# SNESL formalization

Dandan Xue

September 3, 2017

# 0 Level-0

Draft version 0.0.1

## 0.1 Source language syntax

(Ignore empty sequence for now)
Expressions:

$$e ::= x \mid \textbf{let } x = e_1 \textbf{ in } e_2 \mid \varphi(x_1, ..., x_k) \mid \{e : x \textbf{ in } y \textbf{ using } \cdot\}$$

$$\varphi = \textbf{const}_n \mid \textbf{iota} \mid \textbf{plus}$$

Values:

$$n \in \mathbf{Z}$$

$$v ::= n \mid \{v_1, ..., v_k\}$$

## 0.2 Type system

$$\tau ::= \textbf{int} | \{\tau_1\}$$

Type environment $\Gamma = [x_1 \mapsto \tau_1, ..., x_i \mapsto \tau_i]$.

- Judgment $\boxed{\Gamma \vdash e : \tau}$

$$\frac{}{\Gamma \vdash x : \tau} (\Gamma(x) = \tau) \qquad \frac{\Gamma \vdash e_1 : \tau_1 \qquad \Gamma[x \mapsto \tau_1] \vdash e_2 : \tau}{\Gamma \vdash \textbf{let } x = e_1 \textbf{ in } e_2 : \tau}$$

$$\frac{\vdash Typ_\varphi(\tau_1, ..., \tau_k) : \tau}{\Gamma \vdash \varphi(x_1, ..., x_k) : \tau} ((\Gamma(x_i) = \tau_i)_{i=1}^k) \qquad \frac{\Gamma[x \mapsto \tau_1] \vdash e : \tau}{\Gamma \vdash \{e : x \textbf{ in } y \textbf{ using } \cdot\} : \{\tau\}} (\Gamma(y) = \{\tau_1\})$$

- Auxiliary Judgment $\boxed{\vdash Typ_\varphi(\tau_1, ..., \tau_k) : \tau}$

$$\frac{}{\vdash \textbf{const}_n() : \textbf{int}} \qquad \frac{}{\vdash \textbf{iota}(\textbf{int}) : \{\textbf{int}\}} \qquad \frac{}{\vdash \textbf{plus}(\textbf{int}, \textbf{int}) : \textbf{int}}$$

## 0.3 Source language semantics

- Judgment $\boxed{\rho \vdash e \downarrow v}$

$$\frac{}{\rho \vdash x \downarrow v} (\rho(x) = v) \qquad \frac{\rho \vdash e_1 \downarrow v_1 \qquad \rho[x \mapsto v_1] \vdash e_2 \downarrow v}{\rho \vdash \textbf{let } e_1 = x \textbf{ in } e_2 \downarrow v}$$

$$\frac{\vdash Eva_\varphi(v_1, ..., v_k) \downarrow v}{\rho \vdash \varphi(x_1, ..., x_k) \downarrow v} ((\rho(x_i) = v_i)_{i=1}^k) \qquad \frac{([x_i \mapsto v_i] \vdash e \downarrow v_i')_{i=1}^k}{\rho \vdash \{e : x \textbf{ in } y \textbf{ using } \cdot\} \downarrow \{v_1', ..., v_k'\}} (\rho(y) = \{v_1, ..., v_k\})$$

- Auxiliary Judgment $\boxed{\vdash\ Eva_{\varphi}(v_1, ..., v_k) \downarrow v}$

$$\frac{}{\vdash\ \mathbf{const}_n() \downarrow n} \qquad \frac{}{\vdash\ \mathbf{iota}(n) \downarrow \{0, 1, ..., n-1\}} \qquad \frac{}{\vdash\ \mathbf{plus}(n_1, n_2) \downarrow n_3}\ (n_3 = n_1 + n_2)$$

## 0.4 SVCODE syntax

Stream id:

$$s \in \mathbf{N} = \{0, 1, 2...\}$$

Stream tree:

$$st ::= s \mid (st_1, s)$$

SVCODE expressions:

$$\varphi ::= \mathtt{Ctrl} \mid \mathtt{Const_a} \mid \mathtt{ToFlags} \mid \mathtt{Usum} \mid \mathtt{MapTwo} \mid \mathtt{ScanPlus}$$

