] \to [t_1', \mu', Th', L']}{[acquire\ t_1\ t_2, \mu, Th, L] \to [acquire\ t_1'\ t_2, \mu', Th', L']} \quad \text{(E-Acquire1)}$$

$$\frac{X \notin L}{[acquire\ mutex < X > \ t_2, \mu, Th, L] \to [t_2, \mu, Th, L \cup X]} \quad \text{(E-AcquireV)}$$