SVCODE program:

$$\begin{aligned} p ::= &\ \epsilon \\ &\mid s := \psi(s_1, ..., s_i) \\ &\mid st := \mathtt{WithCtrl}(s, p) \\ &\mid p_1; p_2 \end{aligned}$$

Target language values:

$$b \in \{\mathtt{T}, \mathtt{F}\}$$
$$a ::= n \mid b \mid ()$$
$$\rightharpoonup v ::= \langle a_1, ..., a_i \rangle$$

Define some operations on $\rightharpoonup v$ for convenience:

- $\rightharpoonup v_1 \mathbin{++} \rightharpoonup v_2$: append $\rightharpoonup v_2$ to $\rightharpoonup v_1$
- $\mathtt{tail}(\langle a_1, a_2, ..., a_i \rangle) = \langle a_2, ..., a_i \rangle$

## 0.5 SVCODE semantics

- Judgment $\boxed{\langle p, \sigma \rangle \downarrow^{\rightharpoonup v_c} \sigma'}$

$$\frac{}{\langle \epsilon, \sigma \rangle \downarrow^{\rightharpoonup v_c} \sigma} \qquad \frac{SEva_{\varphi}(\rightharpoonup v_1, ..., \rightharpoonup v_k) \downarrow^{\rightharpoonup v_c} \rightharpoonup v}{\langle s := \varphi(s_1, ..., s_k), \sigma \rangle \downarrow^{\rightharpoonup v_c} \sigma[s \mapsto \rightharpoonup v]}\ ((\sigma(s_i) = \rightharpoonup v_i)_{i=1}^{k})$$

$$\frac{}{\langle st := \mathtt{WithCtrl}(s, p), \sigma \rangle \downarrow^{\rightharpoonup v_c} \sigma[st \rightarrowtail \langle\rangle]}\ (\sigma(s) = \langle\rangle)$$

$$\frac{\langle p, \sigma \rangle \downarrow^{\rightharpoonup v_s} \sigma'}{\langle st := \mathtt{WithCtrl}(s, p), \sigma \rangle \downarrow^{\rightharpoonup v_c} \sigma'}\ (\sigma(s) = \rightharpoonup v_s = \langle a_1, ..., a_i \rangle)$$

$$\frac{\langle p_1, \sigma \rangle \downarrow^{\rightharpoonup v_c} \sigma'' \qquad \langle p_2, \sigma'' \rangle \downarrow^{\rightharpoonup v_c} \sigma'}{\langle p_1; p_2, \sigma \rangle \downarrow^{\rightharpoonup v_c} \sigma'}$$

- Auxiliary Judgment $\boxed{SEva_{\varphi}(\rightharpoonup v_1, ..., \rightharpoonup v_k) \downarrow^{\rightharpoonup v_c} \rightharpoonup v}$

$$\frac{Block_{\varphi}(\rightharpoonup v_{11}, ..., \rightharpoonup v_{k1}) \Downarrow \rightharpoonup v_1 \qquad SEva_{\varphi}(\rightharpoonup v_{12}, ..., \rightharpoonup v_{k2}) \downarrow^{\mathtt{tail}(\rightharpoonup v_c)} \rightharpoonup v_2}{SEva_{\varphi}(\rightharpoonup v_{11} \mathbin{++} \rightharpoonup v_{12}, ..., \rightharpoonup v_{k1} \mathbin{++} \rightharpoonup v_{k2}) \downarrow^{\rightharpoonup v_c} \rightharpoonup v}\ (\rightharpoonup v = \rightharpoonup v_1 \mathbin{++} \rightharpoonup v_2)$$

$$\frac{}{SEva_{\varphi}(\rightharpoonup v_1, ..., \rightharpoonup v_k) \downarrow^{\langle\rangle} \langle\rangle}$$

- Auxiliary Judgment $\boxed{Block_{\varphi}(\rightharpoonup v_1, ..., \rightharpoonup v_k) \Downarrow \rightharpoonup v}$

$$\frac{}{\texttt{Const}_{\texttt{a}} \Downarrow \langle a \rangle} \qquad \frac{}{\texttt{ToFlags}(\langle n \rangle) \Downarrow \langle \texttt{F}_1, ..., \texttt{F}_n, \texttt{T} \rangle} \qquad \frac{}{\texttt{MapTwo}(\langle n_1 \rangle, \langle n_2 \rangle) \Downarrow \langle n_3 \rangle}\ (n_3 = n_1 + n_2)$$

$$\frac{Unary_{\varphi}(\langle \texttt{F} \rangle, ..., \rightharpoonup v_{k1}) \Downarrow \rightharpoonup v_1 \qquad Block_{\varphi}(\rightharpoonup v_{12}, ..., \rightharpoonup v_{k2}) \Downarrow \rightharpoonup v_2}{Block_{\varphi}(\langle \texttt{F} \rangle \mathbin{++} \rightharpoonup v_{12}, ..., \rightharpoonup v_{k1} \mathbin{++} \rightharpoonup v_{k2}) \Downarrow \rightharpoonup v}\ (\rightharpoonup v = \rightharpoonup v_1 \mathbin{++} \rightharpoonup v_2)$$

$$\frac{Unary_{\varphi}(\langle \texttt{T} \rangle, ..., \rightharpoonup v_k) \Downarrow \rightharpoonup v}{Block_{\varphi}(\langle \texttt{T} \rangle, ..., \rightharpoonup v_k) \Downarrow \rightharpoonup v}$$

$$\frac{Unary_{\varphi, n_0}(\langle \texttt{F} \rangle, ..., \rightharpoonup v_{k1}) \Downarrow^{n_0'} \langle n_1 \rangle \qquad Block_{\varphi, n_0'}(\rightharpoonup v_{12}, ..., \rightharpoonup v_{k2}) \Downarrow \rightharpoonup v_2}{Block_{\varphi, n_0}(\langle \texttt{F} \rangle \mathbin{++} \rightharpoonup v_{12}, ..., \rightharpoonup v_{k1} \mathbin{++} \rightharpoonup v_{k2}) \Downarrow \langle n_1 \rangle \mathbin{++} \rightharpoonup v_2}$$

$$\frac{Unary_{\varphi, n_0}(\langle \texttt{T} \rangle, ..., \rightharpoonup v_k) \Downarrow \langle n_1 \rangle}{Block_{\varphi, n_0}(\langle \texttt{T} \rangle, ..., \rightharpoonup v_k) \Downarrow \langle n_1 \rangle}$$

- Auxiliary Judgment $\boxed{Unary_{\varphi}(\langle b \rangle, ..., \rightharpoonup v_k) \Downarrow \rightharpoonup v}$

$$\frac{}{\texttt{Usum}(\langle \texttt{F} \rangle) \Downarrow \langle () \rangle} \qquad \frac{}{\texttt{Usum}(\langle \texttt{T} \rangle) \Downarrow \langle \rangle}$$

$$\frac{}{\texttt{ScanPlus}_{n_0}(\langle \texttt{F} \rangle, \langle n \rangle) \Downarrow^{n_0 + n} \langle n_0 \rangle} \qquad \frac{}{\texttt{ScanPlus}_{n_0}(\langle \texttt{T} \rangle, \langle \rangle) \Downarrow \langle n_0 \rangle}$$

## 0.6 Translation

- $$\frac{\delta \ \vdash \ e \xRightarrow[s_1]{s_0+1} (p, st)}{\delta \ \vdash \ e \xRightarrow[s_1]{s_0} \texttt{let } s_0 := \texttt{Ctrl}; p \texttt{ in } st}$$

- Judgment $\boxed{\delta \ \vdash \ e \xRightarrow[s_1]{s_0} (p, st)}$

$$\frac{}{\delta \ \vdash \ x \xRightarrow[s_0]{s_0} (p, st)}\ (\delta(x) = st) \qquad \frac{\delta \ \vdash \ e_1 \xRightarrow[s_0']{s_0} (p_1, st_1) \qquad \delta[x \mapsto st_1] \ \vdash \ e_2 \xRightarrow[s_1]{s_0'} (p_2, st)}{\delta \ \vdash \ \textbf{let } x = e_1 \textbf{ in } e_2 \xRightarrow[s_1]{s_0} (p_1; p_2, st)}$$

$$\frac{Trans_{\varphi}(st_1, ..., st_k) \xRightarrow[s_1]{s_0} (p, st)}{\delta \ \vdash \ \varphi(x_1, ..., x_k) \xRightarrow[s_1]{s_0} (p, st)}\ ((\delta(x_i) = st_i)_{i=1}^{k})$$

$$\frac{[x \mapsto st_1] \ \vdash \ e \xRightarrow[s_1]{s_0+1} (p, st)}{\delta \ \vdash \ \{e : x \textbf{ in } y \textbf{ using } \cdot\} \xRightarrow[s_1]{s_0} (s_0 := \texttt{Usum}(s_2); st := \texttt{WithCtrl}(s_0, p), st)}\ (\delta(y) = (st_1, s_2))$$

- Auxiliary Judgment $\boxed{Trans_{\varphi}(st_1, ..., st_k) \xRightarrow[s_1]{s_0} (p, st)}$

$$\frac{}{\textbf{const}_a() \xRightarrow[s_0+1]{s_0} (s_0 := \texttt{Const}_{\texttt{a}}, s_0)}$$

$$\frac{}{\mathbf{iota}(s) \overset{s_0}{\underset{s_4}{\Longrightarrow}} (p, (s_3, s_0))} \left(\begin{array}{l} s_{i+1} = s_i + 1 \\ p = s_0 := \texttt{ToFlags}(s); \\ \quad s_1 := \texttt{Usum}(s_0); \\ \quad s_2 := \texttt{WithCtrl}(s_1, s_2 := \texttt{Const}_1); \\ \quad s_3 := \texttt{ScanPlus}(s_0, s_2) \end{array}\right)$$

$$\frac{}{\mathbf{plus}(s_1, s_2) \xrightarrow[s_0+1]{s_0} (s_0 := \texttt{MapTwo}(s_1, s_2), s_0)}$$

## 0.7 Value representation

*TODO: define $\sigma(st)$*

Judgment $\boxed{\sigma \vdash v \triangleright_\tau st}$

$$\frac{}{\sigma \vdash n \triangleright_{\mathbf{int}} s} (\sigma(s) = \langle n \rangle) \qquad \frac{(\sigma \vdash n_i \triangleright_{\mathbf{int}} s_i)_{i=1}^k}{\sigma \vdash \{n_1, ..., n_k\} \triangleright_{\{\mathbf{int}\}} (s, s')} \left(\begin{array}{l} \sigma(s) = \sigma(s_1) +\!\!+\, \sigma(s_2) +\!\!+ ... +\!\!+\, \sigma(s_k) \\ \sigma(s') = \langle \texttt{F}_1, ..., \texttt{F}_k, \texttt{T} \rangle \end{array}\right)$$

$$\frac{(\sigma \vdash v_i \triangleright_\tau (st_i, s_i))_{i=1}^k}{\sigma \vdash \{v_1, ..., v_k\} \triangleright_{\{\tau\}} ((st, s), s')} \left(\begin{array}{l} \tau \neq \mathbf{int} \\ \sigma(st) = \sigma(st_1) +\!\!+\, \sigma(st_2) +\!\!+ ... +\!\!+\, \sigma(st_k) \\ \sigma(s) = \sigma(s_1) +\!\!+\, \sigma(s_2) +\!\!+ ... +\!\!+\, \sigma(s_k) \\ \sigma(s') = \langle \texttt{F}_1, ..., \texttt{F}_k, \texttt{T} \rangle \end{array}\right)$$

## 0.8 Correctness proof